Oracle® Scripting

Implementation Guide

Release 11*i*

Part No. B10180-04

August 2004

Oracle Scripting Implementation Guide, Release 11*i*

Part No. B10180-04

Primary Author: Samuel Besalel

Contributors: Norman Chan, Elizabeth Chambers-Carbone, Kumar Pandey, Rathna Sundaralingam, Pradeep Kotha, Vish Acharya, Allen Chan, Anirban Chanda, Sheralyn Fowler.

Contents

2 Introduction

3 Before You Begin

4 Implementation Tasks

5 Implementation Administration Tasks

6 Scripting Administration Tasks

11 Integrating Oracle Scripting

A Upgrading from Releases Prior to 11.5.6

B Oracle Scripting Profile Options

C Oracle Discoverer Workbooks

Send Us Your Comments

Oracle Scripting Implementation Guide, Release 11*i*

Part No. B10180-04

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: appsdoc_us@oracle.com
- FAX: (650) 506-7200 Attn: Oracle Applications Documentation Manager
- Postal service:
  Oracle Corporation
  Oracle Applications Documentation Manager
  500 Oracle Parkway
  Redwood Shores, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Audience for This Guide

Welcome to Release 11*i* of the Oracle Scripting Implementation Guide.

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.
- Oracle Scripting

  If you have never used Oracle Scripting, Oracle suggests you attend one or more of the Oracle Scripting training classes available through Oracle University.
- The Oracle Applications graphical user interface.

  To learn more about the Oracle Applications graphical user interface, read the *Oracle Applications User's Guide*.

See Other Information Sources for more information about Oracle Applications product information.

How To Use This Guide

This guide contains the information you need to understand and use Oracle Scripting. Sections include:

The **Introduction** includes an overview of Oracle Interaction Center software products.

Detailed Product Description describes which aspects of Oracle Scripting are new, modified, or obsolete in this release, and discusses various features of Oracle Scripting. This section includes information on Oracle Scripting's architecture, and lists responsibilities that may be required for using Oracle Scripting. Concepts sections are also included for each component of Oracle Scripting.

Before You Begin mentions related product documentation, identifies various dependencies and minimum software and hardware requirements, and provides general scalability and performance guidelines. Information regarding installing and accessing Oracle Applications are also described.

Implementation Tasks lists an implementation task sequence, serving as a map to the tasks required for your implementation. This section then includes a discussion to help you understand various Oracle Scripting users, and specific requirements for creating administrator users and agent users. Described in detail are tasks for implementing the Scripting Engine agent and web interfaces, as well as describing details for setting up two administrative interfaces and when each is required.

Implementation Administration Tasks includes administration tasks that may be required for implementation of Oracle Scripting. These include the creation and administration of Oracle Applications users; steps for setting up debugging and message logging; enabling the Oracle JInitiator Java Console; overriding the requirement for an agent's membership in a resource group with Sales role; and how to grant JTF roles for administrators. Also included are steps for setting system profile values, finding application and responsibility ID values, determining a default application based on a responsibility ID value, how to find the FND User_ID for a known user, and how to remove the Quick Find menu from the Scripting Administration console.

Scripting Administration Tasks documents administration performed from the Scripting Administration console. This section describes functionality which allows users to launch Script Author as a Java applet, administer Oracle Scripting files such as deployed scripts and custom Java, and generate and view a panel footprint report.

Survey Resources Administration Tasks includes tasks required to administer survey resources supporting two technology stacks, which are displayed when executing scripts in the Scripting Engine Web interface using a Web browser. These steps are not relevant for implementations using the Scripting Engine agent interface only.

Survey Campaign Administration Tasks documents administration performed from the Survey Administration console. This section details administration of survey campaigns, including controlling the view of defined items, and defining, modifying, and deleting survey campaigns, cycles, and deployments. You can view tasks describing how to activate deployments or to view and interpret responses to active deployments or send reminders to list members for targeted deployments.

Survey campaign creation and administration tasks are prerequisite to executing *any* script in a Web browser using the Scripting Engine Web interface. This section also includes tasks for setting up invitations or reminders for targeted deployments, and describes administration required for concurrent programs used in support of survey operations for Oracle Scripting.

Campaign and List Administration Tasks addresses administration of Oracle Marketing where administration of this application affects Oracle Scripting users. For example, if using Oracle Scripting from Oracle TeleSales or Oracle Collections, you must first administer campaigns in Oracle Marketing. Additionally, in order to execute survey campaigns using the survey component of Oracle Scripting, Oracle Marketing lists are required. These tasks are detailed here for your convenience. For more information on administration and implementation of Oracle Marketing, refer to *Oracle Marketing Implementation Guide*.

Implementation Verification includes a description of tasks to verify each component of Oracle Scripting, including a tutorial to create and deploy a simple script (using the Script Wizard or using graphical Script Author tools) to assist in testing the Scripting Engine interfaces.

Integrating Oracle Scripting discusses integration of Oracle Scripting with other applications by Oracle Scripting component, by technology stack, and by business application integration. This section addresses specific steps required to integrate Oracle Scripting with business applications including Oracle TeleSales, Oracle Collections, and Oracle TeleService. Additionally, integration with Oracle Marketing (a list and brief description of that application's building block scripts) are discussed.

Appendix A, Upgrading from Releases Prior to 11.5.6, provides specific information to assist you in upgrading from any pre-11.5.6 release of Oracle Scripting to any subsequent release.

Appendix B, Oracle Scripting Profile Options, provides specific information on profile options for Oracle Scripting and other Oracle applications settings required to implement various Oracle Scripting components.

Appendix C, Oracle Discoverer Workbooks, provides information on the survey reports that can be generated for analysis of survey information using Oracle Discoverer workbooks.

There are also **Abbreviations and Acronyms** and **Glossary** appendices to assist you in decoding many abbreviations, acronyms and terms.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/

Accessibility of Code Examples in Documentation

JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Other Information Sources

You can choose from many sources of information, including documentation, training, and support services, to increase your knowledge and understanding of Oracle Scripting.

If this guide refers you to other Oracle Applications documentation, use only the Release 11*i* versions of those guides.

Online Documentation

All Oracle Applications documentation is available online (HTML or PDF). Some online help patches are available on Oracle*MetaLink* or Oracle iSupport. Oracle Scripting online help is available for the Scripting Administration and Survey Administration components.

All Oracle Applications documentation is available online (HTML or PDF).

- **PDF Documentation-** See the Online Documentation CD for current PDF documentation for your product with each release. This Documentation CD is also available on Oracle*MetaLink* and is updated frequently.
- **Online Help -** You can refer to Oracle Applications Help for current HTML online help for your product. Oracle provides patchable online help, which you can apply to your system for updated administrator documentation. Oracle Scripting includes online help for the Scripting Administration console and the Survey Administration console. No online help is available for end users of the Scripting Engine or Script Author components of Oracle Scripting. No system downtime is required to apply online help.
- **Release Content Document -** See the Release Content Document for descriptions of new features available by release. The Release Content Document is available on Oracle*MetaLink*.
- **About document -** Refer to the About document for information about your release, including feature updates, installation information, and new documentation or documentation patches that you can download. The About document is available on Oracle*MetaLink*.

Related Guides

Oracle Scripting shares business and setup information with other Oracle Applications products. Therefore, you may want to refer to other guides when you set up and use Oracle Scripting.

You can read the guides online by choosing Library from the expandable menu on your HTML help window, by reading from the Oracle Applications Document Library CD included in your media pack, or by using a Web browser with a URL that your system administrator provides.

If you require printed guides, you can purchase them from the Oracle Store at http://oraclestore.oracle.com.

Guides Related to All Products

Oracle Applications User's Guide

This guide explains how to enter data, query, run reports, and navigate using the graphical user interface (GUI). This guide also includes information on setting user profiles, as well as running and reviewing reports and concurrent processes.

You can access this user's guide online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

Guides Related to This Product

Oracle Scripting Developer's Guide 11i

This guide is intended for advanced users of Script Author. Included is information on the various Script Author commands, how to use them and where to define them; a description of Best Practice Java methods you can use to extend your scripts; and detailed information on performing advanced tasks in a graphical script. This document lists seeded reusable commands, instructing on their use and integration with seeded building block scripts; and describes each building block script. Information on Best Practice survey scripts is also included. Finally, extensive information is included regarding customizing scripts, with particular attention to customizing panel layouts.

Oracle Scripting Implementation Guide 11*i*

This guide describes how to implement Oracle Scripting components and test the implementation appropriately. This guide also details task-based steps for administering Oracle Scripting. You can access Oracle Scripting administration procedures online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

Oracle Scripting User Guide 11i

This guide explains how to understand the various components of Oracle Scripting. It includes information on planning Oracle Scripting implementations, details specifics on using Script Author to create graphical and wizard scripts, and describes how to execute scripts using the Scripting Engine (in both the agent and Web interfaces).

Installation and System Administration

Oracle Applications Concepts

This guide provides an introduction to the concepts, features, technology stack, architecture, and terminology for Oracle Applications Release 11*i*. It provides a useful first book to read before an installation of Oracle Applications. This guide also introduces the concepts behind Applications-wide features such as Business Intelligence (BIS), languages and character sets, and Self-Service Web Applications.

Installing Oracle Applications

This guide provides instructions for managing the installation of Oracle Applications products. In Release 11*i*, much of the installation process is handled using Oracle Rapid Install, which minimizes the time to install Oracle Applications and the Oracle technology stack by automating many of the required steps. This guide contains instructions for using Oracle Rapid Install and lists the tasks you need to perform to finish your installation. You should use this guide in conjunction with individual product user guides and implementation guides.

Upgrading Oracle Applications

Refer to this guide if you are upgrading your Oracle Applications Release 10.7 or Release 11.0 products to Release 11*i*. This guide describes the upgrade process and lists database and product-specific upgrade tasks. You must be either at Release 10.7 (NCA, SmartClient, or character mode) or Release 11.0, to upgrade to Release 11*i*. You cannot upgrade to Release 11*i* directly from releases prior to 10.7.

About Document

For information about implementation and user documentation, instructions for applying patches, new and changed setup steps, and descriptions of software updates, refer to the "About" document for your product. "About" documents are available on Oracle*MetaLink* for most products starting with Release 11.5.8.

Maintaining Oracle Applications

Use this guide to help you run the various AD utilities, such as AutoUpgrade, AutoPatch, AD Administration, AD Controller, AD Relink, License Manager, and others. It contains how-to steps, screenshots, and other information that you need to run the AD utilities. This guide also provides information on maintaining the Oracle applications file system and database.

Oracle Applications System Administrator’s Guide

This guide provides planning and reference information for the Oracle Applications System Administrator. It contains information on how to define security, customize menus and online help, and manage concurrent processing.

Oracle Alert User’s Guide

This guide explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

Oracle Applications Developer’s Guide

This guide contains the coding standards followed by the Oracle Applications development staff and describes the Oracle Application Object Library components that are needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. This manual also provides information to help you build your custom Oracle Forms Developer forms so that the forms integrate with Oracle Applications.

Oracle Applications User Interface Standards for Forms-Based Products

This guide contains the user interface (UI) standards followed by the Oracle Applications development staff. It describes the UI for the Oracle Applications products and how to apply this UI to the design of an application built by using Oracle Forms.

Other Implementation Documentation

Oracle Applications Product Update Notes

Use this guide as a reference for upgrading an installation of Oracle Applications. It provides a history of the changes to individual Oracle Applications products between Release 11.0 and Release 11*i*. It includes new features, enhancements, and changes made to database objects, profile options, and seed data for this interval.

Oracle Workflow Administrator's Guide

This guide explains how to complete the setup steps necessary for any Oracle Applications product that includes workflow-enabled processes, as well as how to monitor the progress of runtime workflow processes.

Oracle Workflow Developer's Guide

This guide explains how to define new workflow business processes and customize existing Oracle Applications-embedded workflow processes. It also describes how to define and customize business events and event subscriptions.

Oracle Workflow User's Guide

This guide describes how Oracle Applications users can view and respond to workflow notifications and monitor the progress of their workflow processes.

Oracle Workflow API Reference

This guide describes the APIs provided for developers and administrators to access Oracle Workflow.

Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Oracle Scripting implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This guide also provides information on creating custom reports on flexfields data.

Oracle eTechnical Reference Manuals

Each eTechnical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for a specific Oracle Applications product. This information helps you convert data from your existing applications, integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. Oracle eTRM is available on Oracle*Metalink*

Oracle Applications Message Manual

This manual describes all Oracle Applications messages. This manual is available in HTML format on the documentation CD-ROM for Release 11*i*.

Training and Support

Training

Oracle offers training courses to help you and your staff master Oracle Scripting and reach full productivity quickly.

You have a choice of educational environments. You can attend courses offered by Oracle University at any one of our many Education Centers, you can arrange for our trainers to teach at your facility, or you can use Oracle Learning Network (OLN), Oracle University's online education utility. In addition, Oracle training professionals can tailor standard courses or develop custom courses to meet your needs. For example, you may want to use your organization structure, terminology, and data as examples in a customized training session delivered at your own facility.

Support

From on-site support to central support, our team of experienced professionals provides the help and information you need to keep Oracle Scripting working for you. This team includes your Technical Representative, Account Manager, and Oracle’s large staff of consultants and support specialists with expertise in your business area, managing an Oracle8*i* server, and your hardware and software environment.

Oracle*MetaLink*

Oracle*MetaLink* is your self-service support connection with web, telephone menu, and e-mail alternatives. Oracle supplies these technologies for your convenience, available 24 hours a day, 7 days a week. With Oracle*MetaLink*, you can obtain information and advice from technical libraries and forums, download patches, download the latest documentation, look at bug details, and create or update TARs. To use Oracle*MetaLink*, register at http://metalink.oracle.com/.

Do Not Use Database Tools to Modify Oracle Applications Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using Oracle Applications can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

About Oracle

Oracle develops and markets an integrated line of software products for database management, applications development, decision support, and office automation, as well as Oracle Applications, an integrated suite of more than 160 software modules for financial management, supply chain management, manufacturing, project systems, human resources and customer relationship management.

Oracle products are available for mainframes, minicomputers, personal computers, network computers and personal digital assistants, allowing organizations to integrate different computers, different operating systems, different networks, and even different database management systems, into a single, unified computing and information resource.

Oracle is the world's leading supplier of software for information management, and the world's second largest software company. Oracle offers its database, tools, and applications products, along with related consulting, education, and support services, in over 145 countries around the world.

Your Feedback

Thank you for using Oracle Scripting and this implementation guide.

Oracle values your comments and feedback. In this guide is a reader's comment form that you can use to explain what you like or dislike about Oracle Scripting or this guide. Mail your comments to the following address:

Oracle Corporation
Oracle Applications Documentation Manager
500 Oracle Parkway
Redwood Shores, CA 94065
USA

1 Detailed Product Description

This section includes the following topics:

- New in this Release
- Modified in this Release
- Obsolete in this Release
- Features
- Architecture
- Responsibilities
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.1 New in this Release

This section includes the following topics:

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks
- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments

- New Table Added to IES Schema
- New Profile Settings
- Scripts Can Now Be Deployed from the Command Line

See Also

- Modified in this Release
- Obsolete in this Release
- Features
- Architecture
- Responsibilities
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.1.1 Single Sign-On Support for Scripting Engine Agent Interface

The Scripting Enginee agent UI now supports Single Sign-On (SSO) requirements for all Oracle E-Business Suite Applications. When using the Oracle Forms-based interface to execute scripts, Oracle Scripting transactions use SSO authentication methods. This is an architectural upgrade to ensure Oracle Scripting is consistent with other applications. There is no change for end users of the Scripting Engine agent interface (whether executing scripts from Oracle TeleSales, Oracle TeleService, or Oracle Collections).

Exceptions

- If the Oracle Applications session times out (if an ICX session timeout occurs) during a script transaction in which a Forms command is invoked, the user must again log into an Oracle Applications session (using the same authentication method and user account as was used to start the expired session) before the Forms command will execute successfully.

See Also

- Web Interface Execution Supports Two Technology Stacks

- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Table Added to IES Schema
- New Profile Settings
- Scripts Can Now Be Deployed from the Command Line

1.2 Web Interface Execution Supports Two Technology Stacks

Scripts can be executed in the Scripting Engine Web interface using either the Oracle Applications Framework (OA Framework or OAF) technology stack or the Oracle CRM Technology Foundation (JTT) technology stack. This option is established as a survey campaign property and, once saved, cannot be changed.

When a survey campaign is established using the new Base Tech Stack option set to *OA Framework*, the script runs using the new OAF technology stack. This technology stack supports HTML survey resources, and eliminates the requirement for a survey administrator to define survey resources. If no header or footer section resource is defined, the header or footer respectively will simply not appear in the header or footer sections of the HTML page at runtime.

If no error page or final page resource is defined, default error page and final pages will be provided by the application, as appropriate.

This technology stack also supports using a URL to redirect users to a specific Web page (either upon encountering a server-side runtime error, or at the end of the script), instead of specifying specific error or final page survey resources, respectively. This technology stack does not support JSP survey resources.

If the Base Tech Stack option is set to *Deprecated - JTT*, the script continues to execute in a Web browser using the JTT technology stack. Each survey campaign set up in this technology stack still requires JSP survey resources to be designated as the header section, error page and final page. The footer section remains optional.

Oracle Corporation strongly recommends that all customers using the Scripting Engine Web interface plan and implement new survey deployments using the new Oracle Applications Framework functionality. The older technology stack is deprecated and will eventually be obsolete.

Survey administrators select which technology stack a new survey campaign uses at runtime by selecting the appropriate option from the Base Tech Stack list on the Create Survey Campaign page.

New features and functionality are available using the new technology stack, and will continue to be introduced. Most of these improvements will not be available to the older technology stack.

Runtime Support for Redirecting for Correct Technology Stack

In order to eliminate potential problems with Oracle Scripting supporting two technical stacks, the following functionality has been added to the product in support of Scripting Engine Web interface operations:

- If a deployment ID for an OAF survey deployment is submitted to IESSVYMAIN.JSP, the system will redirect the JSP template portion of the URL to OA.JSP for display. The required database connection file parameter will also be added as appropriate.
- If a deployment ID for an OAF survey deployment is submitted to IESSVYMENUBASED.JSP, an error message displays, indicating that the requested deployment is an OAF survey and that the JSP can only support JTT-based surveys.
- If a deployment ID for a JTT survey deployment is submitted to OA.JSP, the system will redirect the JSP template portion of the URL to IESSVYMAIN.JSP for display.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Table Added to IES Schema
- New Profile Settings
- Scripts Can Now Be Deployed from the Command Line

1.2.1 OA Framework Survey Resources Support HTML and URL Redirect

Survey campaigns using the new OA Framework architecture support HTML survey resources (saved as `.HTM` or `.HTML` files). OA Framework survey campaigns do *not* support JavaServer Page (JSP) survey resources.

> **Note:** The deprecated JTT technology stack still requires all survey resources in JSP format. These do not require actual dynamic content (they can be pure HTML), but must be saved as `.JSP` files.

The physical HTML file corresponding to an OA Framework survey resource definition is uploaded from the Define Survey Resources page of the Survey Administration console.

OA Framework survey campaigns do not require survey resources to be associated to execute scripts at runtime. If header section and footer section survey resources are not associated with the survey campaign, then headers or footers will simply not appear in the header or footer sections of the HTML page at runtime. If no specific error page or final page resource is specified, a simple error page or final page will be automatically generated by the application.

Instead of using a specified HTML page when an error is generated by the application or upon completion of the script, a URL can be designated, which redirects the user of the script to an appropriate location on a Web server.

Oracle Corporation strongly recommends that all customers using the Scripting Engine Web interface plan and implement new survey deployments using the new Oracle Applications Framework functionality, including HTML survey resources (when appropriate) or URL redirects for error and final pages.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Table Added to IES Schema
- New Profile Settings

- Scripts Can Now Be Deployed from the Command Line

1.2.2 Targeted Survey Deployments Support Over 200 List Members

Targeted survey deployments use Oracle One-to-One Fulfillment to send invitations and reminders (known to Oracle One-to-One Fulfillment as master documents) to members on an Oracle Marketing list. Oracle Scripting now calls an Oracle One-to-One Fulfillment SUBMIT_MASS application program interface (API), which supports an unlimited number of respondents in a deployment. In previous releases, each deployment supported a maximum of approximately 200 list members. This limitation is actually based on the number of characters in the concatenated "To:" address line.

Related to this improvement in functionality, all Oracle Scripting implementations using the Scripting Engine Web interface for targeted (list-based) deployments must update the query associated with each master document to ensure it conforms to the query format required by the updated API.

For more information, see Query Syntax Requirements for Targeted Survey Deployments.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks
- OA Framework Survey Resources Support HTML and URL Redirect
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Table Added to IES Schema
- New Profile Settings
- Scripts Can Now Be Deployed from the Command Line

1.2.3 Interaction History and Fulfillment Content History Tracking Enabled

Interaction History Transaction Tracking for Script Execution

The Scripting Engine now records Oracle Customer Interaction History activity records during script execution, regardless of execution interface. The only requirement is that the business application invoking the script passes a valid interaction ID. If so, Oracle Scripting creates an activity record, and adds the script transaction ID to that record, providing a linkage between the interaction record (from an Oracle Customer Interaction History perspective) and the questions and responses provided during execution of the script, as well as linking any timing and footprinting information recorded for the script transaction.

The activity created by Scripting will specify an activity type of **Script** (when executed in the Scripting Engine agent interface) or **Survey** (when executed in a Web browser using the Scripting Engine Web interface).

As of this release, Oracle TeleSales and Oracle Collections both pass the interaction ID and thus causes the interaction ID to be tracked and the script transaction to be linked.

Suspending Tracking for Sending Survey Master Documents

As of Oracle Scripting release 11.5.10 or Interaction Center Family Pack R, Oracle Scripting's integration with Oracle One-to-One Fulfillment inherits the new capability to capture Oracle Customer Interaction History tracking when invitation and reminder e-mail messages are sent. Oracle Customer Interaction History tracking can be overridden at the fulfillment server level, to stop the recording in Oracle Customer Interaction History of the invitation and reminder e-mail messages sent by the fulfillment server. (This does not affect tracking of interaction IDs for script execution, which occurs regardless). Note, however, that overriding tracking of invitation and reminder master documents in support of survey campaigns is *not recommended by Oracle Corporation.* This behavior should not be overridden unless you are experiencing performance issues. Even in this case, the implications of changing the default behavior to stop recording Oracle Customer Interaction History interactions should first be considered carefully.

For more information regarding interaction IDs, see *Oracle Customer Interaction History Implementation Guide*.

Oracle One-to-One Fulfillment E-Mail Tracking

As of Oracle Scripting release 11.5.10 or Interaction Center Family Pack R, Oracle Scripting's integration with Oracle One-to-One Fulfillment inherits the new capability to capture e-mail tracking statistics. Specifically, statistics captured include the number of e-mail messages that are sent, unopened, opened (if "E-mail Tracking" option is enabled), malformed, and bounced back. The statistics are displayed for each "mass" e-mail request in the Request History details page of Oracle One-to-One Fulfillment. This is referred to as *content history tracking.*

In releases 11.5.8 and 11.5.9, the fulfillment server had two parameters that could be used to override content history and interaction history tracking. These parameters are **-Dhistory.NoContents** (referred to in Oracle Interaction Center Server Manager as Content History) and **-Dhistory.NoInteractions** (referred to in Oracle Interaction Center Server Manager as No Interactions). The tracking occurs when products in the Oracle E-Business Suite call Oracle One-to-One Fulfillment APIs.

As of this release, the fulfillment server is improved by adding more robust error recovery. Part of the error recovery functionality is dependent upon the data written to content history. As part of this enhancement, the -Dhistory.NoContents parameter has been deprecated and is no longer interpreted by the fulfillment server. From this release and onward, content history is always written, regardless of the setting of the the -Dhistory.NoContents flag. The -Dhistory.NoInteractions parameter is still valid.

References

Additional details on viewing Oracle One-to-One Fulfillment content history and Oracle Customer Interaction History can be found in the product documentation for Oracle One-to-One Fulfillment and Oracle Customer Interaction History, respectively. Specifically:

- For more information on the fulfillment server, see *Oracle One-to-One Fulfillment Implementation Guide*.
- For more information regarding tracking in Oracle Customer Interaction History of e-mail messages sent by Oracle One-to-One Fulfillment, see *Oracle Customer Interaction History Implementation Guide*.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks
- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Table Added to IES Schema
- New Profile Settings
- Scripts Can Now Be Deployed from the Command Line

1.2.4 Query Syntax Requirements for Targeted Survey Deployments

As a result of Oracle Scripting’s migration to the Oracle One-to-One Fulfillment SUBMIT_MASS API, implementations using the Scripting Engine Web interface for targeted (list-based) deployments must update the query associated with each master document to ensure it conforms to the query format required by the API.

All targeted survey deployments must have a query associated with the survey deployment's master document. This query must follow a defined format with respect to column aliases and content. Oracle Scripting provides a base template query (IES Template Query for Targeted Survey Deployments) that is seeded in the list of Oracle One-to-One Fulfillment queries. The code for this query is provided in the section Seeded Survey List Query.

References

For exact query syntax and guidelines for use of this query, see Seeded Survey List Query.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks
- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Oracle Marketing List Filter Enabled for Survey Deployments
- New Table Added to IES Schema
- New Profile Settings
- Scripts Can Now Be Deployed from the Command Line

1.2.5 Oracle Marketing List Filter Enabled for Survey Deployments

Oracle Marketing offers some filtration of marketing lists. If a customer is logically removed from an Oracle Marketing list (stored in table AMS_LIST_ENTRIES), the Enabled_Flag parameter is set to "N." Applications taking advantage of this filtration would exclude flagged list records from the final list executed. In previous releases of Oracle Scripting, this parameter was not evaluated. As of this release (Oracle Interaction Center Family Pack R or later or release 11.5.10 or later), any active survey deployment associated with a list entry for which the parameter is set to "N" will not be sent invitation or reminder master documents.

For more information on how to use Oracle Marketing list management functionality to filter list contents, refer to Oracle Marketing product documentation.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks
- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Table Added to IES Schema
- New Profile Settings
- Scripts Can Now Be Deployed from the Command Line

1.2.6 New Table Added to IES Schema

A new transactional data table, IES_FOOTPRINTING_DATA, has been added to the IES schema to support footprinting. For more information, see Schema Changes for Oracle Scripting Footprinting.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks
- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Profile Settings
- Scripts Can Now Be Deployed from the Command Line

1.2.7 New Profile Settings

A new system profile, IES: Survey Resources Caching, determines whether survey resources for Oracle Applications Framework surveys are cached at runtime. Valid values are On and Off. If no value is set, caching occurs by default (same behavior as setting to On).

For more information, see Oracle Scripting Profile Options.

In support of Oracle Scripting's Oracle Applications Logging Framework Migration, several profiles which affect message logging are now required to be configured. These include the following profiles:

- FND: Diagnostics
- FND: Debug Log Enabled
- FND: Debug Log Filename
- FND: Debug Log Level
- FND: Debug Log Module
- Guest User Password

The Guest User Password profile is only relevant if you want to view message logs for scripts executed in the Scripting Engine Web interface as a guest user.

For specifics on these profiles, see Oracle Scripting Profile Options > Information, Error and Debug Logging Profile Options.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks
- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Table Added to IES Schema
- Scripts Can Now Be Deployed from the Command Line

1.2.8 Scripts Can Now Be Deployed from the Command Line

Prior to this release, the only way to deploy a Script Author script into an Oracle Applications 11*i* environment was to open the script using Script Author, and either deploy it from the graphical tools, or deploy the script from the Script Wizard save options.

Now, you can deploy scripts at the file level from the command line. Either script creation method (using graphical tools or the Script Wizard) is supported for deploying scripts from the command line, providing the script is saved to the file system as a graphical script.

Since this does not provide syntax validation, you *must* ensure that you open the script you want to deploy using the Script Author applet, validate the syntax of the script (making any changes necessary to pass the syntax check), and save the script to a local or network file system prior to deploying it from the command line.

Published scripts cannot be deployed using this method.

Manually move the graphical script file to the APPL_TOP of the applications tier prior to deploying from the command line. As another prerequisite, your environment must be set correctly.

Limitations

- Wizard scripts cannot be deployed directly using this method. You must first graph the wizard script and then save it to the file system prior to deploying using this method.
- Published scripts cannot be deployed using this method.
- Script Author applet used must be in the same environment from which the command line compiler is run.
- No product patching can be performed after saving the script and prior to deploying it from the command line.
- Scripts must be syntactically correct to use this deployment method.
- Scripts created prior to Oracle Scripting release 11.5.9 must be converted to a current Script Author version prior to successful deployment.

See Also

- Single Sign-On Support for Scripting Engine Agent Interface
- Web Interface Execution Supports Two Technology Stacks

- OA Framework Survey Resources Support HTML and URL Redirect
- Targeted Survey Deployments Support Over 200 List Members
- Interaction History and Fulfillment Content History Tracking Enabled
- Query Syntax Requirements for Targeted Survey Deployments
- New Table Added to IES Schema
- New Profile Settings

1.3 Modified in this Release

This section includes the following topics:

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

See Also

- New in this Release
- Obsolete in this Release

- Features
- Architecture
- Responsibilities
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.3.1 Updated Terminology

Answers, Answer Definitions, or Nodes Are Now Questions

Question user interface controls are the controls you select in Script Author when creating or editing a script, in order to add a text field, text area, radio button, check box, button, drop-down list, password field, checkbox group, or multi-select list box to a panel in a script.

In the past, question UI controls have sometimes been referred to as *nodes*, *answers*, or *answer definitions*. For example, in the graphical Script Author UI, when defining a question UI control you must select Answers from the panel properties tree. The window in which UI controls are designated is referred to as the Answer Entry Dialog window.

In keeping with newer terminology adopted with introduction of the Script Wizard feature, and in the interest of clarity, these will now be referred to either as questions or as question UI controls.

The use of the word *node* to depict a starting point or ending point on the canvas of a graphical script is unchanged.

Lookup Values Are Now Answer Choices

In keeping with newer terminology adopted with introduction of the Script Wizard feature, and in the interest of clarity, the property of questions formerly referred to as lookup values will now be referred to as answer choices. In graphical scripts, answer choices can be associated with any non-text type question UI (other than the single checkbox, which takes its value from the Label for Reporting property). In wizard scripts, radio buttons, drop-down lists, checkbox groups and multi-select list UI types accept answer choices.

Thank You Pages Are Now Final Pages

For survey resources referenced in the Survey Administration console component of Oracle Scripting, the user interface sometimes uses the term Thank You Page and sometimes uses the term Final Page. Both of these terms refer to the page resource displayed after the final panel in a script executed in a Web browser in the Scripting Engine Web interface. These two terms are equivalent; the term primarily used in this document is Final Page resource.

List-Based Deployments Are Now Targeted Deployments

For survey campaigns to be executed by individuals receiving an e-mail message with an invitation or reminder to participate, the targeted sample or population (identified by an Oracle Marketing list) are now referred to as targeted deployments.

Setting a Survey Deployment to Active Status is Now Called Activating

Using the Survey Administration console, after creating a deployment for a specific parent cycle (which in turn belongs to a specific parent survey campaign), you must set its status to Active prior to execution in the Scripting Engine Web interface. This process is now known as *activating a deployment*. In previous releases, it was referred to as deploying a deployment. The UI is updated to reflect this terminology adjustment.

Static Panel is Now Script Information Area

When defining text fields or timers in a graphical script to appear in the Scripting Engine agent interface at runtime, the UI navigation tree for the properties of the global script includes a Static Panel option. The area in which static or dynamic text or timers appears above the panel presentation area at runtime is now referred to as the script information area, to reflect the ability of this feature to contain dynamic information.

Shortcut Panel is Now Shortcut Button Area

When defining programmable buttons in a graphical script that appear in the Scripting Engine agent interface at runtime, the UI navigation tree for the properties of the global script includes a Shortcut Panel option. The area in which defined shortcut buttons appear is now referred to as the shortcut button area.

Disconnect Groups and WrapUp Groups

The Disconnect group is automatically included in every script created using the Script Wizard. This group contains a shortcut property of WrapUpShortcut, and a group name of Disconnect. The subgraph of this group contains no panels, but is syntactically correct, containing a default branch linking the start node and the termination node. At runtime, in the Scripting Engine agent interface, this group is called when the end user clicks the Disconnect button, bringing the script from whichever panel is in focus to a valid state of completion.

This group is now distinguished from a WrapUp group, which typically contains one or more panels to collect information from each script end user prior to exiting the script. Note, however, that both typically contain the WrapUpShortcut shortcut name. Only one of these two groups should be included in a script. Each must exist at the root graph level, and each should be connected with a default branch to a termination node on the root graph.

See Also

- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.2 Some Oracle Scripting Building Blocks Superseded

Since release 11.5.6, Oracle Applications has been seeded with several scripts that were designed to provide samples of how developers using Script Author could invoke Oracle E-Business APIs in order to perform some useful tasks for business-to-business (B2B) interactions. The scripts are associated with reusable

Script Author Java commands that are also seeded with the product. These scripts (referred to as Oracle Scripting building blocks) and the seeded Script Author commands (available from the Script Author command library) have been documented in *Oracle Scripting Developer's Guide*.

As of Oracle Scripting release 11.5.9, the functionality in several of the seeded Oracle Scripting building blocks has been superseded by scripts delivered with Oracle Marketing. These superseded building blocks include:

Script	**Function**
IESEVENT.SCR	Register B2B Contact for Event
IESINTC.SCR	Create B2B Contact Interest
IESINTO.SCR	Create Org Interest

The above Oracle Scripting seeded commands and building blocks are still functional, and are supported (to the limited degree that custom components are warranted to function). However, their use is not recommended.

The robust seeded scripts that replace the seeded Oracle Scripting building block scripts were developed by the Oracle Marketing team. These new building block scripts expand the offering previously available to include scripts for business-to-consumer (B2C) functions in addition to B2B functions. The new Oracle Marketing scripts also take advantage of changes to the APIs used to create leads and events that have been introduced in Oracle Marketing release 11.5.9. The API changes included in the new scripts will not be reflected in the original set of Oracle Scripting building block scripts.

Oracle Marketing building block scripts are seeded in Oracle Applications on the applications tier, and can be downloaded from the applications server from the product top directory ($AMS_TOP/scripts).

References

- The seeded Oracle Marketing scripts are documented in Oracle Marketing Implementation Guide for release 11.5.9, available on Oracle*MetaLink* (part number B10588-01).
- These scripts are briefly described in Integrating with Oracle Marketing.

See Also

- Updated Terminology

- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.3 Oracle Applications Logging Framework Migration

As of this release, Oracle Scripting has migrated from IEO server logging functionality to the Oracle Applications Logging Framework, for those aspects of error and informational message logging supported by that technology stack. This new integration ensures that administrators can look in either the central log viewer application, or a single central logging file (if established), for error and informational messages associated with Oracle applications. The changes that affect users include configuring product logging settings, and the method of viewing error and informational message logging output.

In previous versions, users were required to log into Oracle CRM Technology Foundation (JTT) technology stack HTML-based applications as a user with either the Interaction Center Server Log Viewer, the Interaction Center Server Manager, the Call Center HTML Administration, or the CRM HTML Administration responsibility. Server logs were then accessed specific to product code IEO. For Oracle Scripting, IEO server logging is now obsolete. As of Oracle Scripting release 11.5.10 or Interaction Center Family Pack R, all messages are logged and viewed using the Oracle Applications logging framework.

Logs (if enabled) are viewed in the common logging infrastructure from HTML applications in the Oracle Applications Framework (OAF) technology stack only.

Thus, to view Oracle Scripting error, informational, and debug message logs, you can log into the Survey Administration console as a user with the Survey Administrator responsibility, and click the Diagnostics global link or icon.

Based on the profile setting and level, the Diagnostics global link appears in the UI only if the FND: Diagnostics profile is set to **Yes**. This is typically set at the user level.

Four FND profile settings affect the manner in which information is logged. Sample settings for these profiles for general use are provided below. For more information, see the guidelines following.

Profile	Recommended Level	Setting
FND: Debug Log Enabled	Application	Yes
FND: Debug Log Filename	Application	(null)
FND: Debug Log Level	Application	Error
FND: Debug Log Module	Application	ies%

Guidelines

- The above samples are intended for general use. If specific issues are encountered, you may want to modify certain profiles. For example, to diagnose errors, set FND: Debug Log Level to **Statement** at the user level to provide more information. If encountering problems with specific application modules (for example, the Scripting Engine), you may want to set FND: Debug Log Module to **ies.engine** at the user level.
- If enabled, log messages by default are stored in the applications database. If a log filename (for example, apps.log) is provided, message logs can be stored to a single physical file.
- Some system resources are consumed by enabling log messages. Thus, it is recommended to set log messaging at the lowest appropriate level.
- Enterprises may have varying needs for accessing log files. A knowledgeable system administrator is aware of the impact of logging at various levels, and its impact on performance (if any). Your system administrator may have a schedule to archive logs or clear them on a certain schedule, and may have requirements for setting logging at various levels for different applications. For final determination, consult your system administrator.

- This integration includes a new set of APIs to be used in custom code (one for each of the six levels of logging: Unexpected, Error, Exception, Event, Procedure, and Statement) to log messages to the same common infrastructure. Implementations using custom code that require viewing error or informational messages will need to incorporate the appropriate API into their custom code, and redeploy that code in order to view those messages using the Applications Common Logging Infrastructure. For more information, see *Oracle Scripting Developer's Guide*.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.4 Schema Changes for Oracle Scripting Footprinting

To resolve an issue where the sequence of panel records in the footprinting tables (and the individual survey response view which depends on that data) may differ from the actual sequence of panels visited by the survey respondent, slight modifications to the footprinting schema have been introduced.

The two transactional data tables (IES_QUESTION_DATA and IES_PANEL_DATA) are now joined by a new third table named IES_FOOTPRINTING_DATA.

No changes have been made to the structure of the IES_QUESTION_DATA table. As in previous releases, the column PANEL_DATA_ID in the IES_QUESTION_DATA table continues to be a foreign key to the PANEL_DATA_ID column in the IES_PANEL_DATA table.

As before, the IES_PANEL_DATA table contains one record for each panel instance that is created at runtime for a script session in the Scripting Engine. Thus, each occurrence of a panel being viewed for the first time in a script interaction results in a record being added to the IES_PANEL_DATA table. This table no longer contains records for activation of *existing* panel instances (these records, created when you go back to a panel previously visited during the current script interaction, are now stored in the new IES_FOOTPRINTING_DATA table). Records in IES_PANEL_DATA are retained even if (due to a change in flow at runtime) the panel represented by the record is deleted from the progress area. In this case, the value in the DELETED_STATUS column is equal to 1. The SEQUENCE column represents the final sequence of all displayed panel instances at the end of transaction (deleted panels have no sequence). The ELAPSED_TIME column in this table is no longer used, and will be set to null for all new records. This information is now stored in the new table.

In the new IES_FOOTPRINTING_DATA table, each record represents an activation of a panel instance (either by first displaying the panel, or by enabling the panel after it has already been displayed in the current script interaction). The column PANEL_DATA_ID in the IES_FOOTPRINTING_DATA table is a foreign key to the PANEL_DATA_ID column in the IES_PANEL_DATA table.

Each record includes the column ELAPSED_TIME (representing the total time, in milliseconds, that the panel instance was in an enabled state). The SEQUENCE column represents the sequence of each activation (this can be different than the sequence in IES_PANEL_DATA).

These are schema changes only; there are no changes in visible user functionality.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack

- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.5 Debug Mode Behavior Modified

This section addresses two system profile options for debugging.

Prior to this release, the system profile option IES : Debug Mode controlled a variety of components. In Interaction Center release FP-R and Oracle Scripting release 11.5.10 and later, this profile option only controls the Scripting Engine Java runtime client (the Scripting Engine agent interface). When enabled, all errors that occur on the client side are sent to the Oracle JInitiator console (for example, if trying to click Continue to exit a panel without first selecting a selection from the drop-down list).

For information on setting up this console to view errors, see Enabling the Oracle JInitiator Java Console

Oracle Corporation recommends disabling Debug Mode unless trying to troubleshoot a specific issue with the forms/Java-based Oracle Scripting client known as the Scripting Engine Agent interface. Debug mode is for providing additional dialog messages to narrow a problem when one exists.

The IES : AUTHOR DEBUG MODE system profile option is now reserved. Do not set this profile option unless directed to do so by Oracle Support services.

References

- Enabling the Oracle JInitiator Java Console
- Oracle Scripting Profile Options

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded

- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.6 Survey Execution Supports OA Framework Technology Stack

This release supports the Scripting Engine Web interface (execution of a survey or Web script in an Oracle Applications 11*i*-certified Web browser) using the Oracle Applications Framework technology stack. This runtime functionality is *in addition to* continuing support of Web interface execution in the deprecated Oracle CRM Technology Foundation (JTT) technology stack.

Concurrent functionality of both technology stacks ensures that Oracle Corporation customers using long-term deployments (for example, deployments spanning six months to one year) can continue to execute and complete all survey campaign deployments without interruption. This ability helps ensure that the data collated from survey deployments using the older technology stack is valid, consistent, and thorough, in keeping with the business objectives of executing the survey campaign. Users maintain the ability to compare survey deployment data between deployments in the same technology stacks and between deployments executed using the old technology stack and its eventual replacement.

Oracle Corporation strongly recommends that all customers using the Scripting Engine Web interface plan and implement new survey deployments using the new Oracle Applications Framework functionality. The older technology stack is deprecated and will eventually be obsolete. Survey administrators select which technology stack new survey campaigns use at runtime by selecting the appropriate option from the Base Tech Stack list on the Create Survey Campaign page. The options listed include **OA Framework** (the new technology stack) and **Deprecated - JTT** (the previous technology stack).

New features and functionality are available using the new technology stack, and will continue to be introduced. Most of these improvements will not be available to the older technology stack.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.7 Survey Resource Changes for OA Framework Technology Stack

Survey Resources for the deprecated Oracle CRM Technology Foundation (JTT) technology stack do not change. As before, header section, error, and final page resources are required, and footer section resources are optional. All survey resources for the old technology stack must be HTML or JSP files saved with a `.JSP` file extension. The physical files corresponding to the survey resources defined in the Survey Administration console must be uploaded to the APPL_TOP (in $OA_HTML) independently from the user interface.

In contrast, survey resources for the new Oracle Applications Framework technology stack include several changes, as summarized below:

- OAF survey campaigns are not dependent on the inclusion of survey resources; survey resources are optional for OAF survey campaigns.
- OAF survey campaigns use HTML files (saved with an `.HTML` or `.HTM` file extension) as survey resources. JSP files are *not* supported by OA Framework.
- HTML survey resources can be uploaded to the database from the Define Survey Resources page of the Survey Administration Console.
- OAF survey campaigns can associate a URL on an existing Web server to use as an error page or a final page (instead of an HTML or JSP page). When appropriate, the runtime survey or Web script user is redirected to the specified URL.
- If you designate a URL survey resource as the error page, the script end user's browser is redirected to the specified URL when an error occurs at runtime. In addition, the following key-value pairs will be appended to the URL:

Information Type	Key
Error information	ERR
Deployment ID	dID
Transaction ID	tID
Respondent ID	rID

Other aspects of OAF survey resources include the following:

- If you do not specify a header section or footer section, no header or footer displays at runtime.
- If you do not specify a final page or URL, a default page will appear at the completion of the script displaying a simple thank you message.
- If you do not specify an error page or URL, a default page will appear displaying information about the error from the application.
- If you select **Display Error on Top** next to the Error Page field, application-generated error information displays on top of the custom error page when an error occurs at runtime. This is accessible only to OAF survey campaigns.

Just as with JTT survey campaigns, a survey resource must be defined (from the Survey Resources tab) before it can be associated with a survey campaign. Survey resources can be changed at any time for an active survey campaign.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.8 Oracle Applications Guest User Replaces JTT Guest User

When a script is executed in the Scripting Engine Web interface in a non-hosted scenario, an Oracle Applications session is initiated using a guest user login. From an Oracle Scripting perspective, this Oracle Applications login allows the user to execute a script in a Web browser only, granting no other privileges in the Oracle Applications session. No responsibilities are typically associated with the guest user.

In consonance with Oracle Corporation's migration to common standards and components, as of Interaction Center Family Pack R or later or release 11.5.10 or later, the Scripting Engine relies upon the Oracle Applications guest user. In previous releases, this guest user was set up and configured as a JTT advanced system property.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded

- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.9 Responsibility Change for Targeted Surveys

For Oracle Scripting implementations using targeted (list-based) surveys for Oracle Interaction Center Family Pack R or later or release 11.5.10 or later, you must use an updated Oracle One-to-One Fulfillment responsibility. In previous releases, the Fulfillment Administrator responsibility was appropriate. From this release forward, a new responsibility, One-to-One Fulfillment Administrator, is also available. *You must use this new responsibility in order to access the new Oracle One-to-One Fulfillment menu structure.* This requires accessing each Oracle Applications account with the old responsibility, locating all users with the Fulfillment Administrator responsibility, terminating access to that responsibility (by providing an end effective date), and associating the new One-to-One Fulfillment Administrator responsibility with the user (with a current effective date).

For more information, see *Oracle One-to-One Fulfillment Implementation Guide* for Oracle Interaction Center Family Pack R or later, or Oracle Applications release 11.5.10 or later.

See Also

- Updated Terminology

- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys

1.3.10 Changes in Suspend and Restart Functionality

When the Suspendable global property of a graphical script is **true** (when suspendability is enabled), and a script transaction being executed in the agent interface is suspended, commands executed in a suspended segment which access external environments will now be re-executed during transaction restart. The ability to use this functionality is dependent upon the launching application being appropriately enabled; each application must provide a method through which users can identify suspended script segments and specify which one to resume. The script suspension and resumption functionality is only available to the Scripting Engine agent interface.

As another improvement, the state of the shortcut button area is now restored at restart to its state at the time the transaction was suspended.

Finally, if the Apache session times out in the middle of a Scripting Engine agent interface transaction, the Scripting Engine now automatically suspends the transaction. When the user first attempts to perform an activity in the script after the timeout, Oracle Scripting attempts to start a new Scripting session in the Apache JVM, and then restarts the suspended transaction. If the Oracle Applications session times out as well (if an ICX session timeout occurs), the user must again log into an Oracle Applications session (using the same authentication method and user account as was used to start the expired session) before the restart process can begin.

Note: You must log into the personal home page (PHP) in order to execute scripts using the Scripting Engine agent interface. As of Interaction Center Family Pack R or later or release 11.5.10 or later, Oracle Scripting does not support the f60cgi URL (commonly referred to as the "Forms login").

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.11 Support for List Management Filtering in Targeted Surveys

Enhancements have been made to Oracle Scripting’s integration with Oracle Marketing, to better support advanced features of Oracle Marketing list management. When obtaining respondents from Oracle Marketing lists, all list records with an enabled_flag of "N" are now filtered out. Thus, any filtering of list entries performed in Oracle Marketing (including sampling) are now reflected in the final list used for survey deployment.

Details on how using list management functionality is found in Oracle Marketing product documentation.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.12 Improvements in Database Connection Editing

In past releases, database connection details were specified in Script Author dialogs accessible from the object dictionary of a block, or in the command window for a PL/SQL command.

These windows have been changed. Connection information available in these windows is now restricted to using the login connection (the default) or reusing an existing connection.

Database connections used by Oracle Scripting at runtime are now defined in two ways:

1. By logging into an Oracle Applications session (which provides the login connection).
2. By selecting **Script > Connection List** from Script Author.

To define a new connection in the connection list, from the Connection List window, click **Add**.

Connection information includes the connection name, host name, port number, database SID, username, and password. For authentication, enter a username with apps privileges (for example, in a test instance, use apps/apps).

To modify an existing database connection, from the Connection List window, click **Edit**.

Note that any changes made to the connections in this list will affect all objects which reference the connection within the script.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.13 Survey Administration Console User Interface Changes

Survey Campaign Tab

- The Create Survey Campaign page now includes a new property, "Base Tech Stack." This drop-down list, in the Create Survey Campaign area, includes two options, OA Framework and Deprecated-JTT. This property is now displayed as a new column in the Survey Campaigns List page. The only time this property can be set is upon survey campaign creation. This property is not modifiable for a survey campaign.

- The Survey Campaign Details and Deployment Details pages are now referred to as Update Survey Campaign and Update Deployment pages, respectively.
- The Create Survey Campaign page and the Update Survey Campaign page now include an additional boolean property associated with the error page survey resource. This checkbox only applies to OAF survey campaigns, and is thus context-sensitive (it only appears when the base technology stack is set to OA Framework). The checkbox appears to the right of the Error Page field in the Survey Resources area of these pages. Setting this property to **true** (selecting the box) causes application-generated error information to display at the top of the custom error page.
- On the Update Deployment page, the Deploy button is now labelled Activate.
- On the Update Deployment page, for targeted deployments, in the Hosting Options area, the Survey Web URL field changes based on the base tech stack. For example, for a targeted JTT deployment, the known parameters for the survey URL appear (and is not modifiable). For a stand-alone deployment, the URL appears in a modifiable text field, allowing you to change information such as the Apache web server or port.

Survey Resources Tab

The Define Survey Resources page is now context sensitive to allow for survey resource differences between the JTT and OAF technology stacks.

- A new Resource Type field lists the type of resource being defined. **HTML File Upload** and **URL for Redirect** apply to OAF survey campaigns only. **Deprecated - JSP** applies to JTT survey campaigns only.
- When **HTML File Upload** is selected as the resource type, a new File to Upload field appears. This enables you to upload a physical HTML file to the database as the physical survey resource corresponding to the survey resource definition you are creating.
- When **URL for Redirect** is selected as the resource type, a Resource URL field appears. Into this field you must enter the hypertext transfer protocol URL of a page to display at runtime in lieu of a defined error or final page.
- When **Deprecated - JSP** is selected as the resource type, a Resource JSP field appears. This corresponds to the field previously labeled File Name, and identifies the JSP file name that must be loaded manually to $OA_HTML in order for this definition to function as a survey resource at runtime.

Search and Select Pages

- Implementations with the latest OAF code will have access to a Quick Select column anytime you search the database.
- When searching and selecting survey resources, the results table now includes a Resource Type column which indicates if the resource is JSP, HTML, or a URL.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Script Author User Interface Changes
- Script Launching Behavior Modified

1.3.14 Script Author User Interface Changes

Connection Dialog Changes

Database connection information is no longer entered into the Object Dictionary of a block or the Connection area of the Command Properties window for a PL/SQL command. Connection information available in these windows is now restricted to using the login connection (the default) or reusing an existing connection. Definition of "existing connections" is now performed in a graphical script by selecting **Script > Connection List** from the Script Author menu.

For more information, see Improvements in Database Connection Editing

Menu Changes

- The **Script** menu now has a **Connection List** option.
- The **Tools** menu now has a **Wizard** option, from which the Script Wizard is launched.

Toolbar Changes

The toolbar now contains a Script Wizard icon.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Launching Behavior Modified

1.3.15 Script Launching Behavior Modified

When you click **Launch Script Author** from the Home page of the Scripting Administration console, a new Web browser window opens, and the Oracle JInitiator applet launches. Subsequently, Script Author opens in a separate window.

In previous releases, if a session timeout occurred, returning to the Home page and clicking Launch Script Author refreshed the existing JInitiator session. The existing Script Author window would then again be accessible, with all connections restored.

To support changes in Oracle JInitiator versions 1.3.x and later that may cause Script Author to close to to hang, the behavior of the Script Author applet launching method has changed.

As of Oracle Scripting release 11.5.10 or Interaction Center Family Pack R, a new Web browser window with the JInitiator plug-in displayed opens each time the Launch Script Author button is pressed from the Home tab of the Survey Administration console. As before, the existing Script Author window is again accessible, with all connections restored.

When a Script Author session times out, *do not attempt to close or refresh the first Oracle JInitiator browser window* from an expired session. Doing so causes the Script Author applet to close or hang.

Instead, return to the Home page of the Scripting Administration console and click any button or link. You will be prompted to log in again. After you provide this information and return to the Home page of the Scripting Administration console, click **Launch Script Author**. A new browser window opens for a fresh session of the Oracle JInitiator plugin, and then the existing Script Author window again gains focus.

See Also

- Updated Terminology
- Some Oracle Scripting Building Blocks Superseded
- Oracle Applications Logging Framework Migration
- Schema Changes for Oracle Scripting Footprinting
- Debug Mode Behavior Modified
- Survey Execution Supports OA Framework Technology Stack
- Survey Resource Changes for OA Framework Technology Stack
- Oracle Applications Guest User Replaces JTT Guest User
- Responsibility Change for Targeted Surveys
- Changes in Suspend and Restart Functionality
- Support for List Management Filtering in Targeted Surveys
- Improvements in Database Connection Editing
- Survey Administration Console User Interface Changes
- Script Author User Interface Changes

1.4 Obsolete in this Release

This section includes the following topics:

- Requirements for Logging Responsibilities Obsolete
- Lead Creation Building Block and Seeded Command Obsolete

See Also

- New in this Release
- Modified in this Release
- Features
- Architecture
- Responsibilities
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.4.1 Requirements for Logging Responsibilities Obsolete

As of Oracle Scripting release 11.5.10 or Interaction Center Family Pack R, with the migration to the Oracle Applications Logging Framework, no specific responsibilities are required to be assigned to users of Oracle Scripting in order to view message logs. In previous versions, users were required to log into Oracle CRM Technology Foundation (JTT) technology stack HTML-based applications as a user with either the Interaction Center Server Log Viewer, the Interaction Center Server Manager, the Call Center HTML Administration, or the CRM HTML Administration responsibility.

The CRM HTML Administration responsibility is still required to implement Oracle Scripting in order to configure JTF advanced system properties. However, users and administrators no longer need access to the Interaction Center Server Log Viewer, the Interaction Center Server Manager, or the Call Center HTML Administration responsibilities.

New system profile setting requirements are now applicable for the purpose of logging and viewing message logs. For more information, see Oracle Applications Logging Framework Migration.

See Also

- Lead Creation Building Block and Seeded Command Obsolete

1.4.2 Lead Creation Building Block and Seeded Command Obsolete

Prior to Oracle Applications release 11.5.10 or Interaction Center Family Pack R, leads were managed using the Oracle TeleSales business application.

As of this release, leads are objects created and managed in the Oracle Marketing application. The basic structure of a lead has recently been modified by that product development team. The parameters collected in the Oracle Scripting lead creation building block script were intended for use with leads created by Oracle TeleSales. The APIs and functions in that building block script therefore do not correlate with the appropriate information required for a lead valid for Oracle Marketing. Thus, although the Oracle Scripting building block functionality *may* still appear functional at runtime, executing scripts using this functionality does *not* create a valid lead according to the business rules of the Oracle Marketing application. It is therefore obsolete and no longer supported.

In order to create leads using a script, Oracle Corporation recommends you use one of the building blocks created and maintained by the Oracle Marketing development team. Note also that these building blocks expand the offerings from business-to-business applications only to both B2B and B2C leads. The relevant Oracle Marketing building blocks include:

Script File Name	Brief Description of Script Function
AMSLDB2C.SCR	B2C Lead creation
AMSLDB2B.SCR	B2B Lead creation (replaces obsolete Oracle Scripting building block IESLEAD.SCR)
AMSLDWCA.SCR	Lead creation with customer authentication

These scripts are seeded with Oracle Marketing in current releases, and can be downloaded from the applications server from the product top directory ($AMS_TOP/scripts). Oracle Corporation suggests that users first transfer the files to a local directory, then launch Script Author, open each relevant building block script, modify as required, and deploy to the specific instance.

Enterprises using the create lead building block that upgrade to Oracle Applications release 11.5.10 or later, or Interaction Center Family Pack R or later, must modify their scripts accordingly.

References

- For a list of the Oracle Marketing building block scripts, see Integrating with Oracle Marketing.

See Also

- Requirements for Logging Responsibilities Obsolete

1.5 Features

Oracle Scripting is a set of tools to facilitate the process of gathering or exchanging information for the benefit of the enterprise. Oracle Scripting is composed of four components: Script Author, the Scripting Engine, the Scripting Administration console, and the Survey Administration console.

This section includes the following topics:

- Script Author Features
- Scripting Engine Features
- Scripting Administration Console Features
- Survey Administration Console Features
- Accessibility

See Also

- New in this Release
- Modified in this Release
- Obsolete in this Release
- Architecture
- Responsibilities
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.5.1 Script Author Features

The component of Oracle Scripting that provides script developers with tools to create, modify, and deploy scripts is referred to as Script Author. Script Author is a Java applet launched from the Scripting Administration console. Using this authoring environment, trained developers build scripts to be executed using the Scripting Engine. Scripts created using the graphical tools are referred to as graphical scripts. Scripts created using the Script Wizard feature of Script Author are referred to as wizard scripts. Custom code can be associated with graphical scripts globally, or at certain points in the script (corresponding with flow of the script at runtime). This code can execute a variety of customized or predefined commands to extend the functionality of the script.

This section includes the following topics:

- Script Author Graphical Layout
- Panels
- Groups
- Blocks
- Graphical Script Object and Branch Properties
- Panel Layout Editor
- Script Wizard
- Data Dictionary
- Script Author File Management

See Also

- Scripting Engine Features
- Scripting Administration Console Features
- Survey Administration Console Features
- Accessibility

1.5.1.1 Script Author Graphical Layout

The Script Author is a graphical layout environment. Script developers build graphical scripts by selecting *objects* from one of two tool palettes, and placing them on the work area (the canvas). Objects are connected by *branches*.

Objects include configurable objects (panels, groups and blocks) and non-configurable objects (start and termination nodes). Objects have properties which instruct the Scripting Engine how to behave when each object is reached in the flow of a script.

Branches direct the flow of the script at runtime (they tell the Scripting Engine where to go next), based on instructions received in the script. Flow in the progression of the script is controlled by branch type, and can also be designated through the evaluation of actions programmed into the script. Actions contain Script Author commands. Branch types include default, distinct, conditional, and indeterminate branches. Each enforces different business rules to control flow of the script.

Project and Syntax Tab

The left side of the Script Author development environment displays either a Project tab or a Syntax tab.

The Project tab contains a table displaying properties of the currently selected object. Above this table, you can list objects on the canvas using one of four views:

- The script view displays all objects as defined on the canvas, including their hierarchy.
- The group view displays an alphabetical listing of groups defined in the script only.
- The block view displays an alphabetical listing of blocks defined in the script only.
- The panel view displays an alphabetical listing of panel defined in the script only.

The Syntax tab provides a list of each object on the canvas that breaks Script Author syntax rules.

Debug Pane

Clicking on any listed error will cause a section of the canvas to be replaced with the Debug pane. The debug pane lists any available information about the selected problem.

Status Bar

At the very bottom of the Script Author work area, below the canvas and the Project and Syntax tabs, is a status bar. This indicates the status after a defined action such as checking syntax, saving a script, or deploying a script to the database.

Hierarchical Graphs

Scripts consist of one or more graphs (collections of Script Author objects). Each graph automatically includes a *start node* which indicates where runtime flow begins. No properties can be associated with a start node (it is a non-configurable object). A start node is represented in a graphical script by an equilateral triangle.

Each graph contains a termination node, represented in a graphical script by a red octagon. As the shape suggests, it represents the end of processing. Typically, each graph contains at least one termination node and may contain any number (you can include separate termination nodes for any specific flow on a graph).

Processing of a script in any graph begins with the start node on the root graph, and proceeds to evaluate each configurable object (panels, groups, and blocks) in the chosen path for that script session. The script session at runtime ends when the termination node on the root graph is reached.

A script can contain any number of graphs. The graph shown when you first create or open a script is referred to as the *root graph*. Any new graph (subordinate in hierarchy to the root graph), created at any level of a script, is known as a *child graph*. The graph from which a sub-graph is created is known as the *parent graph*. The root graph has no parent.

New graphs are created using group or block objects. After creating a group or block object in a graphical script, you must explicitly select it and navigate down into it to view the child graph. Any graph can contain any number of other Script Author objects (including any number of additional child graphs, if required).

When an object in the flow of the script contains a child graph in the form of a group or block, any objects included in that child graph are evaluated before the flow on the root graph continues to be evaluated. A child graph at any level is exited at runtime when the termination node on that graph is reached in the flow of the script. Processing (flow of the script) then returns to the next object in the parent graph.

Certain syntactical rules apply to every graph. All objects in a script must be connected with branches. At minimum, each graph must contain a start node and a termination node, connected by a default branch. When you include a group or block in a script, its child graph must meet the same syntactical rules as the root graph.

See Also

- Panels
- Groups
- Blocks
- Graphical Script Object and Branch Properties
- Panel Layout Editor
- Script Wizard
- Data Dictionary
- Script Author File Management

1.5.1.2 Panels

A panel is the primary object of a script, and is the most configurable object. A panel is represented in a graphical script by a blue sphere. The panel contains the information that is displayed by the Scripting Engine at runtime (in the panel presentation area, for the Scripting Engine agent interface, or in the browser window, for the Scripting Engine Web interface). At runtime, each panel displays a button at the bottom of the panel. When the end user clicks the button, this registers the end user’s responses to the questions in the panel, and progresses the script to the next panel.

Panels are created and edited using the Script Wizard or graphical tools. Regardless of which method was used to create it, the panel is the only Script Author object to display at runtime in both interfaces. (Groups, blocks and branches simply contain information for processing a script). Each panel must contain at least one question, and may contain as many as needed.

Each question in a panel displays a corresponding question user interface control (button, checkbox, text area, drop-down, and so forth) with which the script end user enters a response at runtime.

Question UI Control Differences

The most obvious difference between panels created with the Script Wizard and in a graphical script is the limitation of question user interface types. The table below compares the question UI types available to graphical scripts versus those accessible in the Script Wizard. Minor differences in control names are indicated based on the labels provided by the user interface for each script creation method.

Graphical Script	Wizard Script	End User Action Required
Text Field	Text	Text entry
Text Area	Text Area	Text entry
Radio Button	Radio Button	Select only one (required)
Check Box	[None]	Boolean (null or selected)
Button	[None]	Click (required)
Drop Down	Dropdown	Select only one (required)
Password	Password	Text entry
Checkbox Group	Checkbox Group	Select zero, one, or many
Multi-Select List Box	Multiselect List	Select zero, one, or many

Panel Properties Differ Based on Creation Method

Panels in wizard or graphical scripts may contain static panel text, and one or more questions. Questions of the appropriate type that are explicitly defined in a wizard script may include question-level validation commands. In addition, panels in graphical scripts may contain graphic images, dynamic text using embedded values, and validation, database or cursor lookup, or other commands (at the panel level or the question level).

Explicit or Implicit Question Definition

If the purpose of a wizard script panel is simply to provide information to the script end user, then Script Wizard users are not required to define any questions for the panel. The Script Wizard will automatically generate a question called Continue, which displays a Continue button at runtime.

In contrast, panels created with graphical tools *must* contain one or more explicitly defined questions. If the question UI control selected for a panel is Button, the panel may not contain any other questions. A display value (the contents of the button) and a value must be provided for each button question type created, in the Lookup entry window of the question data dictionary.

Although identical in appearance to the Button question UI type, panels in graphical scripts containing one or more question UI types other than Button will automatically display a Continue button at script runtime. This button is not required to be defined.

See Also

- Script Author Graphical Layout
- Groups
- Blocks
- Graphical Script Object and Branch Properties
- Panel Layout Editor
- Script Wizard
- Data Dictionary
- Script Author File Management

1.5.1.3 Groups

A group is a configurable object represented in a graphical script by a green cube clustered with three small colored spheres. A group represents a sub-graph in a graphical script. Unique properties of a group include *shortcuts* and *permissions*.

Groups are processing units only. Any commands associated with the group are executed when reached in the flow of the script. If a group contains panels, those panels display at runtime when reached in the flow of a script transaction. However, no aspect of a group is displayed at runtime.

Each group can contain a single shortcut, which is a name property. Shortcuts can be used at runtime to route the flow of a script to the group containing the specific shortcut. This requires use of the jumpToShortcut API.

Permissions are not implemented and should not be used. No support is available for this feature.

The primary purpose for a group is to provide organization for a script by grouping related functionality. It provides the ability to invoke its functionality (like a callable subroutine) from any point in the script using a shortcut.

See Also

- Script Author Graphical Layout
- Panels

- Blocks
- Graphical Script Object and Branch Properties
- Panel Layout Editor
- Script Wizard
- Data Dictionary
- Script Author File Management

1.5.1.4 Blocks

A block is a configurable object represented in a graphical script by a single green cube. A block represents a sub-graph in a graphical script. Unique properties of a block include its type and its object dictionary.

Blocks are processing units only. Any commands associated with the block are executed when reached in the flow of the script. If a block contains panels, those panels display at runtime when reached in the flow of a script transaction. However, no aspect of a block is displayed at runtime.

Each block can be designated by type to perform a database action (insert, query/select, or update), or to serve as a clearly designated container for one or more APIs.

For database functions, the object dictionary is used to determine connection information, join conditions, and query data such as constraints and columns.

See Also

- Script Author Graphical Layout
- Panels
- Groups
- Graphical Script Object and Branch Properties
- Panel Layout Editor
- Script Wizard
- Data Dictionary
- Script Author File Management

1.5.1.5 Graphical Script Object and Branch Properties

The two non-configurable objects in every graph of a graphical script are the start node and the termination node. These have no names or configurable properties associated with them.

Each configurable object and every branch created using Script Author graphical tools has properties associated with it. These include a name property and a comments property. The comments property is blank by default. In each graphical script, every configurable object and branch placed on a graph is automatically assigned a unique default name. Script developers should replace default names for objects with meaningful names. It is also recommended to provide meaningful names to distinct, conditional, or indeterminate branches to provide an indication of each branch's function. This is not required for default branches; names assigned to default branches are not displayed in Script Author.

The script itself is considered a configurable object. Thus, it too has a name and comment property. These are referred to as global script properties. The global script name is the name under which the script is stored in the database when deployed from Script Author. This must be distinguished from the name assigned to the script file, which may differ.

When saving a graphical script to the filing system, provide the file name with a file extension of .SCRIPT or .SCR for ease of use. When opening a graphical script, Script Author uses a file type filter which, by default, only recognizes these two file extensions.

You can also open a script file with no file extension, or with a different extension than the defaults. To do so, you must change the file type filter selection to **All Files (*.*)** in the ScriptChooser window.

Object properties for every object and branch can be accessed in one of two ways:

1. By double-clicking on the object.
2. By right-clicking on the object and selecting **Edit Blob Properties** from the pop-up menu. This option is not available on the Macintosh platform.

Additionally, properties can be assigned to the global script by selecting **File > Script Properties**.

In order to be syntactically correct, all objects on a particular graph must have properties associated with them, and be connected with appropriate branching from the start node to the termination node. Panels must contain at least one *question*. Groups and blocks must contain proper termination of their child graphs.

See Also

- Script Author Graphical Layout
- Panels
- Groups
- Blocks
- Panel Layout Editor
- Script Wizard
- Data Dictionary
- Script Author File Management

1.5.1.6 Panel Layout Editor

The panel layout editor feature of the Script Author allows script developers to quickly define simple HTML layouts for a panel within a graphical script without writing HTML code. This simple graphic interface provides users with the ability to:

- Enter and format simple panel content
- Manipulate typeface, font size and color
- Create bulleted and ordered lists
- Import existing HTML or export panel content as HTML
- Create hypertext links
- Insert embedded values
- Import GIF or JPG images

> **Note:** When an image is imported using the panel layout editor, the HTML for the panel must be exported from the panel in a graphical script, and modified to reference the image in a path accessible to the Oracle Applications Web server at runtime. The modified code is then re-imported into the panel, so the image is available to all script end users at runtime.

In addition, the panel layout editor produces syntactically correct HTML content for Scripting-specific features such as embedded values without having to learn custom Oracle Scripting HTML tags and syntax.

Questions (or question UI controls), images, and text can be interspersed and formatted as desired.

The panel layout editor includes one-click formatting of text into two distinct styles. *Instructional text* indicates to the script end user specific instructions. *Spoken text* indicates text an agent would speak (using the agent interface) or that a customer or survey respondent would read for communication of primary information for that panel.

The panel layout editor is not intended to be used for complex layout, creation of use of HTML tables, or the use of cascading style sheets, which are not supported with Oracle Scripting. To create complex layout, or to use features of HTML such as tables, it is expected that script developers will use full-function third-party HTML editors (or code HTML by hand) and simply import the HTML content into Script Author using the panel layout editor.

See Also

- Script Author Graphical Layout
- Panels
- Groups
- Blocks
- Graphical Script Object and Branch Properties
- Script Wizard
- Data Dictionary
- Script Author File Management

1.5.1.7 Script Wizard

As of Oracle Scripting release 11.5.9 or later, or Interaction Center Family Pack Q or later, Script Author includes a Script Wizard. Using this feature, script developers can quickly and easily create simple scripts or surveys by providing script information in a series of windows known as a wizard.

Scripts created with the Script Wizard are called wizard scripts. All features of wizard scripts are compatible with those of graphical scripts, and wizard scripts can be graphed or converted to a copy of a graphical script for viewing or modification using standard graphical tools. At this time, there is no backward compatibility, nor can graphical scripts be converted to wizard scripts.

The Script Wizard makes script development more accessible to non-technical users, although an understanding of business rules and business process flow is still absolutely essential to developing successful scripts using this feature. Since wizard scripts are created from Script Author, access to the Scripting Administrator responsibility is also required.

Wizard scripts can be created, edited, saved to the database without deployment, or deployed to the database. Unlike graphical scripts, wizard scripts cannot be saved locally in a computer’s file system. Wizard scripts can only be listed in a user interface from within the Script Wizard.

Using a SQL tool such as SQL*Plus, you can list properties for saved or deployed scripts. The metadata for saved scripts are stored in table IES_DEV_SCRIPTS, and the metadata for deployed scripts are stored in table IES_DEPLOYED_SCRIPTS. To differentiate between script types, wizard scripts are assigned a SCRIPT_CATEGORY value of 0, whereas graphical scripts are assigned a SCRIPT_CATEGORY value of 1.

Highlights of wizard scripts include the following:

- A wizard script cannot be saved or deployed until it is syntactically correct. When you complete a wizard script, you can save it and continue working in the wizard, you can save and exit the script, or you can save, deploy, and exit. The product of progressing through the Script Wizard and saving your work is always a valid, deployable wizard script.
- From the Script Wizard, existing wizard scripts can be edited, copied, deleted, graphed for use in the graphical Script Author tools, or deployed.
- Each wizard script automatically contains a valid Disconnect button. This is only visible in the Scripting Engine agent interface.
- Each wizard script automatically contains a valid Suspend button, unless the Suspend feature is disabled through other means. This is only visible in the Scripting Engine agent interface.
- When defining question detail, wizard script developers can include default answers to questions by providing a value in the Default Value field. These can be accepted or changed at runtime.
- When defining question detail, using a few clicks, wizard script developers can include question validation for responses provided at runtime. You can check that the response provided is either (a) an integer, (b) an integer within a

designated start and end range, (c) a valid date, or (d) a date that is not in the past (e.g, the date is equal to SYSDATE or in the future).

- Flow of a script can be controlled in part by providing an exit panel sequence. This method allows panels meant to contain information only (in contrast to panels designated to collect information) to automatically be provided with a Continue button without requiring the Script Wizard user to specify a panel answer.

Wizard scripts can contain panels, default branches, distinct branches, start and termination nodes, and a Disconnect group. Panels contain questions which may have predefined answer choices (lookup values).

For advanced wizard script features, any question may contain a default answer. Additionally, questions requiring text input may have the aforementioned question validation. Corresponding features in graphical scripts are provided, respectively, by defining constant commands or Java commands referencing custom or best practice Java methods. These are defined for graphical scripts in the data dictionary for the particular question.

See Also

- Script Author Graphical Layout
- Panels
- Groups
- Blocks
- Graphical Script Object and Branch Properties
- Panel Layout Editor
- Data Dictionary
- Script Author File Management

1.5.1.8 Data Dictionary

Each question in a panel has a data dictionary associated with it. For question UI types that support multiple answer choices (radio buttons, drop-down lists, checkbox groups, and multi-select list boxes), these answer choices are determined by the contents of the question's data dictionary.

Data Dictionary and Wizard Scripts

The following are methods by which the data dictionary for a question is populated when you create a wizard script:

- Providing answer choices using the Answer Manager in the Script Wizard populates a question’s data dictionary.
- Changing any defaults in the Script Wizard Define Question Detail causes question validation to be associated with the specified question's data dictionary.

Data Dictionary and Graphical Scripts

In graphical scripts, you can access the data dictionary for any question. This window has three tabs: General, Table info, and Lookups.

In the General tab, you can:

- Deselect the default Boolean Collectible property, ensuring that answer collection does not record the end user's response to this question.
- Associate question validation by defining a Script Author Java command.
- Associate a Script Author default command with this question.
- Identify a data source for this question.

In the Table info tab, you can identify table, column, format mast and data type information for a question.

In the Lookups tab, you can:

- Associate a specific table for a table lookup.
- Specify a cursor lookup, which will populate the answer choices with the current value of the scripting cursor at that point in the script at runtime.

  The cursor is populated by any SQL select statement, and is flushed each time a new query is executed.
- Associate a command lookup for this question, specifying a Script Author command.
- Specify hard-coded answer choices for this question.

Accessing the Data Dictionary in a Graphical Script

Access the data dictionary as follows:

1. In the Panel tree, select **Answers**.

2. In the Answers pane, click **Add**.

   The Answer Entry Dialog window appears.

3. Click **Edit Data Dictionary**.

The Edit Data Dictionary window appears.

See Also

- Script Author Graphical Layout
- Panels
- Groups
- Blocks
- Graphical Script Object and Branch Properties
- Panel Layout Editor
- Script Wizard
- Script Author File Management

1.5.1.9 Script Author File Management

As of Oracle Scripting release 11.5.9 or later, or Interaction Center Family Pack P or later, Script Author is a Java applet launched from a valid Oracle Applications session. In previous releases, Script Author was a two-tiered Windows-only client application that required separate database login and authentication at the APPS user level. With the introduction of the Java applet version of Script Author, these restrictions are eliminated.

Script Author Editing Forward and Backward Compatibility

Scripts saved using client versions of Script Author (versions 1.6.1.02 and earlier) *can* be opened in Java applet (1.6.2 and later) versions of Script Author. However, scripts created or modified and saved with the Java applet version (1.6.2 or later) version of the author are *not* backward compatible, and cannot be opened in previous Script Author versions. A warning message will display for users attempting to save an older script in the applet version.

Note that older scripts can be imported as groups without change to the original older script file. This method is recommended when older files must retain their editability in client application versions of Script Author.

Runtime Compatibility

Oracle Corporation recommends that all scripts deployed to the database from the Script Author standalone application (Script Author versions prior to 1.6.2) be redeployed using the Script Author Java applet, accessed from the Scripting Administration console.

Single Script Editing

The Script Author application can only open a single script at a given time. Opening a second script causes any open scripts to close, prompting the user to save any changes made to the file. This restriction applies to client application and Java applet versions of Script Author.

- To modify two scripts simultaneously, you must open two separate instances of Script Author (from separate Oracle Applications sessions, using two different compatible Web browsers). Objects or data will not be able to be conveyed using the clipboard when working in this fashion, although import and export capabilities are available.
- To deploy the same script to multiple environments (for example, to move a fully tested script from a development to a production environment), you must first save a copy of the script from the original environment to a local or shared file system. Then you must log out, log into the target environment, launch the Script Author applet, open the script, and deploy it to the environment.

Database Connection Information

The Script Author Java applet is accessed from a valid Oracle Applications session, and is automatically aware of the database location of scripts for that environment. Thus, opening Script Author seeded or stored commands in the command library, opening scripts from the database, and saving deployed scripts to the database all occur without the need to establish database connections.

To open deployed scripts from or deploy scripts to a different database instance, you must log into the Oracle Applications instance for that environment and launch the Script Author applet from the corresponding Scripting Administration console.

Connecting to the database from the Script Author client application (prior to Interaction Center Family Pack P or Oracle Scripting release 11.5.9) requires specification of the database host machine, port, and SID, plus use of the APPS level user name and password.

See Also

- Script Author Graphical Layout

- Panels
- Groups
- Blocks
- Graphical Script Object and Branch Properties
- Panel Layout Editor
- Script Wizard
- Data Dictionary

1.5.2 Scripting Engine Features

The Scripting Engine is a set of base Java classes that process a Script Author script in runtime. The script provides a customized set of pre-programmed actions to take based in part on responses provided by the end user. At runtime, the Scripting engine interprets:

- Instructions in the script,
- The end user's responses to questions and answers, and
- Custom code referenced in the script.

Interpretation of Custom Code

Custom code is referenced in the script by associating commands to objects in the script (or to the script itself) before it is deployed to the database from Script Author. Based on the type of command, the code may be stored in the meta-data of the script itself (e.g., constant commands or PL/SQL commands), in the Scripting session (Blackboard commands), in custom code deployed to the applications server (Java commands), or in the database (commands referencing stored PL/SQL packages).

Two Runtime Interfaces

There are two runtime interfaces, both of which receive information from the base Java methods that together comprise the Scripting Engine.

The first is the *Scripting Engine interaction center agent user interface* (agent interface). This is a Java-based UI launched from an Oracle Applications session when an agent starts a script. Typically, scripts are executed from within one of three Oracle business applications: Oracle TeleSales, Oracle Collections, or the Customer Support module of Oracle TeleService.

Note: Technically, agents or software implementors can also launch a script in "standalone" mode from within an Oracle Applications session. However, this method of launching a script is not supported, and is suggested for use primarily for script testing.

In current releases, scripts viewed in this model present as Java components wrapped in an Oracle form. In an Oracle Applications session, Oracle Forms runs as an applet on the client desktop. Each component, such as the Scripting Engine agent interface, executes as Java code subject to the rules under which Java applets must run. HTML interpretation and rendering is provided by Java classes in the SWING library of JavaSoft's Java Development Kit (JDK) 1.1.8.

The second interface is the *Scripting Engine survey runtime user interface* (Web interface), which is essentially HTML running JavaServer Pages (JSP) in an Oracle Applications 11*i*-certified Web browser. This runtime interface is intended for two models: for users of self-service Web applications to run a script in a Web browser, or for individuals to view a script to participate in a survey. Both sets of users run the script using their Web browser to access a particular URL. The URL specifies a set of parameters that identify an enterprise's Oracle Applications-specific Apache Web server, invoke a JSP page for survey operations, specify a deployment identification code (or dID), and (for targeted or list-based survey deployments), a unique respondent identification code or rID). When taking a survey by visiting a survey URL sent in an invitation or e-mail reminder, or when coming across the survey URL on an enterprise’s Web site, the survey respondent logs into an Oracle Applications session established for a guest user. This user has no privileges other than the ability to participate in a survey. For individuals taking a survey from an integrated self-service Web application such as Oracle iSupport, the Oracle Applications login information for the current valid Oracle Applications session is used, and the individual selects a link from iSupport to execute the script in a Web browser.

In this model, the Scripting Engine interprets the respondent's responses, the script, and custom code, passing metadata and instructions to the Web browser to be interpreted on the respondent's client computer. In addition to a script, survey execution of scripts requires a survey campaign to be established and deployed, and requires the presence of additional JSP resources (to provide a header and footer section for each survey page, an error page to display if required, and a final page after completion of the survey).

What Information Is Displayed?

Of the objects created in Script Author, only the contents of a panel display in runtime in the Scripting Engine (the other objects control processing). Each panel contains, at minimum, one answer definition, which displays the appropriate answer control at runtime. Panels may include any number of answer definitions. At the script developer's discretion, the WYSIWYG (What You See Is What You Get) interface allows customized display of formatted text and images to appear in a panel, interspersed (if desired) between answer controls. At runtime, each panel displays a button at the bottom of the panel. When the end user clicks the button, this registers the end user's responses to any other answer controls in that panel (radio buttons, checkboxes, drop-down or multi-select lists, or text in a text field, text area, or password field), and progresses the script to the next panel.

See Also

- Script Author Features
- Scripting Administration Console Features
- Survey Administration Console Features
- Accessibility

1.5.3 Scripting Administration Console Features

The Scripting Administration console is an HTML administration user interface for script developers and administrators. As of this release, this component relies upon the Oracle CRM Technology Foundation (JTT) technology stack.

The Scripting Administration console has three primary functions: to launch the Script Author Java applet, to provide administration of Oracle Scripting files, and to provide access to agent application reports.

This section includes the following topics:

- Script Author Applet
- Oracle Scripting File Administration
- Oracle Scripting Agent Interface Reports

See Also

- Script Author Features
- Scripting Engine Features

- Scripting Administration Console Features
- Accessibility

1.5.3.1 Script Author Applet

From the Home tab, logged-in users of the Scripting Administration console can launch Script Author as a Java applet. No additional login information is required to launch the applet, connect to the database, access the command library, or deploy scripts.

1.5.3.2 Oracle Scripting File Administration

From the Administration tab, administrators can administer deployed scripts and Java archive files used by Oracle Scripting. Specifically, you can perform the following:

- View and delete deployed scripts.
- View, upload, update, and remove custom Java archives in support of Scripting operations.
- Set and remove the Global property for uploaded JAR files. The global property enables a specified set of code to automatically be loaded and available to all active scripts.
- Map Java archives to specified scripts to enable that code to be explicitly loaded and available to the script.

This console is accessed by logging into Oracle HTML-based applications using a user account with the Scripting Administrator responsibility.

1.5.3.3 Oracle Scripting Agent Interface Reports

From the Reports tab, you can generate and view panel footprint reports for a specified script. This report is instrumental in tuning a script, and indicates for each session or interaction of a script which panels were visited and the duration of the visit (in milliseconds). This is currently the only agent interface report for Oracle Scripting.

To generate meaningful information, the designated script must collect footprinting and answer collection information. These are global attributes of a script. At minimum, the Answer Collection property must be selected (this will also result in the collection of footprinting data).

Footprinting and Answer Collection

Answer collection is the recording of end user responses ("answers") to all question UI controls ("questions") that are marked in the script as collectable. Answers are collected for each transaction or session of the script running in the Scripting Engine, in either interface. If enabled (at the script level), answer collection data is collected in table IES_QUESTION_DATA.

Footprinting is the recording in the database of which panels in a script transaction were visited, and the duration of time in milliseconds before the next panel is requested. Footprint data is stored in table IES_PANEL_DATA.

For each script, you can designate the collection of footprint data by enabling the Footprinting option as a global script property. For each new script created, this option is selected by default. These options are also enabled automatically for any script created using the Script Wizard.

Additionally, regardless of whether the Footprinting option is enabled, footprinting data is now also automatically saved to the database when the Answer Collection option is enabled. This change (introduced in Interaction Center FP-Q and later or release 11.5.9 or later) improves the quality of data saved from a script transaction for reporting purposes. This ensures a link between each response provided at runtime, and the specific panel instance from which that response was provided. Accordingly, an additional column (PANEL_DATA_ID) is contained in the answer collection table, IES_QUESTION_DATA. This column contains the foreign key reference to the footprinting table, IES_PANEL_DATA.

If the Footprinting option is selected but the Answer Collection option is not, only footprinting data for each session or transaction of that script will be saved to footprinting tables. Obviously, if neither option is selected for a specific script, no footprinting or answer collection data is saved.

If neither footprinting nor answer collection are enabled for a script:

- No data will be available from which to view individual responses from the Responses tab of the Survey Administration console.
- No data will display in the panel footprint report executed from the Reports tab of the Scripting Administration console.

If answer collection is disabled for a script, but footprinting is enabled:

- No data will be available from which to view individual responses from the Responses tab of the Survey Administration console.
- Footprinting data will still be recorded in the IES_PANEL_DATA table if enabled for the script.

References

For more information, see Footprinting and Answer Collection

1.5.4 Survey Administration Console Features

The survey component of Scripting was previously known as iSurvey.

The Survey Administration console is an HTML administration user interface for survey campaign administrators and managers. This console integrates functionality from various Oracle applications for the benefit of administering and monitoring survey campaigns. Oracle Scripting, Oracle One-to-One Fulfillment, and Oracle Marketing functionality are all accessible from this single user interface. The Survey Administration console has the following primary functions:

1. To administer survey campaigns, cycles, and deployments in support of a survey campaign. This is performed from the Survey Campaigns tab of the Survey Administration console. Additionally, you can view responses received from existing survey campaigns by viewing deployment detail in the response view.
2. To administer survey resources, which are JSP files that appear as header or footer sections, error pages, or final pages for scripts executed in a Web browser. This is performed from the Survey Resources tab of the Survey Administration console.
3. To perform list management supporting targeted survey deployments. This accesses Oracle Marketing functionality. This is performed from the Audience tab of the Survey Administration console.
4. To perform management of Oracle One-to-One Fulfillment master documents, queries, and templates, which provide the ability to send through e-mail invitations and reminders for targeted survey deployments. This is performed from the Invitations tab of the Survey Administration console.

1.5.5 Accessibility

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/

Accessibility of Code Examples in Documentation

JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites.

User Interfaces

Oracle Applications user interfaces required for administration of Oracle Scripting are rendered in HTML, making them available to users of assistive technology.

Script Development

New with this release, the Script Wizard feature of the Script Author component extends script development capabilities to people who are blind or visually impaired. The Script Wizard provides an alternative to the script development tools using visual layout tools, and can be used in combination with appropriate assistive technology (AT) such as screen readers. For example, the JAWS screen reader works with Java applications. Other AT products may apply as appropriate.

See Also

- Script Author Features
- Scripting Engine Features
- Scripting Administration Console Features
- Survey Administration Console Features

1.6 Architecture

Oracle Scripting 11.5.6 and later features the Apache mid-tier architecture. This architecture improves scalability of scripts executed at runtime, over the caching architecture introduced in release 11.5.4, and follows the Oracle Applications philosophy of a distributed architecture using the Apache Web server as the middle tier.

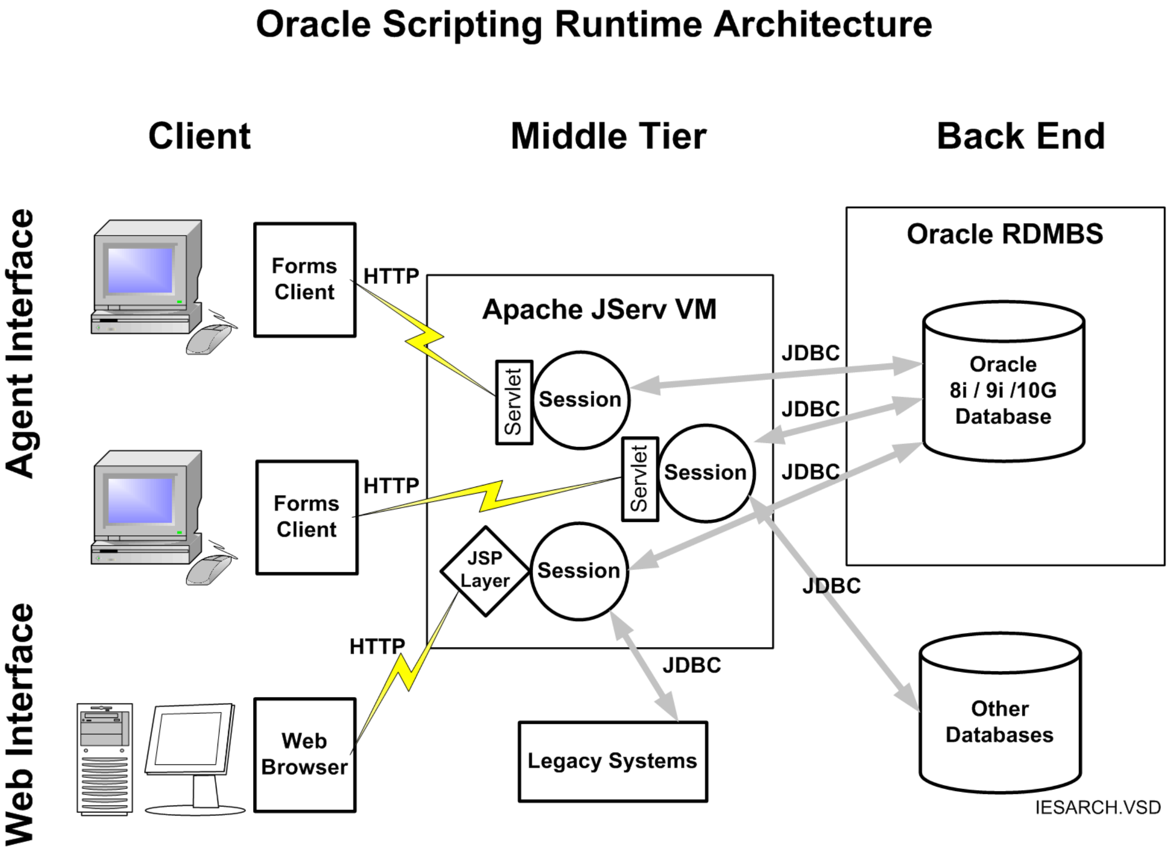
This section includes the following topics:

- Apache Mid-Tier Architecture
- Caching Architecture
- Oracle Scripting and the Session Object

See Also

- New in this Release
- Modified in this Release
- Obsolete in this Release
- Features
- Responsibilities
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.6.1 Apache Mid-Tier Architecture



Using the Apache mid-tier architecture, business logic is executed on the enterprise Web server. The Apache JServ provides the Java Virtual Machine (JVM), executing the script at runtime as a servlet. Scripts are executed by the Scripting Engine using two different interfaces.

All communications with the client (for either Scripting Engine interface) are through the hypertext transfer protocol. Scripts can be executed over a firewall with appropriately configured secure HTTP (HTTPS). From the middle tier, the session itself communicates with the data tier (the Oracle Applications database, as well as any other database that may be used) using Java database connectivity (JDBC).

Agent Interface

Customer service or interaction center agents typically use Oracle Scripting through a Forms-based business application such as Oracle TeleSales, Oracle Collections, or Oracle TeleService. Regardless of how agents are using the Scripting Engine agent interface, when an agent requests a script (or launches an application that automatically requests a script), a session is started using HTTP between the Forms client (the agent workstation) and the Scripting servlet on the Apache server. Back-end communication is handled with JDBC.

Web Interface

Individuals using the Web interface of the Scripting Engine are executing a script either as a survey, or as a Web script from self-service applications such as Oracle iSupport. The script end user accesses a specified URL from a supported Web browser, sending the message through HTTP to launch a script associated with a particular set of requirements. These requirements are defined in the survey campaign, and include cycle and deployment parameters. From a Web interface perspective, the Scripting Engine is a set of Java classes on the middle tier that receives requests from the JSP layer (including the message to launch a script), and processes them. All business logic as determined by the survey questionnaire script, custom code or database calls, and respondent actions are evaluated by the Scripting Engine, which passes processing information to the JSP layer to interpret, which in turn sends information to the Web browser to display.

Configuration File Customization No Longer Required

Using the latest functionality supporting this architecture, custom Java code can be deployed to the Oracle Applications database using the Scripting Administration console. This precludes the need to manually deploy custom code to the applications server, and eliminates the need to specify the custom Java code's class path in a configuration file. Code loaded from the Scripting Administration console can be made available globally to all scripts, or can be mapped to specific scripts. Either way, the Scripting Engine obtains the appropriate code from the database dynamically using JDBC so the code is available to the script at runtime.

Using the Apache mid-tier architecture, appropriately compiled and packaged custom Java code that was manually deployed to the applications server should now be deployed to the database using the Scripting Administration console and mapped accordingly. No customization or updates to the JSERV.PROPERTIES file will be required thereafter.

Additionally, individuals upgrading to the Apache mid-tier architecture from the caching architecture will no longer need to customize or update their

APPSWEB.CFG files to reference custom Java. All Java archive code loaded to the database using the Scripting Administration console will be made available to the Scripting Engine based on its script mapping or global status.

See Also

- Caching Architecture
- Oracle Scripting and the Session Object

1.6.2 Caching Architecture

Oracle Corporation also provides limited support for the caching architecture. The caching architecture is not supported for new users of Oracle Scripting. Only users upgrading from previous releases of Oracle Scripting and who are already using the caching architecture will continue to be supported in this architecture by Oracle Support Services. Oracle Corporation recommends that caching architecture users upgrade to the Apache mid-tier architecture.

Using the caching architecture, business logic is executed using the Java Virtual Machine (JVM) provided on the client by Oracle JInitiator. Scalability over the previous architecture is much improved, at the expense of executing the JVM on client workstations (fat client model). This may necessitate increased processing power on agent workstations and is inconsistent with the Oracle Applications release 11*i* philosophy. Use of the caching architecture also requires customization of APPSWEB.CFG for specification of the class path for any custom Java to be used. Corresponding Java archives must then be deployed to the applications server in the same class path referenced in the configuration file, typically in a directory such as <JAVA_TOP>/ies_custom. For more information, see Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File".

See Also

- Apache Mid-Tier Architecture
- Oracle Scripting and the Session Object

1.6.3 Oracle Scripting and the Session Object

Oracle Scripting is a stateful application. The application incorporates the concept of the script "transaction," and monitors the Scripting Engine user's state throughout the session.

A script transaction begins when a script is launched in the Scripting Engine (agent or Web interface), and completes when the script runs to completion. Completion is indicated by reaching a termination node on the root graph of a graphical script, or by reaching a path in a panel created in the Script Wizard for which the next destination is "End script." In the agent interface, clicking the Disconnect button results in directing the session to the termination node on the root graph; clicking the Suspend button, for the purposes of this discussion, can be considered the temporary completion of a script.

All script transactions require a valid Oracle Applications session.

The Session Object

Session is the representation of state for a particular user. At script runtime, a Scripting Engine user is connected with the scripting servlet. For each individual executing a script, there is a session object running in the Java Virtual Machine (JVM) of the Apache JServ (for Apache mid-tier architecture implementations) or in the JVM of Oracle JInitiator (for caching architecture implementations).

Agents using the Scripting Engine agent interface maintain the session for the duration of their Oracle Applications session. Within one session, users can run one or more subsequent script transactions. Only one script can be executed at a time.

Monitoring State

Oracle Scripting is a stateful application, and relies strongly on the tracking of the user's state through the session object. Some examples of the session's function include:

- Loads the script from the applications database
- Processes branching logic and answers provided by the client
- Executes custom code

By tracking state for each user, Oracle Scripting tracks where in the flow of a script a user is, and how much time (in milliseconds) is spent in each panel. This is known as footprinting. Oracle Scripting also provides for data collection associated with the transaction footprint, and provides the functionality of the Scripting blackboard.

Number of Transactions Per Session May Differ By Interface

Users of the Scripting Engine agent interface (typically interaction center agents) tend to run multiple script transactions in one Oracle Applications session. The session is retained as long as the agent is logged into Oracle Applications.

Each Oracle Scripting transaction executed in the Scripting Engine agent interface is typically associated with one customer (whether the party is identified or not). The Oracle Scripting end user (usually an interaction center agent) initiates an Oracle Scripting transaction, typically in response to reaching a customer on the telephone (inbound or outbound calls). The agent then uses the script to control the flow of the conversation and capture information required for the intended business purpose.

Oracle Corporation recommends Oracle Scripting in the agent interface to be used in combination with an integrated business application. Agents either specifically launch a script, or a designated script is launched by the integrated business application, to address the needs of a customer. The agent navigates through the script as required, and exits the script when the script business purpose is complete, ending the transaction. Each invocation of a script is a separate Oracle Scripting transaction.

For scripts executed in the Scripting Engine Web interface, the number of Oracle Scripting transactions per Oracle Applications session can differ.

In the deprecated JTT technology stack, when using hosted Web scripts or surveys from a self-service Web application, multiple transactions can be executed. (These surveys the JSP template IESSVYMENUBASED.JSP in the survey URL).

This is also true Web scripts or surveys executed in the OA Framework technology stack, when executing scripts from an authenticated Oracle Applications session.

In either of these models, a self-service Web application user logs into the appropriate application, beginning an Oracle Applications session. When a survey URL or link is accessed, the authentication used for the calling Web application is passed to the database, and the user is authenticated to begin an Oracle Scripting transaction, resulting in the appearance in the user's Web browser of the first panel in the script. The survey or Web script user answers the questions or otherwise navigates through the script. When the question in the final panel of the script has been answered, the final page survey resource is displayed, and the script transaction is completed. The user cannot navigate backwards in the Web browser after reaching the final page, because the transaction has ended. However, the self-service Web application, and the Oracle Applications session, are still valid. If supported by the business rules of the governing survey campaign, a second Web script can be launched from the Web application. Each invocation of a script is a separate transaction.

In contrast, only a single transaction can be executed when using a guest user, regardless of technology stack. Thus, all sessions in the JTT technology stack with a survey URL including the JSP template IESSVYMAIN.JSP include only a single Oracle Scripting transaction. Additionally, any Web script or survey executed in the

OA framework technology stack that is not hosted by a self-service Web application can include only a single transaction.

When the Web script user or survey respondent navigates to a designated survey URL in a Web browser, an Oracle Applications session starts, using the applications Guest User account. The Oracle Scripting transaction begins, and the first panel of the script appears in the Web browser. The user answers the questions or otherwise navigates through the script. When the question in the final panel of the script has been answered, the final page survey resource is displayed, and the script transaction is completed. At the completion of the single transaction, the session is invalidated. The user cannot navigate backwards in the Web browser after reaching the final page, because the transaction has ended. Thus, each survey questionnaire session will include only a single invocation of a script (a single Oracle Scripting transaction).

For more information, see Survey URL and Oracle Self-Service Web Applications.

Post-Transaction Processing

When a script transaction ends, the Scripting Engine must execute several tasks. These include:

- Update record in IES_TRANSACTIONS table with script end time
- Write answers collected during transaction into appropriate table (IES_QUESTION_DATA), assuming answer collection is enabled for the script
- Write footprinting data into appropriate table (IES_PANEL_DATA), assuming answer collection or footprinting is enabled for the script

End-of-script processing occurs in separate thread using a different database connection. This allows the next script transaction to be started (in the existing thread) before the data from the prior session is completely written to the database. The data from the previous transaction is concurrently written to the database using the newly instantiated thread. The net end user effect is the ability of agent interface users to start a new Scripting transaction much sooner.

References

- Survey URL
- Oracle Self-Service Web Applications.

See Also

- Apache Mid-Tier Architecture

- Caching Architecture

1.7 Responsibilities

Various responsibilities may be required for using Oracle Scripting, based on which component you are using and which integrated applications you want to use with Scripting. Additionally, many more responsibilities are required for implementing this product. For more detailed information on the various users required to implement and use Oracle Scripting, see Implementation Responsibility Matrix.

This section includes the following topics:

- Scripting Responsibilities
- Survey Responsibilities

See Also

- New in this Release
- Modified in this Release
- Obsolete in this Release
- Features
- Architecture
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.7.1 Scripting Responsibilities

User Type	Responsibility	Function	For Component(s)
Agent	Scripting User Scripting Agent	Launch scripts in "standalone mode" from Oracle Forms, typically to test deployed scripts.	Scripting Engine for agent interface
Agent	Customer Support	For agents accessing Oracle Scripting from within the integrated Oracle TeleService application.	Scripting Engine for agent application

User Type	Responsibility	Function	For Component(s)
Agent	TeleSales Agent	For agents accessing Oracle Scripting from within the integrated Oracle TeleSales application.	Scripting Engine for agent application
Agent	Collections Agent	For agents accessing Oracle Scripting from within the integrated Oracle Collections application.	Scripting Engine for agent application
Administrator	Scripting Administrator	Access the Scripting Administration console. This is required to: ■ Launch Script Author as a Java applet, accessible from the Home tab. ■ Administer Oracle Scripting files, including viewing and removing deployed scripts, viewing, uploading, updating or removing custom Java archive files. ■ Generate, view and analyze the Panel Footprint agent interface report.	Scripting Administration console

See Also

- Survey Responsibilities

1.7.2 Survey Responsibilities

User Type	Responsibility	Function	For Component(s)
Administrator	Survey Administrator	Access the Survey Administration console. This is required to: ■ Define survey resources (header sections, footer sections, final pages, error pages in JSP format) ■ Create survey campaigns and cycles. ■ Define or activate survey deployments ■ View specific responses provided by survey respondents ■ Administer list information for targeted deployments ■ Administer invitation or reminder master documents, queries and templates for targeted deployments	Survey Administration console

User Type	Responsibility	Function	For Component(s)
Administrator	iSurvey User	Schedule or run concurrent programs.	Survey Administration console
Administrator	One-to-One Fulfillment Administrator	▪ Access the Fulfillment Administration console. ▪ Set up or change fulfillment groups ▪ Set up or change mail servers associated with fulfillment groups ▪ View failed fulfillment requests after concurrent programs succeed in passing requests to the fulfillment server.	Survey Administration console These fulfillment administration tasks are required for implementations using targeted survey deployments
Administrator	Oracle Marketing Super User	▪ Access the Marketing Administration console. ▪ Establish campaigns, assign agents to campaigns, assign scripts to agents or scripts to campaigns ▪ Create, import, modify lists	Scripting Engine for agent application If Oracle TeleSales or Oracle Collections agents must launch a specific script, the script must be associated with a campaign or campaign schedule. Survey For targeted survey deployments, lists must be created and administered.

See Also

- Scripting Responsibilities

1.8 Script Author Concepts

This section includes the following topics:

- Scripts
- Global Script Properties
- Oracle Scripting Users
- Custom Code

- Using Custom Java

See Also

- New in this Release
- Modified in this Release
- Obsolete in this Release
- Features
- Architecture
- Responsibilities
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.8.1 Scripts

A script is a miniature program built using Script Author by trained functional users known as script developers.

A script facilitates the flow of information based on customized business requirements. Each script enforces business rules programmed into it by script developers.

At runtime, scripts can display text, prompts, and images to its end users. The information displayed, and the order in which that information is presented, is based on:

1. A predetermined set of requirements (business rules, custom code, and specific scripted text) that have been programmed into a script.
2. Responses provided by the end user. Responses come in the form of mouse clicks or keyboard actions.

Script developers can create two types of script from Script Author. Using the standard graphical tools, script developers create graphical scripts by selecting Script Author objects from one of two tool palettes in the user interface, and placing the selected object on the graphical workspace known as the canvas. The developer then associates properties to each object. Objects are connected to each other, similar to a flow chart. Flow of the script at runtime is determined in part by the choice of branch used to connect objects. For graphical scripts, flow can also be determined dynamically for each script runtime session, using commands that evaluate data received into the script.

Decision-enabling data for graphical scripts includes previous end user responses for the current session, information received from external database tables or applications through forms commands, blackboard commands or PL/SQL commands, and events or conditions such as the time of day.

Using the Script Wizard, script developers can provide script information in a sequence of windows called a wizard. The resulting wizard script can be saved when syntactically correct, and can be deployed to the database from the Script Wizard user interface.

Decision-enabling data for wizard scripts includes previous end user responses for the current session, and validation of text responses provided if question validation was specified by the script developer.

See Also

- Global Script Properties
- Oracle Scripting Users
- Custom Code
- Using Custom Java

1.8.2 Global Script Properties

Global script properties are those properties that apply to an entire script. This is in contrast to properties of a specific object in a script such as a panel, group, block, or branch. Global script properties are assigned default values when a script is created.

The first property, Script Type, is determined when you select Graphical script or Wizard script in Script Author (**File** > **New**). This property is not modifiable.

From the Script Author visual layout, you can view and update all modifiable global script properties from the File menu (**File** > **Script Properties**) or by right-clicking on an empty portion of the canvas (**Edit** > **Edit Blob Properties**).

From the Script Wizard, you can view and update all modifiable global script properties in the first wizard window (Define Script Properties).

For more information, including a table of global properties and a description of each, see *Oracle Scripting User Guide* > Understanding Script Author > Understanding Global Script Properties.

See Also

- Scripts
- Oracle Scripting Users
- Custom Code
- Using Custom Java

1.8.3 Oracle Scripting Users

This section includes the following topics:

- Script Author Users
- Scripting Administrative Users
- Scripting Engine Users
- Survey Administrative Users

See Also

- Scripts
- Global Script Properties
- Custom Code
- Using Custom Java

1.8.3.1 Script Author Users

The Script Author development environment is the component of Oracle Scripting which provides the sole means of creating scripts for execution in any Oracle Scripting runtime Scripting Engine interface.

As of Oracle Scripting release 11.5.9 or later (or Interaction Center Family Pack P or later), Script Author is a Java applet accessed through Oracle Applications by a user with the Script Administrator responsibility. In previous versions, Script Author was a standalone Windows Java application that required separate implementation, installation, and setup. Regardless, users that access Script Author are referred to as script developers.

The Script Author provides a graphical user interface intended for the functional user with some technical knowledge.

The graphical development environment provides for reuse of defined commands and existing Scripting components. The script developer will define these commands. Hooks in the Script Author UI reference technical components (for example, Oracle Forms, custom Java methods, and PL/SQL packages stored in the database) which are primarily developed externally. These components provide sophisticated functions to accomplish the functionality most enterprises want. They must be developed by individuals certified and knowledgeable in the relevant technologies. The script developer must work with these highly technical resources to ensure the custom components are appropriately integrated into the script. The script developer must also ensure the code is appropriately loaded in the database or applications server (based on approach) and properly referenced in the script. This will ensure the code is available to the base Java classes that provide Scripting runtime functionality for the Scripting Engine agent or Web interface at runtime.

As of Oracle Scripting release 11.5.9 or later (or Interaction Center Family Pack Q or later), Script Author includes a Script Wizard component accessed through the menu or Script Author toolbar from Script Author. With this new feature, less technical users can quickly and easily create simple scripts or surveys by providing script information in a series of windows known as a wizard.

Script Wizard users have limited access to the more technical features available in the graphical development environment. This provides the opportunity to divide script development amongst, for example, business process engineers with full knowledge of the business and flow requirements of a script, after which they can turn the script over to more experienced developers to add hooks, commands, reference Java or Forms, and so forth.

Users of Script Author include campaign administrators, Java developers or database programmers, business process engineers, and experienced interaction center agents with technical aptitude. The keys to successful script development are: (1) familiarity with how Oracle Scripting captures and processes data, (2) knowledge of associated technologies (including access to experts in these technologies), and (3) adequate training and familiarity with existing documentation.

Using the standalone Script Author Java application, no responsibility was required to launch Script Author on a Windows client. For current and future releases, only the Script Author Java applet, accessed through a validated Oracle Applications session, is supported.

Since script developers can deploy, delete, and otherwise affect scripts in production, and can manipulate information in the applications or other enterprise database, Oracle Corporation strongly recommends that only trusted users be provided with the Scripting Administrator responsibility. For more information, see Creating an Administrator for Oracle Scripting.

See Also

- Scripting Administrative Users
- Scripting Engine Users
- Survey Administrative Users

1.8.3.2 Scripting Administrative Users

Users of the HTML-based Scripting Administration console include script developers (see Script Author Users), who launch the Script Author applet from the Home tab, and administer deployed scripts and custom Java archive files from the Administration tab. Interaction center campaign administrators or system administrators (as well as script developers) will also typically access this console to run panel footprint reports to help tune a script's structure, increase agent performance and reduce average talk time. To access the Scripting Administration console user interface from the CRM Home Page login (or the Single Sign-On login, if implemented), these users must have the Scripting Administrator responsibility.

See Also

- Script Author Users
- Scripting Engine Users
- Survey Administrative Users

1.8.3.3 Scripting Engine Users

End users of the Scripting Engine agent interface are trained interaction center agents ("agents") or customer service representatives. These are non-technical users who have received simple but thorough training in the Java-based Scripting Engine interface. These individuals include call center agents taking or presenting information over the telephone, as well as interaction center agents taking advantage of other media such as intranets, enterprise portals over the World-Wide Web, and so forth.

Scripting Engine agent interface users typically launch scripts from a business application such as Oracle TeleSales, Oracle Collections, or the Customer Support component of Oracle TeleService. For testing purposes, the agent interface can also be launched in "standalone" mode. Agent interface users must have access to the appropriate Oracle Applications responsibility to launch the integrated business application from which Oracle Scripting is integrated.

End users of the Scripting Engine Web interface include users of targeted (list-based) or standard (non-list-based) survey campaign deployments, or self-service Web application users. Survey respondents may have been invited to participate in a survey through an e-mail message or through navigation to a survey-enabled site. Self-service Web application users access a script or survey in a Web browser through a self-service Web application scenario such as Oracle iSupport.

To access a survey using a self-service application, the appropriate responsibility to access that application is required. No Oracle Applications responsibilities are required of the end user to execute a script as a standard or targeted survey deployment. The user simply accesses the given survey URL in any Oracle Applications 11*i*-certified Web browser.

See Also

- Script Author Users
- Scripting Administrative Users
- Survey Administrative Users

1.8.3.4 Survey Administrative Users

Users of the JSP/HTML-based survey campaign administrative console are non-technical users with access to detailed project requirements. These individuals are typically interaction center survey campaign administrators or system administrators.

To access the Survey Campaign administrative interface from Oracle Personal Homepage (PHP) login (or the Single Sign-On login, if implemented), these users must have the Survey Administrator responsibility.

See Also

- Script Author Users
- Scripting Administrative Users
- Scripting EngineUsers

1.8.4 Custom Code

Custom code is used to obtain information needed during the execution of the script, or to enforce specific business rules. Custom code supported by Oracle Scripting includes Java commands, PL/SQL commands, SQL calls to the database, Scripting blackboard commands, forms commands, and constant commands.

Interpretation of Custom Code

Custom code is referenced in the script by associating commands to objects in the script (or to the script itself) before it is deployed to the database from Script Author. Based on the type of command, the code may be stored in the meta-data of the script itself (e.g., constant commands or PL/SQL commands), in the Scripting session (blackboard commands), on the applications server (custom Java commands exposed to the application by defining the class path), or in the database (custom Java commands deployed to the database from the Scripting Administration console, and PL/SQL packages stored in the database).

See Also

- Scripts
- Global Script Properties
- Oracle Scripting Users
- Using Custom Java

1.8.5 Using Custom Java

Custom Java can be used with Oracle Scripting in two ways: executing commands at runtime, and replacing agent interface panels with Java beans.

Executing Script Author Commands

Scripts that reference custom Java methods associated with a Script Author command can execute each method as specified by the command at runtime.

This applies to all scripts executed in the Scripting Engine, using either the agent interface or the Web interface.

Replacing Panels with User Interface Java Beans

Scripts can also use custom Java beans to replace an entire panel in the runtime session on the agent client workstation. This provides agent users with extended functionality as customized in the Java bean. This bean is executed by Oracle JInitiator on the client.

This applies only to scripts executed in the Scripting Engine agent interface.

> **Note:** Replacing a single answer in a panel with a user interface Java bean is no longer supported functionality.

Java Compilation and Oracle JInitiator Dependencies

> **Note:** This section includes information about Java Development Kit (JDK) and Java Runtime Engine (JRE) compatibility with Oracle JInitiator and with Oracle Applications. The information provided includes version numbers certified at the time of publication. Certification and compatibility information frequently changes. For the latest information, consult Oracle*MetaLink* or Oracle iSupport.

When using custom Java in support of Oracle Scripting, there are compilation dependencies based on specific circumstances. These are based primarily on the purpose of the Java archive in question (to provide Script Author commands or to replace panels with Java beans), and include the following environmental factors:

- The level of Java Runtime Environment (JRE) used on the Apache Web server for your Oracle Applications environment
- The level of Java Development Kit (JDK) used to compile your custom Java code
- The Oracle JInitiator version used on the client
- The Oracle Scripting architecture type used at the enterprise.

Custom Java for use as a Script Author command is executed by the Java Virtual Machine in the Scripting session. Using the Apache mid-tier architecture of Oracle Scripting, custom Java code is executed on the Apache Web server. Thus, this code must be compiled using a version of JDK that is compatible with the JRE used on the Apache Web server (this must be the same level or lower). At this time, appropriate JDK versions may include JDK 1.3, 1.2, or 1.1.8.

Using the caching architecture of Oracle Scripting, custom Java code is executed by Oracle JInitiator on the client workstation (for the agent interface). Thus, this code must be compiled using a version of JDK that is compatible with Oracle JInitiator on the agent client (this must be the same level or lower). At this time, appropriate versions may include 1.3 or 1.1.8.

All scripts executed as surveys using the Scripting Engine Web interface use the JVM of the Apache Web server. Thus, this code must be compiled using a version of JDK that is compatible with the JRE used on the Apache Web server (this must be the same level or lower). At this time, appropriate JDK versions may include JDK 1.3, 1.2, or 1.1.8.

Custom Java that replaces a script panel with a user interface Java bean at runtime executes on the agent client. Thus, this code must be compiled using a version of JDK that is compatible with Oracle JInitiator on the agent client (this must be the same level or lower). At this time, appropriate versions may include 1.3 or 1.1.8.

Java Archive File Format Requirements

The source code for Java methods or Java beans must be compiled into executable class files and packaged into Java archives in one of two file formats, as described in the table below:

Format	Description	Example
JAR	Java Archive (JAR)	SOURCE.JAR
ZIP	WinZip archive	SOURCE.ZIP

Note: Oracle Corporation recommends using JAR file formats, although both JAR and ZIP file formats are supported. For the purposes of this document, the term "Java archive" applies to both file formats, assuming the archive contains appropriately compiled and packaged code.

Note: When the Scripting Administration console references Jar files or Jars, appropriately packaged ZIP files are included in this definition.

See Also

- Scripts
- Global Script Properties

- Oracle Scripting Users
- Custom Code

1.9 Scripting Engine Concepts

This section includes the following topics:

- Scripting Engine Function
- Scripting Engine Agent Interface
- Scripting Engine Web Interface
- Footprinting and Answer Collection
- Script End Users

See Also

- New in this Release
- Modified in this Release
- Obsolete in this Release
- Features
- Architecture
- Responsibilities
- Script Author Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

1.9.1 Scripting Engine Function

The Scripting Engine processes a script in runtime, including displaying the text, images, questions and prompts in the script, interpreting the flow based on end user responses and custom code.

Scripts developed with Script Author can be executed in one of two Scripting Engine interfaces: the agent interface, a Java application running in an Oracle form, and the Web interface, a sequential interpretation of a script with each panel represented by one JSP page.

See Also

- Scripting Engine Agent Interface
- Scripting Engine Web Interface
- Footprinting and Answer Collection
- Script End Users

1.9.2 Scripting Engine Agent Interface

The agent interface is used by customer service or interaction center agents, typically from within Oracle TeleSales, Oracle Collections, or the Customer Support module of Oracle TeleService. Scripts can also be run in the agent interface in "standalone" mode (still from an active Oracle Applications session). However, this is recommended primarily for script testing.

The Scripting Engine agent interface is designed particularly with the needs of interaction center agents in mind. Like the Scripting Engine Web interface, panels (that may contain text and images, and that *must* contain at least one question per panel) display in the central viewing region at runtime. In addition, this user interface includes a progress panel. This feature displays the flow path of the current script, and answers provided for the active session. This allows an agent to have an overall view at all times of where they are in the script. Also specific to this interface are a programmable script information area, which can contain static or dynamic content (text or timers); a programmable shortcut button area that can display buttons containing shortcut commands (to navigate through the script to specified locations, or to associate any Script Author API); a status bar; and a programmable Disconnect button. Optionally, a Suspend button can also be displayed.

See Also

- Scripting Engine Function
- Scripting Engine Web Interface
- Footprinting and Answer Collection
- Script End Users

1.9.3 Scripting Engine Web Interface

The Web interface supports obtaining survey data, feedback or opinions from customers who respond to a survey questionnaire. The survey questionnaire is a

Script Author script executed as a series of JavaServer pages in a Web browser. This user interface includes only panel display. There is no progress area, script information area, button bar, or Disconnect button. End users can use the browser's Back and Forward controls to aid with navigation.

Executing a script in the Web interface first requires administration, in the form of establishing a set of survey campaign requirements using the Survey Administration console. These requirements include creating a survey campaign and defining an associated cycle and deployment, identifying JSP components known as survey resources, and identifying a deployed script as the questionnaire.

- Survey data can be solicited by sending out e-mail invitations and reminders, leveraging Oracle Marketing's list management capabilities and Oracle One-to-One Fulfillment's e-mail template, data field merging, and delivery capabilities. These are known as targeted or list-based survey deployments.
- Survey data can also be solicited by links on an enterprise Web site.
- Finally, survey data can be solicited by customized links on self-service Oracle Application user interfaces such as Oracle iSupport.

Each of these models relies on a survey URL that defines the Oracle Applications instance, and identifies (at minimum) survey campaign and deployment identification code (dID) parameters. Other potential parameters in the URL include a respondent identification code (rID, which associates an individual with a list record), and any custom parameters to pass information into the Scripting session for evaluation and use during the interaction. Accessing this URL launches the script in the Scripting Engine Web interface.

The Scripting Engine interprets the respondent's responses to questions in the script, processes the script, and executes any custom code, passing metadata and instructions to the Web browser to be interpreted on the respondent's client computer.

See Also

- Scripting Engine Function
- Scripting Engine Agent Interface
- Footprinting and Answer Collection
- Script End Users

1.9.4 Footprinting and Answer Collection

Footprinting is the recording in the Oracle Applications database of which panels in a script transaction were visited by a script end user, in sequence, and the duration of time (in milliseconds) before the next panel is requested. Footprint data is stored in the IES_PANEL_DATA table in the Oracle Applications schema.

Answer collection is the recording in the Oracle Applications database of end user responses ("answers") to all answer controls ("questions") that are marked in the script as collectable. Answer collection data is stored in the IES_QUESTION_DATA table in the Oracle Applications schema.

Footprinting and answer collection are both global script properties. For any given script, these features are either on or off. These options can be viewed and set in Script Author by selecting **File > Script Properties**.

If enabled, footprinting data and answers are collected for each transaction or session of the script running in the Scripting Engine, in either interface.

In order for footprint data to be collected, either the Footprinting option or the Answer Collection option must be selected at the global script level. Otherwise, regardless of which specific answers are marked as collectable, information for each script end user will be discarded at the end of each script session.

Footprinting and Answer Collection Dependencies

- Answers are only collected for questions designated as collectable. The Collectable option is a boolean property of a Script Author question (answer definition), represented by a checkbox. *This option is selected by default for all questions defined in Script Author.* You can modify any single question so that answers provided at runtime are not collected, by clearing the Collectable? option selection in the data dictionary for that answer definition.
- Answers are only collected for scripts for which the Answer Collection option is selected. To prevent answers designated as collectable from being collected for all questions in a script, you can clear the Answer Collection option at the global script level. Doing so will prevent survey summary reporting capabilities for scripts with answer collection disabled.
- Answer collection requires footprinting. Therefore, regardless of whether the Footprinting option is explicitly selected, footprinting data will be collected for any script with the answer collection option selected.

  If the Footprinting option is selected but the Answer Collection option is not, only footprinting data for each session or transaction of that script will be saved to IES_PANEL_DATA. If neither option is selected for a specific script, no footprinting or answer collection data is saved.

- Saving footprint data whenever answer collection is enabled ensures, for each script session, a link between each response provided at runtime, and between the specific panel instance from which that response was provided. In the answer collection table (IES_QUESTION_DATA), column PANEL_DATA_ID) contains the foreign key reference to IES_PANEL_DATA (the footprinting table).

See Also

- Scripting Engine Function
- Scripting Engine Agent Interface
- Scripting Engine Web Interface
- Script End Users

1.9.5 Script End Users

This section includes the following topics:

- Agent Users
- Self-Service Web Application Users
- Web Interface Guest Users

See Also

- Scripting Engine Function
- Scripting Engine Agent Interface
- Scripting Engine Web Interface
- Footprinting and Answer Collection

1.9.5.1 Agent Users

Interaction Center agents typically have access to at least one business application from which scripts are executed. Thus, generally, agent users are required to be members of the enterprise database and may require membership in a resource group with a sales role (necessitating importation and group administration using CRM Resource Manager). Therefore, it is recommended that you perform all three user creation and administration tasks for agent users.

See Also

- Self-Service Web Application Users
- Web Interface Guest Users

1.9.5.2 Self-Service Web Application Users

Self-Service Web application users typically execute a script by selecting a survey URL link added to the customized Web application. Thus, they gain access to an Oracle Applications session prior to executing the script (when logging into Oracle Applications to use the integrated self-service Web application). From an Oracle Scripting perspective, these users have no special requirements (roles, responsibilities or functions) outside of those required for the self-service Web applications.

See Also

- Agent Users
- Web Interface Guest Users

1.9.5.3 Web Interface Guest Users

Users of the Scripting Engine Web interface who are *not* using self-service Web applications (survey respondents or Web script users) typically gain access to an Oracle Applications session by entering a survey URL into a Web browser. A valid survey URL provides access to a guest user Oracle Applications account that is restricted to execution of a script only. The guest user is seeded with Oracle Applications; assuming it has not been disabled, this functionality requires no additional setup.

See Also

- Agent Users
- Self-Service Web Application Users

1.10 Oracle Scripting Administration Concepts

Oracle Scripting provides the ability to create, modify, and deploy scripts (using the Script Author component) that can be executed in the Scripting Engine component. The Scripting Engine has two interfaces (the agent interface and the Web interface), both of which display the script at runtime for their intended audience. Each runtime interface interprets the script, end user input, and any custom code associated with the script. If using the Web interface, you must use the Survey component to create and administer guidelines for the script to be executed within a Web browser.

Scripts are deployed to the applications database using Script Author. Scripts may rely on custom Java code, compiled and deployed as Java archive (JAR) or zipped archive (ZIP) files and referenced in the script. Scripts can also reference PL/SQL procedures stored in the applications database.

In support of Scripting operations, you can use the Scripting Administration console to access Script Author as a Java applet; administer scripts and script Java archive files; and monitor Scripting Engine agent interface reports.

This section includes the following topics:

- Scripting Administration Console
- Scripting Administration Console View List
- Agent Interface Reports

See Also

- New in this Release
- Modified in this Release
- Obsolete in this Release
- Features
- Architecture
- Responsibilities
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Survey Concepts

1.10.1 Scripting Administration Console

The Scripting Administration console provides script administrators the interface to launch Script Author as a Java applet, to view and delete deployed scripts, to view and administer Java archive files, to map Java archive files to specific scripts, and to view agent interface reports. This console is accessed by logging into Oracle HTML-based applications using a user account with the Scripting Administrator responsibility.

See Also

- Scripting Administration Console View List
- Agent Interface Reports

1.10.2 Scripting Administration Console View List

When you view any page displaying a summary list in the Scripting Administration console, the set of records which displays in the list is filtered by the parameter selected in the View list. By default, only items created by the Oracle Applications user account with which you are currently logged in display in each summary list. To view items created by all users, change the value in the View list. When you select a filter option from the View list, the page refreshes. The summary view list displays, listing only the objects that meet the selected criteria.

If displaying a list of Java archive files on the Jar Listings or Jar Mapping pages, for example, the value **My Jars** is the default, resulting in the display of all JAR and ZIP files uploaded using the Scripting Administration console with your user name. To display a list of *all* Java archive files deployed from the Scripting Administration console (including those uploaded by other users in this environment), select **All Jars** from the View list.

If using the View menu to display a list of deployed scripts, additional filtering criteria is available based on a script's active or inactive status. Scripts are deployed to the IES_DEPLOYED_SCRIPTS table of the applications database from Script Author. Active scripts, which can be executed by any Scripting Engine of a compatible code level, contain a value of "1" for the ACTIVE_STATUS field within that table. Inactive scripts contain a value of "0" in this table field. Deployed scripts with ACTIVE_STATUS set to "0" are retained in IES_DEPLOYED_SCRIPTS so that existing footprinting and answer collection data can maintain valid references.

If displaying a list of deployed scripts on the Deployed Scripts page, the value **My Active Scripts** is the default, resulting in the display of all active scripts created with your user name. To display a list of all scripts created with your user name regardless of active status, select **My Scripts**. To display a list of all active scripts created with any user name, select **All Active Scripts**. To display a list of all scripts created with any user name, regardless of active status, select **All Scripts**.

Note that if no objects meeting the filter parameter for a certain category are found, then the list headings will appear for the summary table, with no records listed. As soon as an object is created meeting that criteria, it will appear in a refreshed list.

See Also

- Scripting Administration Console
- Agent Interface Reports

1.10.3 Agent Interface Reports

The ability to report and analyze scripts executed in the Scripting Engine agent interface is critical; enterprises can use information collected in the applications database for various purposes. Reports from information collected in scripting-specific tables of the applications database as a result of Scripting operations, and other data collected from customized scripts and stored in custom tables, can be generated using any analytical tool such as Oracle Discoverer or Crystal Reports. Additionally, the Scripting Administration console provides access to panel footprint reports compiling footprinting data.

Additional reporting for scripts executed in the Web interface is available as part of Oracle Interaction Center Intelligence reporting functionality. Implementation of these reports, and access to the Oracle Discoverer tool, is required. These additional reports require the use of Scripting-specific concurrent programs and summary tables specific to survey operations. For additional information, see Oracle Scripting Survey Concepts.

This section includes the following topics:

- Reports and Data
- Analysis and Tuning with the Panel Footprint Summary Report
- Required Report Parameters

See Also

- Scripting Administration Console
- Scripting Administration Console View List

1.10.3.1 Reports and Data

At this time, the only report available through the Scripting Administration console is a panel footprinting summary report. There are two requirements for receiving and reporting data in the Scripting Administration console:

1. In order to appropriately view reports, two script-level parameters should be enabled. These parameters, Footprinting and Data Collection, are established in the global script properties from Script Author prior to deploying a script.

> **Note:** Technically, footprinting is enabled when the data collection option is selected. However, Oracle Corporation recommends explicitly enabling the Footprinting option if footprinting information is desired for the purposes of reporting.

2. To generate a meaningful report, data to be displayed in the report must already be generated. Therefore, scripts must be executed in order to tabulate data displayed in the report. Each time a script is executed (assuming the appropriate parameters are enabled), data regarding the paths taken in the script (footprinting) and the answers selected during the script session (answer collection) are collected in IES tables in the Oracle Applications database.

   For a panel footprint report, this signifies a script with footprinting specified has been executed at least once, either in the interaction center interface by an agent running through a script, or in the Web interface as a respondent participates in a survey using a Web browser.

See Also

- Analysis and Tuning with the Panel Footprint Summary Report
- Required Report Parameters

1.10.3.2 Analysis and Tuning with the Panel Footprint Summary Report

The panel footprint report may be used either for analysis of surveys, or of scripts in use in the interaction center.

Cost Savings in the Interaction Center

This report is overtly useful to enterprises using scripts in the interaction center, as footprint analysis can lead directly to reducing average talk time for an interaction center agent. Doing so results in measurable reduction in costs and increased agent efficiency.

Disabling Footprinting

Footprinting (the act of recording which panels in a script were visited and for how long) can provide useful script tuning data. However, it also consumes system resources. If an enterprise does not need to view individual responses or generate reports, then footprinting should be disabled at the script level to conserve system resources. Note that even if footprinting is not specifically enabled, but data collection is enabled at the script level, then footprinting data will be collected in order to maintain the integrity of data collected.

See Also

- Reports and Data
- Required Report Parameters

1.10.3.3 Required Report Parameters

The panel footprint report takes the respective parameters listed below.

Report Type	Parameters Required to Run Report
Panel Footprint Summary Report	Script Name
	Start Date
	End Date

See Also

- Reports and Data
- Analysis and Tuning with the Panel Footprint Summary Report

1.11 Oracle Scripting Survey Concepts

The survey component of Oracle Scripting allows enterprises to use Script Author scripts as Web-based survey questionnaires that can be executed in a Web browser (over the Internet or on an intranet). The survey component of Oracle Scripting was previously known as iSurvey.

A survey respondent participates in a survey questionnaire by accessing a specific survey deployment URL either from an enterprise's Web site, from a self-service Oracle Application such as Oracle iSupport, or from an invitation e-mail message inviting survey participation. Using the survey component of Oracle Scripting, enterprises can create, manage, and report on surveys to evaluate customer satisfaction, gain customer input on new initiatives, and gain other feedback from survey respondents. The return data can then be used to improve product lines, target new or improved services, or otherwise improve responsiveness.

The collection of requirements for conducting such an effort is referred to as a survey campaign. Survey campaigns are set up and managed from the Survey Administration console.

This section includes the following topics:

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

See Also

- New in this Release
- Modified in this Release
- Obsolete in this Release
- Features
- Architecture
- Responsibilities
- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts

1.11.1 Survey Administration Console

In order to execute a Script Author script in a Web browser using the Scripting Engine Web interface, survey administrators must define survey campaign requirements in an HTML administration console. This is true regardless of whether the script will be executed as a standard or targeted survey, or from a self-service Web application.

The Survey Administration console is an HTML user interface accessed from the Oracle Applications Personal Homepage (PHP) login by a user with the Survey Administrator responsibility. In this interface, a survey campaign is created and its dependencies (survey resources, Script Author script designated as the questionnaire, a survey campaign, cycle, and deployment) are defined.

Additionally, survey administrators can view individual results from a survey (ongoing or completed) from the response view of the Update Deployment page. Answers provided by respondents are stored in the Oracle Applications database schema.

Essentially, the Survey Administration console provides survey administrators the means to set up, execute, and monitor survey campaigns.

See Also

- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.2 The Survey Questionnaire

The survey questionnaire is a Script Author script created to obtain specific information. End users of the Survey component are customers or prospects viewing a survey questionnaire using any Oracle Applications 11*i*-certified Web browser, or individuals executing a Web script from an Oracle self-service application such as Oracle iSupport. As the individual participating in a survey

("survey respondent") or using the Web script moves through each HTML page (corresponding to a single script panel in Script Author), the objects embedded in the script control complicated processing.

On the end user’s desktop, the user is guided through the script either through a rigidly prescribed order, or through a dynamically determined path (as determined by the custom script built based on the needs of the enterprise). The survey questionnaire script displays as a series of pages (one page per panel) in an HTML-based interface (the Web browser). The underlying technology differs slightly, based on the base technology stack used for execution. Survey campaigns using the OAF base technology stack have more options for administering survey campaign resources than for survey campaigns using the deprecated JTT technology stack, but the customer experience for executing scripts is essentially identical at runtime.

On the Apache Web server associated with the Oracle Applications instance for the enterprise conducting the survey or hosting the Web script, all business logic is executed dynamically. This includes the business rules embedded in the survey questionnaire script (such as rules-based branching, data integration, and commands associated with specific objects and events), the end user’s answers, and any custom Java or PL/SQL code.

See Also

- Survey Administration Console
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.3 Survey Administration Console View List

When you view survey campaigns or survey resources in the Survey Administration console, the set of records which displays is filtered by the value in the View list. By default, only items created by the logged in user displays. To view items created by all users, change the value in the View list accordingly and click **Go**.

For example, when viewing survey resources, the default selection in the View list is "My Survey Resources." Correspondingly, the records that display on the page (if any) consist only of survey resources defined in the Survey Administration console by the Oracle Applications user account with which you are currently logged in. To display all survey resources (including those created by other users in this environment), select **All Survey Resources** from the View list and click **Go**. The Survey Resources page refreshes. The summary view table now lists survey resources (if any) created by all users for this environment. This is applicable to survey campaigns as well.

Note that if no objects of that category have been created, then the list headings will appear, with no entries. As soon as you create an object, it will appear in the refreshed list.

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.4 Survey Reports

After summarizing survey data for optimum performance using Concurrent Manager, reports on scripts executed in a Web browser can also be generated using Oracle Business Intelligence.

Custom reports can also be generated from tables in the Oracle Applications schema as desired, using tools such as Oracle Discoverer.

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.5 Survey Hierarchy, Levels and Objects

The Survey component of Oracle Scripting includes three hierarchical levels: survey campaigns, cycles, and deployments. Each level is a set of requirements to ultimately execute survey campaigns.

The collection of requirements for each level is stored as an object in the database. For this reason, survey campaigns, cycles and deployments are described as objects. Each existing survey object can be listed in the Survey Administration console.

When creating a survey campaign in the Create Survey Campaigns page, you also define a child object, the first cycle. You can subsequently modify the cycle name or create additional cycles for an open or active survey campaign from the Update Survey Campaign page.

Deployments are children objects to cycles. Deployments are created from the Create Survey Deployment page, and subsequently viewed and modified from the Update Deployment page.

You can also view responses to individual surveys taken for a deployment from the Update Deployment page by selecting the Response View option.

This section includes the following topics:

- Survey Object Dependencies
- The Parent-Child Survey Objects Relationship

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.5.1 Survey Object Dependencies

General Survey object dependencies and rules include the following:

- Each survey campaign must have at least one cycle. It may have as many as required.
- Each cycle must have at least one deployment. It may have as many as required.
- Child objects cannot be shared, reused or inherited.
- Child objects can be added to existing parent objects. For example, for an active or open survey campaign, you may add a second cycle, or additional deployments.

- Before being executed, each deployment must be deployed (set to active status).
- The collection of requirements that is actually executed is the deployment, the lowest and most granular collection of execution parameters. Thus:
  - Deployments can be grouped into cycles to execute the same questionnaire and parameters over a different timeline for comparison purposes.
  - Cancelling a deployment does not affect the parent objects.

1.11.5.2 Parent-Child Survey Objects Relationship

In order to execute a survey campaign, you must have at minimum three survey objects, directly associated in a parent-child relationship.

- You must define one survey campaign.
- You must create one cycle that belongs to (or is a "child" of) the survey campaign.
- You must create one deployment that belongs to (or is a "child" of) the cycle.

Each parent object must have at least one child in order to create executable survey campaigns. Each child object can have a "one-to-many" relationship with its parent object. Thus, a survey campaign must have one cycle and may have more than one, and each cycle must have one deployment and may have more than one. There is no limit to how many cycles or deployments can be created.

Why Allow One-to-Many Relationships?

The primary reason to create more than one deployment is to use different lists, since lists are associated at the deployment level. Thus, a cycle with several deployments may be executed over the same time period but with different lists. These deployments would then be executed by different audiences over the same period of time.

The practical reason to create more than one cycle for a single survey campaign is to execute the same deployment (or set of deployments) over a different period of time.

As an example of these principles, company ABC decides to conduct a survey campaign to measure customer satisfaction with its products. It wishes to send out the same survey questionnaire, six months apart: once before it introduces a new line of products, and once afterwards. ABC customers are identified by three lists: direct mail customers, Internet customers, and retail customers.

Since the same survey will be executed in two separate time periods, company ABC survey administrators can choose to define two separate cycles (cycle 1 and cycle 2) as child objects of survey campaign ABC. In this model, each cycle represents a given time period for survey execution. Define three deployments (one for each list) for each cycle. Each deployment for cycle 1 has identical start and end dates (for example, January through March), and each deployment for cycle 2 also has identical start and end dates, for the second cycle of time (for example, July through September). All other parameters are identical.

Alternatively, one cycle can be created to contain all deployments. In this model, the first three deployments (one for each list) contain identical start and end dates for the same period (for example, January through March). The last three deployments for this cycle (again, one for each list) use identical start and end dates for the second period of time, for example, July through September.

No Sharing of Survey Objects

You cannot share or reuse existing survey campaign, cycle, or deployment objects. Nor can you copy an existing object and make modifications to one or more parameters. Even if you require a survey object (for example, a deployment) with the same set of parameters, you must redefine the object, associating it with its parent object (for cycles and deployments).

The only objects that can be shared and used are survey resources.

Therefore, in the above example, Cycle 1 and Cycle 2 will need to be created separately. Also, each of the three deployments for the three ABC lists will need to be created each twice, once for each cycle.

1.11.6 Survey Campaigns

The Survey component relies on the concept of a *campaign*: a focused effort to achieve a particular goal from a targeted population over a specific period of time for a particular business purpose. The goal of a *survey campaign* is typically for an enterprise to obtain specific data by polling a target market segment or population for subsequent analysis and subsequent action. Typical actions might include the offering of a new product or service, or a change in business processes to improve satisfaction. As part of a self-service application, a survey campaign typically measures satisfaction regarding the customer experience for that self-service application.

Survey campaigns have requirements that must be defined based on the campaign goals (for example: target population, purpose for obtaining the information, which information will be gathered, for what purpose, how long the campaign will be conducted, and in how many separate deployments of the same requirements).

The campaign goals are achieved by two processes: creating a survey questionnaire built with the campaign goals in mind, and in administering the campaign from the enterprise. The Survey Administration console supports these functions.

The survey campaign is the top-level object for creating and executing survey questionnaire campaigns. Survey campaigns are created using the Survey Administration console, the primary user interface for the Survey component of Oracle Scripting.

This section includes the following topics:

- Survey Campaign Dependencies
- Survey Campaign Status

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.6.1 Survey Campaign Dependencies

Each survey campaign requires:

- A script

  A script serving as the survey questionnaire must already be created and deployed to the applications database using the Script Author component of Oracle Scripting.

- A name to identify the survey campaign
- If executing surveys campaigns using the deprecated JTF technology stack, three JSP survey resources (header section, final page, and error page survey resources), as described in the section Administering Survey Resources.

  An optional footer section survey resource can also be defined. Each is a JSP page that must be defined in the Survey Administration console, and physically loaded to the $OA_HTML directory on the applications server.

  If executing surveys campaigns using the new Oracle Applications Framework technology stack, you can use HTML survey resources (for all resource types) or URL survey resources (for error page and final page resources). These can be defined and uploaded from the Survey Resources tab.
- At least one cycle

Survey Campaign Status

Status	Description
Open	When it is initially created, a survey campaign is open and remains in this state until the first deployment is activated.
Active	When any deployment is activated, the parent survey campaign changes from open to active status. Once active, a survey campaign status may change to idle only. Its status never returns to open, nor can it be closed prior to changing to idle status. Active is only valid from the open or idle status. A survey campaign cannot be manually set to active by the survey administrator; this status is established by the system based on these business rules.
Idle	A survey campaign status is idle if at least one of its deployments was activated in the past (the survey campaign was previously active and the deployment was previously either pending or active), but the survey campaign currently has no active or pending deployments. Idle is only valid from the active status. A survey campaign cannot be manually set to idle by the survey administrator; this status is established by the system based on these business rules.
Cancelled	If a survey campaign status is open (after it is created), but has either had no deployments defined or has had no deployments activated, the survey campaign can be set to a status of cancelled, indicating that its purpose is no longer relevant. Once cancelled, no updates can be made to the survey campaign or any of its children objects (cycles or deployments). Cancelled is only valid from the open status. Only a survey administrator can change the status of an open survey campaign to cancelled.

Status	Description
Closed	If a survey campaign is idle (if it currently has no active deployments), and its deployments have been successfully run for the intended duration, the survey campaign can be set to a status of closed, indicating that its purpose was fulfilled. Once closed, no updates can be made to the survey campaign or any of its children objects (cycles or deployments). Closed is only valid from the idle status. Only a survey administrator can change the status of an idle survey campaign to closed.

Once created, and prior to defining a deployment, the status of a survey campaign is Open. You can view the status from the Survey Campaigns page. For open survey campaigns, you can change the status from the Status list by selecting **Cancelled** from the list. This invalidates the survey campaign definition, ensuring it can no longer be used.

Unless you explicitly change the status of a survey campaign after creation, it remains open until a child deployment is activated. Once a deployment is activated, the parent survey campaign status changes to active. A survey campaign remains active as long as at least one of its deployments is active or pending. From active, a survey campaign can become idle, and from idle to cancelled or closed. Cancelled and closed statuses only result from manual intervention by the survey administrator. An active survey campaign can automatically change to idle based on events related to its deployments. For more information, see the section on deployment status.

Once all the deployments for a survey campaign have executed successfully, and the survey campaign is in idle status, you can close the survey campaign from the Status list by selecting **Closed** from the list. This indicates that the survey campaign has served its business purpose. Once a survey campaign is closed, no properties of the closed survey campaign, its cycles, or deployments, can be changed. Return data remains available for viewing responses, or from Oracle Business Intelligence, survey reports can be executed for analysis of survey returned data.

1.11.7 Cycles

A cycle is a child object of the survey campaign. It is the smallest set of requirements for survey campaigns, including only a cycle name, and is defined at the same time its parent object (the survey campaign) is created. Each survey campaign must have at least one cycle defined, and may have many.

The purpose for the cycle object is to provide the capability of grouping its children objects (deployments) for reporting purposes. When analyzing data from executed survey deployments, an enterprise can compare the results of one cycle (one grouping of deployments) to the results of another cycle executed at a different time. This comparative analysis provides the ability to quantify over time effectiveness, satisfaction, or other metrics related to the purpose for a survey campaign.

Cycles are defined from the Survey Campaign tab of the Survey Administration console for any open, active, or idle survey campaign. As with all information entered into the Survey Administration console, the cycle name should follow any existing predefined survey campaign requirements provided to the survey administrator.

Cycle Dependencies

Each cycle requires a cycle name. Oracle Corporation recommends that each cycle be uniquely named within a single survey campaign.

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Deployments
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.8 Deployments

A deployment is the most detailed set of execution parameters. There are two types of deployments: targeted (previously referred to as list-based) and standard (previously referred to as non-list-based).

Targeted deployments use Oracle Marketing lists to send e-mail invitations (through Oracle One-to-One Fulfillment) to each member of the list, inviting them to participate in answering a survey questionnaire. These list members can be tracked by individual respondent.

Standard deployments are anonymous and have fewer requirements, as detailed below.

This section includes the following topics:

- Deployment Dependencies
- Targeted Deployment Dependencies
- Deployment Status

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Survey Resources
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

Deployment Dependencies

Each deployment requires:

- A parent survey campaign and cycle
- A deployment name
- A media type (currently Web)
- A status
- A deployment start date and time (equal to SYSDATE or in the future)
- A response end date and time (in the future)

- A deployment type (standard or targeted)

Targeted Deployment Dependencies

For list-based deployments only, the following are also required:

- List name
- Maximum number of responses per person
- Target response percentage
- Hosting option (standalone or menu-based)
- Survey Web URL (automatically generated from logged-in Apache Web server)
- Invitation template name (identifying invitation master document)
- Invitation e-mail message subject heading

If using reminders in addition to invitations, then the following are also required:

- Reminder template name (reminder master document)
- Reminder e-mail message subject heading
- Number of reminders
- Reminder interval (in days)

Deployment Status

Status	Description
Open	A deployment is open when the deployment is initially created and remains in this state until activated by the survey administrator.
Pending	When a targeted deployment is activated, if the deployment date and time are in the future, its status changes from open to pending. The deployment remains in this state until the deployment start date and time equal SYSDATE. At this time, the SUBMIT GROUP FM REQUEST FROM IES concurrent program executes, causing the fulfillment request to be submitted to the fulfillment server. Upon a successful submission of the fulfillment request, the deployment status changes to Active. Pending is only valid from the open status, and applies only to targeted deployments
Error	Error status indicates that the concurrent program generated an error while attempting to submit the fulfillment request to the fulfillment server. This status does not include errors that occur after the fulfillment request is successfully sent to the server. If problems occur that prevent invitation or reminder e-mail messages from being sent, the deployment status remains active, and fulfillment debugging should commence by a fulfillment administrator from the Oracle One-to-One Fulfillment administration console. Error is only valid from the pending status, and applies only to targeted deployments.

Status	Description
Active	A standard deployment is active immediately after it is activated by the survey administrator. A targeted deployment is active when the deployment date and time equal SYSDATE, and the fulfillment request is successfully sent to the fulfillment server by the concurrent program. Note that active status for targeted deployments does not indicate successful delivery of invitation or reminder e-mail messages.
Cancelled	If a deployment has a status of open, its status can be manually set to cancelled, indicating there is no more need to execute this set of requirements. After being cancelled, a deployment cannot be executed. Cancelled is only valid from the open deployment status.
Incomplete	If a deployment has an active status but there is a no interest on the part of the surveying enterprise to view the results, it can be set to incomplete status.
Closed	If a deployment is in active status and has run for its intended duration or has resulted in sufficient responses, it can be set to a status of closed, indicating that its purpose was fulfilled. Deployments in a pending or error status can also be set to closed.

Once a deployment is created, and before it is activated, its status is Open. You can view the deployment status from the Update Survey Campaign page or from the Update Deployment page. From either of these pages you can also change the status of an open deployment by selecting **Cancelled** from the status list. This invalidates the deployment definition, ensuring it can no longer be used.

The only way to change the status of a standard deployment to active is to activate it from the Update Deployment page. The Update Deployment page will refresh, with the status set to Active and a valid survey hyperlink displayed. This is true whether the deploy date is in the past or the future.

The only way to change the status of a targeted deployment to pending is to activate it from the Update Deployment page. If the deployment date and time are in the future, the status will change to pending until they equal SYSDATE. If the deployment date and time are in the past, the concurrent program will attempt to submit the fulfillment request to the fulfillment server immediately. Upon successful submission of the fulfillment request to the server, the deployment status will change to active. For targeted deployments, the deployment details will not display a survey URL, since respondent identification numbers for each list member are required.

Once a deployment is activated, and before the deployment end date is reached, its status can change from active to closed or incomplete. Do this from the Update Deployment page.

- Use incomplete to indicate that there is no interest in executing this deployment and viewing its results.

- Use closed when the deployment goals have been achieved. For example, after the deployment start and end date have passed and the deployment was successfully executed, change active status to closed. As another example, close the deployment if you determine that a sufficient number of responses for a deployment have been received to perform the required analysis or make the appropriate business decisions for which the survey campaign is intended.

For targeted deployments, after the deployment is activated, it may contain either a status of pending (indicating that it is activated and the concurrent request is in the queue or its start date is in the future), active (indicating that the concurrent request completed successfully), or error (indicating that the concurrent program generated an error while creating and sending the fulfillment request). Deployments with a status of error can be modified so that the error can be corrected, and when reactivated will again display a pending status. Targeted deployments can also contain a status of closed or incomplete, following the same guidelines described above for standard deployments.

Error status cannot be set manually. This status, for targeted deployments only, indicates that the concurrent program generated an error while attempting to submit the fulfillment request to the fulfillment server.

Additional Requirements for Targeted Deployments

For targeted survey deployments, once a survey deployment is activated, you must wait until the fulfillment engine completes its activity (completes the fulfillment request and succeeds in sending the invitation master documents to the outgoing mail server specified in Oracle One-to-One Fulfillment) before you will have respondents. The concurrent request ID is listed on the Update Deployment page when displayed in the deployment view, and can be used to track the concurrent request and its status.

Survey administrators who have also been assigned the JTF role JTF_FM_ADMIN are able to view the status and history of fulfillment requests from the Invitations tab of the Survey Administration console.

> **Note:** Granting JTF roles requires the grantor to be assigned the JTF system administrator role, JTF_SYSTEM_ADMIN_ROLE, in addition to whichever other JTF role you want to assign. For this purpose, if required, you can use the seeded SYSADMIN Oracle Applications user account. For more information, see Understanding Users Required for Implementation.

When viewing status, a request status of **Submitted** indicates that the list was successfully sent to the fulfillment engine. An outcome code of **Success** indicates that the fulfillment server was successful in sending the invitation (or reminder) master document. For any other outcome code (e.g., **Partially Successful** or **Failure**), you will need to log into the Fulfillment Administration console to resolve. In this scenario, list members must receive and respond to an e-mailed invitation before you can expect activity for this deployment.

To access the Fulfillment Administration console, a user must be assigned the JTF_FM_ADMIN role and the One-to-One Fulfillment Administrator responsibility. Consult with an Oracle One-to-One Fulfillment administrator if required.

Effects of Setting Deployment to Closed

Immediately upon changing a deployment's status from active to closed, the status for its parent survey campaign changes from active back to idle, *if there are no other active or pending deployments.* The survey campaign will remain active if other deployments have been activated and have not encountered an error.

With survey campaigns for which the last deployment has been changed to closed, you can close the survey campaign, or create another cycle or deployment if you wish to receive more responses for this survey campaign.

Closing a survey campaign or deployment signifies that the requirements or objectives for that object have been met. No new information can be received from a closed survey campaign or deployment.

From the Survey Administration console, responses can still be viewed for closed deployments.

Survey reports can still be viewed for closed deployments.

Effects of Setting Deployment to Incomplete

Once you designate a deployment as incomplete, if there are no other active or pending deployments, the parent survey campaign changes to idle, and can be closed unless you wish to create new cycles or deployments to obtain additional responses. The incomplete deployment status indicates that the requirements or objectives for the deployment are no longer relevant. No new information can be received from a cancelled deployment. While responses and reports can be viewed for this deployment, the information generated by reports is likely to be incomplete.

1.11.9 Survey Resources

This section includes the following topics:

- Survey Resources Differentiated by Technology Stack
- Detailed Description of Survey Resources
- Creating Survey Resources
- Seeded JSP Survey Resources

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Prototypes
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.9.1 Survey Resources Differentiated by Technology Stack

Within the 11.5.10 release, the survey administration console supports two technology stacks for surveys executed at runtime. Survey resources and their requirements differ by technology stack.

The original technology stack implemented by Oracle Scripting is the JTT technology stack. This is still supported in this release, but deprecated from this point forward. The second is the OA framework technology stack. While at runtime there are some small differences in the way scripts appear in a Web browser based on the technology stack, the main manifestation of these differences is evident when administering survey campaigns and when establishing the survey resources that support them.

The base technology stack assigned to a survey campaign cannot be changed after the survey campaign definition is saved. If you are creating new survey campaigns, use of the new technology stack is strongly recommended.

While survey resources provide the same essential functionality regardless of technology stack, they are implemented dramatically differently. The new framework supports more options and more flexibility. The technology stack determined when defining a survey campaign invokes business rules enforced by the application. For example, if you select JTT as the base technology stack for a survey campaign, no OAF resources will be available to assign to that campaign. If you select OAF as the base technology stack, no JTT resources will be accessible to that campaign.

For all sections of this document in which survey resources are discussed, the text is qualified by which technical stack is to be used at runtime.

See Also

- Detailed Description of Survey Resources
- Creating Survey Resources
- Seeded JSP Survey Resources

1.11.9.2 Detailed Description of Survey Resources

Survey resources are objects defined as part of a survey campaign that display in a Web browser to users of the Scripting Engine Web interface at runtime. Survey campaigns support four survey resources (header section, footer section, error page, and final page resources), which include two display types (section resources and page resources).

The two display types appear differently at runtime:

- **Section resources** appear as a section of each HTML page that represents a panel in the script. These do not render in page resource display types.

  The header section appears as a header in each HTML page (panel), just above the panel contents (panel text, graphics, questions, and a Continue button).

  The footer section appears at the foot of each HTML page (panel), immediately below the Continue button.

- **Page resources** display as a separate page at the appropriate time during the execution of a script in a Web browser, and do not represent panels.

  Error page resources are displayed when an error condition occurs during script execution in a Web browser.

  Final page resources are displayed after the Scripting Engine processes the last panel in the flow of a script.

Survey Resources at Runtime

The header and footer section survey resources display, respectively, at the top and bottom of each page of a script executed in a Web browser. The center of the HTML page (below the header section and above the footer section) is where each panel of the script appears. The error page and final page survey resources are full HTML pages. When an error condition occurs while the end user is executing a script, the error page is displayed. After the last panel in a script, the final page is displayed. The full-page error page and final page survey resources do not display the header or footer section resources.

Survey Resource Components

Survey resources are comprised of two components. The first component is the survey resource *definition*. The second component, required for all survey resources except a URL, is the *physical* survey resource file itself.

Survey Resource Definitions Survey resource definitions are created from the Survey Resources tab of the Survey Administration console. When you define a survey resource, you create an object in the database that points to either a physical file (HTML or JavaServer Page, based on technology stack), or a URL (for error page or final page resources, supported for the OAF technology stack only). Survey resources *must* be defined before they can be used in a survey campaign. Once defined, survey resource definitions persist; the same resources can be used by any number of survey campaigns (of the same technology stack) without limit.

Any survey resource you want to reference when creating a survey campaign *must* be defined *before you create the campaign.* Unlike associating a script with a survey campaign, however, you can change any survey resource associated with an open or active survey campaign after survey campaign creation.

Physical Survey Resources The second component of a survey resource is the *physical file* itself. It corresponds to the logical definition of the survey resource, and must be available at runtime (resulting in an error if it is not).

Physical survey resource files must be created by certified, knowledgeable HTML or JavaServer Page developers. This creation is outside the scope of Oracle Applications.

Business rules for physical survey resources differ based on the following:

- Resource type (HTML file, JSP file, or URL)
- Required location for the physical file (database or APPL_TOP) for runtime execution

- Technology stack of the survey campaign

HTML survey resource files are stored in the applications database. Upload the fully tested HTML files when you define the survey resource. HTML survey resource files are supported for OA framework survey campaigns only.

JSP survey resource files are stored on the applications server, in the $OA_HTML directory of the APPL_TOP. There are no mechanisms in place from Oracle Scripting to upload this file. Consult with a system administrator or the Apache Web server administrator if you do not have physical access to the APPL_TOP. JSP survey resource files are supported for JTT survey campaigns only.

URL survey resources must exist on a Web server accessible to the Apache Web server at runtime, at the absolute location corresponding to the definition (for example, http://xxx.xxx). You cannot define URLs as relative (for example, ../home/index.html). To use this survey resource type, it is assumed that the URL already exists on a Web server. If not, the file must be created and loaded so that the URL can be accessed at runtime. This process is outside the scope of Oracle Scripting; no physical component is typically required for a survey administrator using this resource type. URL survey resource files are supported for OA framework survey campaigns only.

Resource Differences Between Technology Stacks

JTT Survey Resources The physical survey resource files for JTT survey campaigns must be saved in JavaServer Page (JSP) format (with a file extension of .JSP), even if they contain no dynamic content.

Header section, error page, and final page survey resources must all be referenced upon creating a JTT survey campaign. Footer section resources are optional.

In support of JTT survey campaigns, four test JSP files are seeded in the appropriate directory on the APPL_TOP, and may be used to test survey functionality. Although these test survey resources exist in the $OA_HTML directory, as with all survey resources, they must still be defined prior to use.

No seeded footer survey resource sample is included with Oracle Applications. For test purposes, you can reference a header resource in the Footer Section field, to ensure it displays as appropriate at runtime.

For more information, see Oracle Scripting Survey Concepts > Survey Resources > Seeded JSP Survey Resources.

OA Framework Survey Resources For survey campaigns defined in the OAF technology stack, survey resources are either URLs for redirect or hypertext markup language (HTML) files. Both `.HTM` and `.HTML` file types are supported (regardless of case). JSP resources are not supported for this technology stack.

For OAF survey campaigns, error page or final page resources can also be URLs accessible to the Web server at script runtime, which redirect the script end user to the specified URL during an error or upon completing the script, respectively. OAF survey campaigns do not require survey resources to be referenced at runtime, although any resources that are referenced must first be defined.

The physical HTML file corresponding to an OAF survey resource definition is uploaded from the Define Survey Resources page of the Survey Administration console at the time of survey resource definition. In the typical flow of creating and administering survey campaigns, the definition of the survey resource (and the uploading of the physical file to the database) precedes the creation of a survey campaign referencing that resource. However, the survey resource used by a survey campaign can be changed at any time (until canceled or completed).

In contrast, survey resources used for JTT survey campaigns can be defined (including the file name of the physical resource) and referenced in a JTT survey campaign before the physical file exists, and before it is uploaded to the applications tier. However, the physical JSP files referenced by their corresponding survey resource definitions must be uploaded to the $OA_HTML directory on the applications server prior to execution of the script referenced in a survey campaign at runtime.

Resource Specification Not Required for OAF Survey Campaigns OA Framework survey campaigns do not require survey resources to be associated to execute scripts at runtime. If header section and footer section survey resources are not associated with the survey campaign, then headers or footers will simply not appear in the header or footer sections of the HTML page at runtime. If no specific error page or final page resource is specified, a simple error page or final page will be automatically generated by the application. If defining a static HTML page as an error page for OAF survey campaigns, you can also require error information to display on top of the error page by selecting the **Display Error on Top** box.

Survey Resource Properties

Survey resources have the following properties:

Property	Description
Logical Name	Identifies the survey resource definition.
Physical Name	The name of the physical file defined in the application by the survey resource definition. Stored in the database (for OAF survey campaigns) or in the $OA_HTML directory (for JTT survey campaigns). Not a property of URL resource types, which instead have a resource URL.
Resource URL	The absolute path for a Web page to which the user will be directed (for OAF survey campaigns at runtime). Applicable to page display type resources only. Not a property of HTML or JSP resource types, which instead have a physical name.
Description	An optional field to contain information describing the survey resource. This property is not displayed in the survey resources list, and can only be viewed from the Survey Resource Details page.
Language	Identifies the language intended to be used to execute in the Scripting Engine Web interface.
Resource type	Identifies the function of the resource. Options include HTML File Upload, Deprecated - JSP, and URL For Redirect.
Display Type	Identifies the function of the survey resource at runtime. Two display types include section resources and page resources. Section resources (when defined for a survey campaign) appear in a designated section of each HTML page representing a panel in the script. Page resources are full pages that appear either upon the browser reaching an error condition, or after displaying the last panel in a script. Page resources do not contain panel content and do not display section resources.

The table below summarizes the attributes of survey resources discussed in this section.

Survey Resource Type	Technology Stack Supported	Section Display Type Supported?	Page Display Type Supported?
Deprecated - JSP	JTT	Yes	Yes
HTML File Upload	OA Framework	Yes	Yes
URL For Redirect	OA Framework	No	Yes

Guidelines

- The Survey Resource Type column lists the type of survey resource as listed in the Resource Type column in the Survey Administration console UI.
- The Technology Stack Supported column indicates which base technology stack is supported by the designated survey resource type.
- The Section Display Type Supported? column indicates whether the section survey resource type (applicable to header sections and footer sections) is supported by the designated survey resource type.
- The Page Display Type Supported? column indicates whether the page survey resource type (applicable to error pages and final pages) is supported by the designated survey resource type.

Base Technology Stack Options

As of this release, the survey component of Oracle Scripting supports two runtime execution models, based on an older and newer technology stack.

Technology Stack Name	Description in UI	Description	Status	Recommendations
JTT	Deprecated - UI	Oracle CRM Technology Foundation (JTT) technology stack.	Deprecated	Do not use
OAF	OA Framework	Oracle Applications Framework	Current	Use for all new survey campaigns

Oracle Corporation strongly recommends creating all new survey campaigns to execute using the Oracle Applications framework. Benefits include:

- Ability to create and execute survey campaigns without defining resources.
- Ability to redirect users to a specified URL upon error condition or upon visiting the last panel in the script.
- Ability to upload HTML resources to the database from the administrative UI.

JTT survey campaigns are still supported at this time, since survey campaigns previously created using this deprecated technology stack may be required to execute for some time to come. Improvements in the software may not be available in the older technology stack as the product evolves. Additionally, enterprises that continue to create new survey campaigns for execution in the deprecated JTT technology stack risk a future loss of data collected during the course of a survey campaign cycle, when upgrading to a subsequent release in which the deprecated technology stack may become obsolete or non-functional. Since collecting survey data may be the primary purpose for executing a survey campaign, such loss can be catastrophic to the business goals of the enterprise. Oracle Corporation is not responsible for a loss of data caused as a result of remaining with a deprecated technology stack.

See Also

- Survey Resources Differentiated by Technology Stack
- Creating Survey Resources
- Seeded JSP Survey Resources

1.11.9.3 Creating Survey Resources

Creating OAF Survey Resources

Creating and modifying the physical HTML files that serve as survey resources for OAF survey campaigns are not accomplished from within the Survey Administration console.

Creation of the physical files for use as survey resources is outside of the scope of Oracle Applications. You can, however, upload HTML files intended for use as survey resources from the Survey Resources tab, at the time of survey resource definition. HTML survey resources are stored in the applications database and available to any survey campaign in the same environment that references the resource at the survey campaign level.

Creating JTT Survey Resources

Creating, modifying, and uploading the physical JSP files that serve as survey resources for JTT survey campaigns are not accomplished from within the Survey Administration console.

Physical survey resource files must be created by certified, knowledgeable JavaServer Page developers, and uploaded to $OA_HTML on the applications tier of the applications server by a system administrator or other individual with the appropriate privileges and access to the APPL_TOP.

Creation of the physical files for use as survey resources is outside of the scope of Oracle Applications.

However, as described in the section Test Survey Resources, four test JSP survey resource files ship with Oracle Applications, seeded in the appropriate directory ($OA_HTML) on the applications server. You can use the four test resources to test your implementation of JTT survey campaigns, and to serve as building blocks, modifying copies of these files for your own use as appropriate.

Note: JSP files *must* be tested on an existing Web server to view appropriately.

Note: Like all customizations, any modifications you make to survey resources are not supported by Oracle Corporation.

Oracle Corporation recommends using individuals certified in Java development and JavaServer Pages technology to create or modify JSP survey resources.

Including Graphics in Survey Resources

Survey resources, like any other HTML or JSP page, may include images and hyperlinks. Survey resources must be located in the $OA_HTML directory on the applications server to use at runtime. Any objects (such as GIF or JPG images) referenced in a survey resource page must also be available to the application server at runtime.

Using Default OAF Survey Resources for Implementation Testing

If you do not associate custom survey resources for a survey campaign using the OAF base technology stack, no header or footer sections will appear for HTML pages representing panels. In addition, a default error page and final page will appear at the appropriate time (upon error, or after visiting the last panel in a script). Omitting the section resources, and utilizing the default page resources, provides a method to verify that survey resources are appropriately displayed upon execution of a survey campaign deployment using the OAF technology stack.

At the same time, you are provided with a layer of abstraction in regard to the need to ensure the product is performing as expected without needing to test the creation of tailored HTML pages. Subsequent to successful execution of a script in a Web browser with all appropriate panel content and default resources displaying, you can assign custom survey resources to an OAF survey campaign and retest.

See Also

- Survey Resources Differentiated by Technology Stack
- Detailed Description of Survey Resources
- Seeded JSP Survey Resources

1.11.9.4 Seeded JSP Survey Resources

Survey resources must be defined in the Survey Administration console and must map to existing JSP documents residing on the server (as described in Uploading JSP Survey Resources.

For survey campaigns in production, you will most likely want to create your own resources. For testing and implementation verification, you may want to use the survey resources seeded with Oracle Applications.

Test Survey Resources

Test JSP survey resources, identified in the table below, ship with Oracle Applications beginning with IES MiniPack G, and available with any Rapid Install from release 11.5.5 and later. These resources are physically located in $OA_HTML on the applications server.

Resource	File Name
Header section	IESSVYTESTHEADER.JSP
Error page	IESSVYTESTERROR.JSP
Final page	IESSVYTESTTHANKU.JSP
Hosted survey error page	IESSVYMENUBASEDTESTERROR.JSP

Note that there is no seeded footer section resource. For testing purposes, to ensure a footer will appear as expected in a production environment, you can use another JSP page (for example, the seeded header section).

Using Test JSP Survey Resources for Implementation Testing

Utilizing these test survey resources provides a method to verify that survey resources are appropriately displayed upon execution of a survey campaign deployment using the JTT technology stack in an HTML user interface. At the same time, you are provided with a layer of abstraction in regard to the need to ensure the product is performing as expected without needing to test the creation of tailored JSP pages. For this purpose, for JTT survey campaigns it is recommended that you first test execution of scripts in a Web browser using seeded test survey resources listed above, to ensure successful implementation. Subsequent to successful execution of a script in a Web browser with all appropriate test resources displaying, you can change the survey resources assigned to a survey campaign to use customized survey resources.

Additional JSP Error Page Resource for Hosted Surveys

Header section and final page seeded JSP survey resources are intended for use with standalone and menu-based survey deployments for JTT survey campaigns. The IESSVYTESTERROR.JSP error page is specifically intended for use with standalone deployments. For hosted scripting operations (self-service Web applications such as Oracle iSupport, or other self-service Web applications customized to provide access to scripts executed in a Web browser), the IESSVYTESTERROR.JSP error page should be used.

To test a hosted scenario, you can use any JSP header section (or HTML page saved in JSP file format). Again, you can use the seeded test header section to test footer section functionality as well. Error and final pages must also be in JSP file format. The error page must adhere to the hosted template. There are two ways to handle the final page:

1. Define as the final survey resource page the JSP page which displays the list of URLs to take the survey. In this way, when the survey is completed, the respondent will be returned to the starting page.
2. Define another JSP page as the final page survey resource, but include in that page a JSP forward command to point the user back to the page that displays the list of URLs to take the survey.

Sample Code for Test JSP Header Section Resource

Following is the source code for the sample test JSP header section. *This sample is provided for informational purposes only.* Oracle Corporation is not responsible for the correct implementation of JSP in your environment, and does not provide support for customized survey resources. Consult appropriate HTML and JSP experts for more information.

```
<!-- $Header: iessvytestheader.jsp $ -->
<table align=center border=0>
     <tbody>
      <tr>
         <td>&nbsp;</td>
         <td class=pageTitletcolspan=4 noWrap>Test Header Section</td>
```

```
          <td>&nbsp;</td>
        </tr>
      </tbody>
  </table>
```

See Also

- Survey Resources Differentiated by Technology Stack
- Detailed Description of Survey Resources
- Creating Survey Resources

1.11.10 Prototypes

Prototype is a boolean characteristic of survey campaigns. The default for this characteristic is null (survey campaigns are not designated as prototypes unless you select this option when creating or editing a survey campaign).

Prototype survey campaigns are identical to standard survey campaigns, except that the script used as the survey campaign questionnaire is not locked. The purpose is to allow survey campaign administrators more freedom to refine requirements for survey campaigns, including modification to the script for the designated survey campaign.

The Survey Campaigns tab includes an option to exclude prototypes. Selecting this option filters prototypes out of the lists of survey campaigns displayed. This option is also null by default (prototype survey campaigns are included in survey campaign lists unless you select this box and click Go).

Although you can also put a prototype survey campaign into production, the script will not be locked. The prototype option can be selected or cleared while a survey campaign status is Open. Thereafter, you cannot change this option.

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns

- Cycles
- Deployments
- Survey Resources
- Survey URL
- Concurrent Programs Supporting Survey Operations

1.11.11 Survey URL

When a survey deployment is activated, the survey administration function of Oracle Scripting generates a URL through which the survey deployment is accessed using the Scripting Engine Web interface.

For standard (non-list-based) surveys or Web scripts, this URL (as generated) can be used to directly execute the survey questionnaire script in an Oracle Applications 11*i*-certified Web browser, as soon as the deployment is activated.

The URL for targeted (list-based) survey deployments is not complete upon survey deployment activation. When the Oracle Marketing list is subsequently processed by Oracle One-to-One Fulfillment, additional URL parameters are concatenated to the system URL for each list member, to identify the unique respondent from the Oracle Marketing list. This complete URL, including respondent ID, is included in each invitation or reminder master document sent from the fulfillment server. If the Maximum Responses Per Person deployment parameter is set to 1 for a targeted deployment, this enforces uniqueness (based on the respondent ID) so that the designated list member can only execute the script in a browser once. For a specific list member, the same respondent ID is used for each instance of an invitation or reminder in a cycle.

The construction of the survey Web URL can differ based on these factors:

- Base technology stack of the parent survey campaign (OA Framework or JTT)
- Deployment type (standard or targeted)
- Hosting options (standalone or menu based)

Survey URL Syntax

The general syntax of the survey URL is:

```
http://<server name>.<domain>:<Apache Web server port>/OA_HTML/<hosted,
standalone, or OA survey JSP template>?<Deployment ID>&<Respondent ID>&<database
connection (DBC) information for current session>
```

Survey URL parameters

- The server name and domain identify the Web server and organization on an Intranet or the Internet.
- The Apache Web server port identifies a specific Apache web server instance either protected by a corporate firewall or accessible to the networked world at large.
- OA_HTML is the Oracle Applications bin on the Oracle Applications server in which HTML files are stored for execution.
- The JavaServer Page template specifies a set of criteria used for the execution of the script in a web browser client using the Scripting Engine Web interface. This parameter identifies which base technology stack is used to execute the script at runtime. For JTT deployments, it also indicates the hosting option selected.
    - If this parameter uses the OA.JSP template, the deployment is created as part of a survey campaign with the Oracle Applications Framework selected as the base technology stack. When the URL is invoked in a Web browser, the script will execute in the Web browser at runtime using the OA Framework technology stack. A URL for such a deployment might appear at runtime similar to the following:

      ```
      OA.jsp?OAFunc=IES_SURVEY_OARUNTIME
      ```

    - If this parameter uses IESSVYMAIN.JSP or IESSVYMENUBASED.JSP as the template, the deployment is created as part of a survey campaign with JTT selected as the base technology stack. When the URL is invoked in a Web browser, the script will execute in the Web browser at runtime using the deprecated JTT technology stack.
    - The JSP used identifies the source of authentication for the Oracle Applications session in which the script is executed. If the base JSP template is IESSVYMAIN.JSP, authentication for the Oracle Scripting session is provided using a guest user. The only function allowed by that guest session is to execute a script in a Web browser, one time, and then the Oracle Applications session is terminated when the final page resource or URL is displayed.
    - If the base JSP template is IESSVYMENUBASED.JSP, authentication from an existing Oracle Applications session is used to launch the Oracle Scripting transaction in a Web browser, and the tab-based menus associated with that session are hosted in the survey or Web script user interface. After the final page resource or URL is displayed, the applications session is maintained,

and the user can access any functionality accessible in the hosted application.

- Deployment ID specifies a unique survey campaign deployment in a specific instance of Oracle Applications.
- Respondent ID is only used for targeted deployments, and identifies a specific list member on an Oracle Marketing list. For standard deployments, this parameter is omitted.
- DBC information specifies the database connection information for an authenticated Oracle Applications session.

JSP Templates for Scripting Engine Web Interface Runtime Execution

Users of the Scripting Engine Web interface access scripts either as survey questionnaires, or as Web scripts. Executing a script as a Web-based survey involves only a single Oracle Scripting transaction (and precludes multiple transactions). Web script use cases support multiple Oracle Scripting transactions if required. The number of Oracle interactions allowed, as well as the authentication scheme and method of initiating an Oracle Applications session, is governed by the use of a specific JavaServer page as the survey or web script template. The template used differs based on two factors: the base technology stack selected for execution at runtime, and the intended authentication for the session. Oracle Scripting currently supports three JSP templates, as described in the table below.

JSP Template	Tech Stack	Authentication	Description
OA.JSP	OAF	For standard deployments, guest user authentication or menu-based hosting.	▪ Used for all survey campaigns executed in the Oracle Applications Framework technology stack. ▪ Supports applications guest user login. This limits users to execution of the script in a Web browser only. ▪ Supports menu-based hosting for standard deployments. This requires an authenticated Oracle Applications session. In this case, the menus from the hosting application are shown in the UI when the script is executed.
OA.JSP	OAF	For targeted deployments, guest user only.	▪ Does not support menu-based hosting for *targeted* deployments. ▪ Uses applications guest user only.
IESSVYMAIN.JSP	JTT	Guest user only	▪ Uses applications guest user login. This limits users to execution of the script in a Web browser only. ▪ Used for JTT survey campaigns. ▪ Default for standard deployments.

JSP Template	Tech Stack	Authentication	Description
IESSVYMENUBASED.JSP	JTT	Uses authentication from existing Oracle Applications session only.	▪ Uses menu-based hosting for standard or targeted deployments. ▪ Targeted deployments for JTT survey campaigns are established by selecting Menubased in the Hosting Options (deployment details). ▪ This requires an authenticated Oracle Applications session. ▪ Used for JTT survey campaigns. ▪ The menus from the hosting application are shown in the UI when the script is executed.

For More Information

For information on adding parameters to the survey URL to pass values into the Oracle Scripting blackboard for a Scripting Engine Web interface session, see the section Passing Parameters to the Web Interface in *Oracle Scripting Developer's Guide*.

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Concurrent Programs Supporting Survey Operations

1.11.12 Concurrent Programs Supporting Survey Operations

Oracle Scripting uses three seeded concurrent programs to support survey component operations. Concurrent programs are executed using Oracle Concurrent

Manager. Seeded concurrent programs can be scheduled to execute at certain times, or can be administered to execute in near-real time.

To schedule or execute concurrent programs supporting Oracle Scripting's survey component, you must access Oracle Forms-based applications with a user assigned the iSurvey User responsibility. This section describes in detail each concurrent program applicable to Oracle Scripting.

This section includes the following topics:

- SUBMIT GROUP FM REQUEST FROM IES Concurrent Program
- Summarize Survey Data Concurrent Program
- Update Deployment Status Concurrent Program

References

For information on executing concurrent programs or checking concurrent program logs, see Administering Concurrent Programs for Survey Execution.

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL

1.11.12.1 SUBMIT GROUP FM REQUEST FROM IES Concurrent Program

When you activate a targeted deployment, a concurrent request is generated which is scheduled to run at the deployment start date and time (specified when defining a deployment). If the deploy start date and time are in the past, the concurrent program executes immediately. The purpose of this initial execution of the concurrent program is to make a request to Oracle One-to-One Fulfillment to send out e-mail invitations to the target population specified by the list.

> **Note:** Immediately upon activating a deployment, when the Update Deployment page refreshes the corresponding Concurrent Request ID is visible at the bottom of the page. You can use this Concurrent Request ID to monitor the Fulfillment Request submission through the Oracle Forms-based interface associated with the iSurvey User responsibility.

If reminders are also established for this deployment, subsequent concurrent requests are also generated (one per reminder, based on the value entered into the Number of Reminders field) to execute at the specified interval (in days) as defined for the deployment. These requests will execute at the appropriate interval (start date and time, plus the number of days defined in the Reminder Interval In Days field). When the concurrent manager runs the request, it attempts to submit a fulfillment request for this survey deployment.

If the request is successful, a fulfillment request ID will be generated. This is visible from the Survey Administration console in the Deployment Detail page, immediately above the Concurrent Request ID.

After activating a deployment, wait a brief period and refresh the page to view the fulfillment request ID. If this ID is listed, the fulfillment request was successfully submitted. If not, the deployment enters the error status.

1.11.12.1.1 Troubleshooting If the fulfillment request is not successful, you have the following options:

- You can attempt to determine and resolve the problem, and reactivate the deployment.
- You can attempt to manually run this concurrent program.

Resolving and Reactivating the Deployment

When a request fails, the deployment status changes to error status, and the Activate button is again visible. Before you attempt to reactivate the survey deployment, check the concurrent program log for information about the specific error. (For more information, see Checking Concurrent Program Logs.)

- Fix the error, and then return to the Survey Administration console to the detail screen of the relevant deployment.
- Click **Activate** to reactivate the deployment.

If the fulfillment request ID appears following a page refresh, the problem is resolved.

Manually Executing the Concurrent Program

When you activate a deployment, the required parameters are automatically passed to the function. If executing manually, you must manually enter the required parameters.

There are two cases in which this concurrent program needs to submitted manually:

1. If the initial deployment fails.
2. If the reminder concurrent program fails.

Follow the procedure detailed in Scheduling or Executing Concurrent Programs, using the parameters for the SUBMIT GROUP FM REQUEST FROM IES concurrent program as listed in the table below.

Parameters 7 and 9 below require specific values only if the purpose for manually executing this request is the failure of a previously scheduled concurrent program for reminders. Initial requests (to send out invitations) do not require these parameters.

No.	Parameter	Meaning	Value
1	p_api_version	PL/SQL API constant required by Apps.	1
2	p_init_msg_list	Initializes a fresh message list prior to loading new messages. This parameter required by Apps, primarily for error tracking. Leaving this parameter null or entering the value FND_API.G_TRUE passes a value of **true** (yes, initialize the list); otherwise, pass FND_API.G_FALSE, to ensure the message list is not initialized.	Leave parameter null (this passes a **true** value)

No.	Parameter	Meaning	Value
3	p_commit	Tells the API whether or not to commit changes. If set to **false**, the concurrent manager commits the database transactions instead of the API. If set to **true**, the API commits the database transaction.	FND_API.G_FALSE
4	p_validation_level	Parameter required by Apps, to determine which validation level is required. Passing a value of 100 results in the API ensuring full validation by passing FND_API.G_VALID_LEVEL_FULL.	100
5	p_deployment_id	Deployment ID for survey campaign deployment	Determine the deployment ID and use it here
6	p_template_id	Template ID for the invitation or reminder template used in survey campaign deployment.	Determine the template ID and use it here. f manually submitting the request, if your initial request failed, pass the ID of the invitation template. If a subsequent reminder request failed, pass the template ID of the reminder template.
7	p_reminder_type	Variable to identify the request as a reminder. For initial request (invitation), null is passed. For reminder requests, the value REMINDER is passed.	If your initial request failed, leave the parameter null (there is no reminder for an invitation request). If a subsequent (reminder) request failed, type the value REMINDER.

No.	Parameter	Meaning	Value
8	p_user_id	User ID value derived from the Oracle Applications user account for the user submitting the concurrent request	Determine the USER_ID value for the appropriate Oracle Applications user and provide it here. You need the apps password for this procedure. Steps to obtain this value: ■ Log into Forms-based applications as a user with the System Administrator responsibility. ■ Locate the record for the appropriate user. ■ With focus on the User Name field, examine the field and variable values (Help > Diagnostics > Examine). Enter the apps password when prompted. ■ In the Block field, enter USER. ■ In the Field field, search for and select USER_NAME. ■ The number in the Value field is the value you must enter as the p_user_id parameter.
9	p_reminder_hst_id	If p_reminder_type parameter is set to reminder, this parameter needs to be set. Otherwise, leave this value null. This is the reminder history ID created by the system when the original deployment is submitted.	If initial deployment concurrent program fails, leave parameter null. If your initial request failed, leave the parameter null (not required for invitation). If a subsequent (reminder) request failed, derive this parameter value by using the following SQL query against the database for your Oracle Scripting environment: `select survey_reminder_hst_id from ies_svy_reminder_hst_v , ies_svy_reminders_v where ies_svy_reminders_v.deployment_id = <p_deployment_id> and ies_svy_reminder_hst_v.survey_reminder_id = ies_svy_reminders_v.survey_reminder_id.`

Guidance

There are no prerequisites for executing this concurrent program. However, in order for this program to have any useful effect:

- The Survey component of Oracle Scripting must be fully implemented.
- Oracle One-to-One Fulfillment must be fully implemented and configured, and a Fulfillment server must be functional.
- A survey campaign and cycle must already exist, and a targeted deployment already defined.
- The associated list must be valid and accessible to the system.
- A targeted deployment must be active.
- The deployment start date and time, as compared to SYSDATE, must be in the past.
- Fulfillment request status is visible in the Survey Administration console in the deployment detail. However, if you need to log into the Fulfillment Administration console for detailed troubleshooting, you must log into Oracle HTML-based applications with a user that has the One-to-One Fulfillment Administrator responsibility.

References

- For more information on implementing, administering, or using Oracle One-to-One Fulfillment, refer to *Oracle One-to-One Fulfillment Implementation Guide*.
- For specific details on executing or scheduling concurrent programs, please refer to Chapter 5 of *Oracle System Administrator's Guide*.

See Also

- Summarize Survey Data Concurrent Program
- Update Deployment Status Concurrent Program

1.11.12.2 Summarize Survey Data Concurrent Program

When interaction center agents execute a script, or when survey respondents participate in a survey, their individual answers are collected in the Oracle Scripting schema of the applications database.

For generating reports on survey data, information must be moved into summary tables that make the data accessible to reporting tools (Oracle Discoverer workbooks) as part of the Interaction Center Intelligence product family.

Run this concurrent program prior to viewing any reports for the most current data, or schedule this program to execute on a regularly scheduled basis (for example, once daily at an off-peak time, when load on the system is not high).

Parameters

There is a single parameter for the Summarize Survey Data concurrent program. The parameter details are listed in the table below.

No.	Parameter	Meaning	Value
1	p_cycle_id	Cycle ID of the specified cycle for which you want to execute this concurrent program.	Locate the appropriate cycle ID from the p_cycle_id list of values and use it here.

Guidance

There are no prerequisites for executing this concurrent program. However, in order for this program to have any useful effect:

- The Survey component of Oracle Scripting must be fully implemented.
- A survey campaign must already have been created and its children objects (survey cycle and survey deployment) defined.
- At least one deployment must be active.
- For survey reports, respondents must have participated in the survey. In this way, data is available in the IES schema of the Applications database to summarize for reporting purposes.
- It is not necessary to run this concurrent program to obtain current data for footprinting reports.

See Also

- SUBMIT GROUP FM REQUEST FROM IES Concurrent Program
- Update Deployment Status Concurrent Program

1.11.12.3 Update Deployment Status Concurrent Program

When you define a survey deployment, one set of criteria includes the range of time for which the deployment is valid. This set includes the deployment start date, and the response end date and time.

A concurrent program defined in Oracle Applications for survey operations must be executed on a regular basis to check this response end date and time. This process occurs from the Concurrent Manager and is performed by an individual with the iSurvey User responsibility (typically an administrator or supervisor).

The Update Deployment Status concurrent program compares the response end date for each deployment against SYSDATE, and changes the status of all deployments from active to closed if SYSDATE is past the response end date and time. Additionally, this concurrent program changes the higher level survey campaign status to idle if the survey campaign contains no active or pending deployments.

Oracle Corporation recommends executing this concurrent program once daily at an off-peak time (a time when load on the system is not high). This concurrent program can be manually executed. For example, executing this concurrent program could be one of the closing operations an interaction center supervisor performs at the end of the day. Alternatively, this program can be scheduled to run automatically, for example, daily at 2 AM.

Parameters

There are no configurable parameters for the Update Deployment Status concurrent program.

Guidance

There are no prerequisites for executing this concurrent program. However, in order for this program to have any useful effect:

- The Survey component of Oracle Scripting must be fully implemented.
- A survey campaign must already exist and its children objects (survey cycle and survey deployment) already defined.
- At least one deployment must be active.
- The deployment end date and time, as defined in the deployment detail and compared to SYSDATE, must be in the past.

See Also

- SUBMIT GROUP FM REQUEST FROM IES Concurrent Program
- Summarize Survey Data Concurrent Program

2 Introduction

2.1 Oracle Interaction Center Overview

Every customer interaction (a telephone call, an e-mail message, or a Web chat conversation) presents an opportunity to win new business or improve customer satisfaction. The Oracle Interaction Center supports the management and processing of customer relationship activity across all channels of customer contact.

The Oracle Interaction Center integrates with several customer relationship business applications in the Oracle E-Business Suite. The Oracle Interaction Center consists of several modules. The modules relating to inbound telephony and outbound telephony are bundled separately.

The Oracle Interaction Center allows access to centralized customer information and business application functionality. Oracle Interaction Center integrates with front office applications (known as Customer Relationship Management or CRM), and back office applications (Enterprise Relationship Planning or ERP), thereby enabling a workflow powered, end-to-end strategic e-business solution.

The Oracle Interaction Center products include:

- Oracle Advanced Inbound Telephony
- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center
- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment

- Oracle Scripting
- Oracle Universal Work Queue

2.1.1 Oracle Advanced Inbound Telephony

Oracle Advanced Inbound Telephony is designed to consistently and effectively handle customer interactions by intelligently routing, queuing and distributing media items.

See Also

- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center
- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment
- Oracle Scripting
- Oracle Universal Work Queue

2.1.2 Oracle Advanced Outbound Telephony

Oracle Advanced Outbound Telephony is another key part of the Oracle E-Business Suite of applications. It is the module of Oracle Interaction Center that addresses outbound telephony. Oracle Advanced Outbound Telephony consists of two main components:

- A tactical list manager, which determines who to call and when to call them
- An outbound dialing engine, which dials numbers and transfers live contacts to call center agents Oracle Advanced Outbound Telephony integrates with and relies on Oracle Marketing to create campaigns and lists to execute.

Oracle Advanced Outbound Telephony serves as the execution arm for these marketing lists to maximize both outbound list penetration and agent productivity.

Oracle Advanced Outbound Telephony also integrates with desktop applications like Oracle TeleSales and Oracle Collections to handle the actual customer interactions. Oracle Advanced Outbound Telephony can be used any time agents need to contact parties via the telephone. Oracle Advanced Outbound Telephony also integrates with Oracle Customer Interaction History to provide feedback that marketers can use to analyze and measure the success of the marketing campaign, thereby providing a closed-loop marketing process.

Oracle Advanced Outbound Telephony does not include any other telephony management modules, and thus requires the use of Oracle Advanced Inbound Telephony.

See Also

- Oracle Advanced Inbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center
- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment
- Oracle Scripting
- Oracle Universal Work Queue

2.1.3 Oracle Customer Interaction History

Oracle Customer Interaction History provides applications with a common framework for capturing and accessing all "interaction" data associated with customer contacts. Oracle Customer Interaction History acts as a central repository and provides a consistent framework for tracking all automated or agent-based customer interactions.

Applications record interactions through the Oracle Customer Interaction History framework itself, or through other applications that use Oracle Customer Interaction History. This information can be accessed by using the Oracle Customer Interaction History user interface, by using the user interface of an application that is integrated with Oracle Customer Interaction History, or by calling the Oracle Customer Interaction History APIs.

See Also

- Oracle Advanced Inbound Telephony
- Oracle Advanced Outbound Telephony
- Oracle Email Center
- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment
- Oracle Scripting
- Oracle Universal Work Queue

2.1.4 Oracle Email Center

Oracle Email Center is a comprehensive solution for managing high volumes of inbound and outbound e-mails. Oracle Email Center reduces the cost per e-mail interaction by automatically replying to certain e-mail inquiries, as well as routing others to a skilled set of agents and providing them with a full featured console with cross application functionality.

Oracle Email Center increases customer satisfaction and reduces customer attrition by providing quick, accurate and consistent responses. It also increases agent's efficiency through the use of a full featured, Email Center agent console thereby reducing agent turnover.

An easy to use graphical user interface called the Self Service Administration console enables the administrator to configure the system, define processing rules and publish business data, while the Supervisor console enables the supervisor to manage e-mail queues and balance workload. In addition, Oracle Interaction Center Intelligence provides a comprehensive set of reports that enable directors and managers to track e-mail activity and relate it to business events. Through its integration with other Oracle E-Business applications, Oracle Email Center provides its agents with cross-application functionality and at the same time provides a utility called Message Component to the business applications for viewing, composing and responding to e-mail.

See Also

- Oracle Advanced Inbound Telephony

- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment
- Oracle Scripting
- Oracle Universal Work Queue

2.1.5 Oracle Interaction Blending

Oracle Interaction Blending is a server in an interaction center server group that delivers media (inbound telephony, web callbacks, and outbound telephony) to agents through the Oracle Universal Work Queue work selector according to service levels set in a service plan.

Service levels define the goals for servicing contact channels. Service plans specify the desired level of service for a particular time of day. Service levels are assigned to time intervals in a service plan.

See Also

- Oracle Advanced Inbound Telephony
- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment
- Oracle Scripting
- Oracle Universal Work Queue

2.1.6 Oracle Interaction Center Intelligence

Oracle Interaction Center Intelligence is a Web-based reporting solution that provides intelligent reports that facilitate day-to-day operational and long-term strategic decisions.

The data is presented to the user in a easy-to-use portal format. This format gives the user a unified, role-based, easily customized view of Interaction Center information, including Oracle Universal Work Queue information, key performance measures relating to agent productivity, speed to answer, and abandon rate.

The product is built on an Oracle proprietary Java-based technology stack (Oracle CRM Foundation, sometimes referred to as Java Technology Framework or JTF).Users of Oracle Interaction Center Intelligence require minimal training, and no additional software is needed on the user's machine other than a Web browser.

Oracle Interaction Center Intelligence is based on a three-tier architecture:

- The front end (client) using an Oracle Applications 11*i*-certified Web browser.
- The middle tier, which contains the Apache Web server and application server, included as part of the installation of Oracle Applications release 11*i*.
- The database tier, using an Oracle8*i*, Oracle9*i* or Oracle Database 10g database.

See Also

- Oracle Advanced Inbound Telephony
- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center
- Oracle Interaction Blending
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment
- Oracle Scripting
- Oracle Universal Work Queue

2.1.7 Oracle Interaction Center Server Manager

Oracle Interaction Center Server Manager is a set of shell scripts and batch files that start, stop and monitor the server processes in an interaction center server group.

This Java server process is required to be installed on each target machine that runs interaction center servers in the Oracle Interaction Center software suite.

See Also

- Oracle Advanced Inbound Telephony
- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center
- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle One-to-One Fulfillment
- Oracle Scripting
- Oracle Universal Work Queue

2.1.8 Oracle One-to-One Fulfillment

Oracle One-to-One Fulfillment is a framework for compiling and distributing fulfillment information to customers in electronic format. Fulfillment information includes generic collateral or personalized cover letters sent to customers or prospective customers.

See Also

- Oracle Advanced Inbound Telephony
- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center
- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle Scripting
- Oracle Universal Work Queue

2.1.9 Oracle Scripting

Oracle Scripting provides enterprises with scripts which guide customers, agents and employees through decision flows based on a series of questions and answers. Scripts can be used for a variety of purposes: as traditional interaction center scripts that guide agents through collection of information and provide proactive alerts; as Web-based surveys for customers, prospects, and employees; and as self-service Web scripts that integrate with enterprise Web pages and provide a mechanism to guide Web customers through decision processes. Oracle Scripting is composed of several components: Script Author, the Scripting Engine, the Scripting Administration console, and the Survey Administration console.

Script Author is the development tool with which customized business requirements are translated into miniature programs known as "scripts." Each implementation of Oracle Scripting employs at least one customized script built by Oracle Consulting, consulting partners, or the enterprise. There are various ways in which scripts can be employed to gather or distribute data for an enterprise. For example, a script can serve to unify an agent's desktop by integrating aspects of various applications, or as a survey questionnaire to solicit specific information from the sample or target population. Script Author offers two ways to create a script; a graphical layout tool and a Script Wizard component.

The Scripting Engine is responsible for displaying the script to the end user, interpreting the end user's responses to questions and answers, and processing custom code developed in support of the script. The Scripting Engine includes two interfaces (one for agents, and one for executing a script using a Web browser). Any script executed in the Web interface requires survey campaign administration.

The Scripting Administration console provides the user interface with which script developers can launch Script Author as a Java applet, and script administrators can administer Oracle Scripting files, as well as generate, view and analyze a panel footprint report.

The Survey Administration console provides the user interface with which survey administrators establish and maintain survey campaign information, define and manage survey deployments, and view responses from data received.

See Also

- Oracle Advanced Inbound Telephony
- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center

- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment
- Oracle Universal Work Queue

2.1.10 Oracle Universal Work Queue

Oracle Universal Work Queue provides one click access to work. It brings media queues, prioritized application work queues, work items and messages together in a single access point.

Applications integrate with Oracle Universal Work Queue to enable customer channels and to centralize access to application work items. Customer channels deliver media (inbound telephone calls, outbound telephone calls, web callbacks, web collaboration requests and e-mail) to applications through Oracle Universal Work Queue.

Applications generate work items (e.g., service requests, leads, opportunities, tasks) and distribute them through assignment, queues and work pools.

Oracle Universal Work Queue provides the unifying link between media queues, prioritized application work queues, application work items and Oracle E-Business applications.

See Also

- Oracle Advanced Inbound Telephony
- Oracle Advanced Outbound Telephony
- Oracle Customer Interaction History
- Oracle Email Center
- Oracle Interaction Blending
- Oracle Interaction Center Intelligence
- Oracle Interaction Center Server Manager
- Oracle One-to-One Fulfillment
- Oracle Scripting

3 Before You Begin

This section includes the following topics:

- Related Product Documentation
- Dependencies
- Minimum Software Requirements
- Minimum Hardware Requirements
- Scalability and Performance Guidelines
- Installing Oracle Scripting
- Accessing Oracle Scripting

3.1 Related Product Documentation

You can choose from many sources of information, including online documentation, training, and support services, to increase your knowledge and understanding of Oracle Scripting.

If this guide refers you to other Oracle Applications documentation, use only the Release 11*i* versions of those guides.

Online Documentation

All Oracle Applications documentation is available online (HTML or PDF). Some online help patches are available on Oracle*MetaLink*. Oracle Scripting online help is only available with installation of Oracle Scripting release 11.5.6 or later.

Related Documentation

Oracle Scripting shares business and setup information with other Oracle Applications products. Therefore, you may want to refer to other product documentation when you set up and use Oracle Scripting.

You can read the documents online by choosing Library from the expandable menu on your HTML help window, by reading from the Oracle Applications Document Library CD included in your media pack, or by using a Web browser with a URL that your system administrator provides.

If you require printed guides, you can purchase them from the Oracle Store at http://oraclestore.oracle.com.

Documents Related to All Products

Oracle Applications User's Guide

This guide explains how to enter data, query, run reports, and navigate using the graphical user interface (GUI) available with this release of Oracle Scripting (and any other Oracle Applications products). This guide also includes information on setting user profiles, as well as running and reviewing reports and concurrent processes.

You can access this user's guide online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

Documents Related to This Product

Oracle Scripting Implementation Guide 11*i*

This guide describes how to implement Oracle Scripting components and test the implementation appropriately. Administration information includes one-time implementation-specific tasks, script-dependent file administration, survey campaign and survey resource administration. Other administration topics include Oracle Marketing campaign and list administration in support of Oracle Scripting, and Oracle One-to-One Fulfillment administration in support of targeted survey campaigns.

Oracle Scripting User Guide

This guide includes information on understanding each Oracle Scripting component and planning Oracle Scripting projects. This guide also includes task-based steps for using Script Author, and describes use of the Scripting Engine.

Oracle Scripting Developer's Guide

This guide is for developers of Script Author scripts and developers of code integrating with scripts. It includes a section explaining how and where to use Script Author commands. Also detailed is information on customizing Oracle Scripting with reusable seeded commands, best practice Java methods and survey scripts, and building block components supporting Oracle Scripting. Specific guidance for customizing panel layouts is also detailed, along with troubleshooting information.

Installation and System Administration

Oracle Applications Concepts

This guide provides an introduction to the concepts, features, technology stack, architecture, and terminology for Oracle Applications Release 11*i*. It provides a useful first book to read before an installation of Oracle Applications. This guide also introduces the concepts behind Applications-wide features such as Business Intelligence (BIS), languages and character sets, and Self-Service Web Applications.

Installing Oracle Applications

This guide provides instructions for managing the installation of Oracle Applications products. In Release 11*i*, much of the installation process is handled using Oracle Rapid Install, which minimizes the time to install Oracle Applications, the Oracle8 technology stack, and the Oracle8*i* Server technology stack by automating many of the required steps. This guide contains instructions for using Oracle Rapid Install and lists the tasks you need to perform to finish your installation. You should use this guide in conjunction with individual product user's guides and implementation guides.

Oracle Applications Supplemental CRM Installation Steps

This guide contains specific steps needed to complete installation of a few of the CRM products. The steps should be done immediately following that tasks given in the Installing Oracle Applications guide.

Upgrading Oracle Applications

Refer to this guide if you are upgrading your Oracle Applications Release 10.7 or Release 11.0 products to Release 11*i*. This guide describes the upgrade process and lists database and product-specific upgrade tasks. You must be either at Release 10.7 (NCA, SmartClient, or character mode) or Release 11.0, to upgrade to Release 11*i*. You cannot upgrade to Release 11*i* directly from releases prior to 10.7.

Maintaining Oracle Applications

Use this guide to help you run the various AD utilities, such as AutoUpgrade, AutoPatch, AD Administration, AD Controller, AD Relink, License Manager, and others. It contains how-to steps, screenshots, and other information that you need to run the AD utilities. This guide also provides information on maintaining the Oracle applications file system and database.

Oracle Applications System Administrator's Guide

This guide provides planning and reference information for the Oracle Applications System Administrator. It contains information on how to define security, customize menus and online help, and manage concurrent processing.

Oracle Alert User's Guide

This guide explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

Oracle Applications Developer's Guide

This guide contains the coding standards followed by the Oracle Applications development staff. It describes the Oracle Application Object Library components needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. It also provides information to help you build your custom Oracle Forms Developer 6*i* forms so that they integrate with Oracle Applications.

Oracle Applications User Interface Standards for Forms-Based Products

This guide contains the user interface (UI) standards followed by the Oracle Applications development staff. It describes the UI for the Oracle Applications products and how to apply this UI to the design of an application built by using Oracle Forms.

Other Implementation Documentation

Multiple Reporting Currencies in Oracle Applications

If you use the Multiple Reporting Currencies feature to record transactions in more than one currency, use this manual before implementing Oracle Scripting. This manual details additional steps and setup considerations for implementing Oracle Scripting with this feature.

Multiple Organizations in Oracle Applications

This guide describes how to set up and use Oracle Scripting with Oracle Applications' Multiple Organization support feature, so you can define and support different organization structures when running a single installation of Oracle Scripting.

Oracle Workflow Guide

This guide explains how to define new workflow business processes as well as customize existing Oracle Applications-embedded workflow processes.You also use this guide to complete the setup steps necessary for any Oracle Applications product that includes workflow-enabled processes.

Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Oracle Scripting implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This manual also provides information on creating custom reports on flexfields data.

Oracle eTechnical Reference Manuals

Each eTechnical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for a specific Oracle Applications product. This information helps you convert data from your existing applications, integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. Oracle eTRM is available on Oracle*MetaLink*

Oracle Manufacturing APIs and Open Interfaces Manual

This manual contains up-to-date information about integrating with other Oracle Manufacturing applications and with your other systems. This documentation includes APIs and open interfaces found in Oracle Manufacturing.

Oracle Order Management Suite APIs and Open Interfaces Manual

This manual contains up-to-date information about integrating with other Oracle Manufacturing applications and with your other systems. This documentation includes APIs and open interfaces found in Oracle Order Management Suite.

Oracle Applications Message Reference Manual

This manual describes Oracle Applications messages. This manual is available in HTML format on the documentation CD-ROM for Release 11i.

Oracle CRM Application Foundation Implementation Guide

Many CRM products use components from CRM Application Foundation. Use this guide to correctly implement CRM Application Foundation.

Training and Support

Training

Oracle offers training courses to help you and your staff master Oracle Scripting and reach full productivity quickly. You have a choice of educational environments. You can attend courses offered by Oracle University at any one of our many Education Centers, you can arrange for our trainers to teach at your facility, or you can use Oracle Learning Network (OLN), Oracle University's online education utility. In addition, Oracle training professionals can tailor standard courses or develop custom courses to meet your needs. For example, you may want to use your organization structure, terminology, and data as examples in a customized training session delivered at your own facility.

Support

From on-site support to central support, our team of experienced professionals provides the help and information you need to keep Oracle Scripting working for you. This team includes your Technical Representative, Account Manager, and Oracle’s large staff of consultants and support specialists with expertise in your business area, managing Oracle Application Server 10g, and your hardware and software environment.

Oracle*MetaLink*

Oracle*MetaLink* is your self-service support connection with web, telephone menu, and e-mail alternatives. Oracle supplies these technologies for your convenience, available 24 hours a day, 7 days a week. With Oracle*MetaLink*, you can obtain information and advice from technical libraries and forums, download patches, download the latest documentation, look at bug details, and create or update TARs. To use Oracle*MetaLink*, register at (http://metalink.oracle.com).

Alerts: You should check Oracle*MetaLink* alerts before you begin to install or upgrade any of your Oracle Applications. Navigate to the Alerts page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade/Alerts.

Self-Service Toolkit: You may also find information by navigating to the Self-Service Toolkit page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade.

See Also

- Dependencies
- Minimum Software Requirements
- Minimum Hardware Requirements
- Scalability and Performance Guidelines
- Installing Oracle Scripting
- Accessing Oracle Scripting

3.2 Dependencies

This section includes the following topics:

- Mandatory Dependencies
- Conditional Dependencies

See Also

- Related Product Documentation
- Minimum Software Requirements
- Minimum Hardware Requirements
- Scalability and Performance Guidelines
- Installing Oracle Scripting
- Accessing Oracle Scripting

3.2.1 Mandatory Dependencies

- Oracle Applications must first be installed using Oracle Applications release 11*i* Rapid Install or release 11*i* Platinum, followed by appropriate upgrades to release 11.5.9 or later.
- In order to execute scripts using Oracle Scripting 11.5.6 or later, scripts must first be compiled and deployed using Script Author 11.5.6 or later.
  - All deployed scripts will display in the Script Chooser, including any scripts deployed using previous versions of Oracle Scripting. However, scripts compiled in versions of Script Author prior to 11.5.6 will not execute.

Selecting a pre-11.5.6 script to run will result in a blank Scripting window with error messages (see troubleshooting).

  - If a script from previous versions of Script Author must be used, open the script using Script Author 11.5.6 or later, recompile, save the updated script, and redeploy to the appropriate applications database.

- For new implementations of Oracle Scripting, you must specify the value Apache Mid Tier/Servlet Architecture in the system profile IES : ARCHITECTURE TYPE in order to use the Scripting Engine agent application.
- You must use an Oracle Applications 11*i*-certified Web browser to launch Oracle Applications. For software certifications, see Oracle*MetaLink*.
- You must have a certified version of Oracle JInitiator to launch Oracle Applications. If not present, you should be able to download the appropriate version when you attempt to access an Oracle Applications URL. For certifications, see Oracle*MetaLink*.
- If using a UNIX server, you must identify an X Server as a display server to dynamically generate graphics using JFC/SWING and AWT, in order to view panel footprint reports accessible from the Reporting tab on the Scripting Administration console.

  See Setting Display Server for UNIX Environments for information on the setting a display server.

  This requirement no longer applies to UNIX server implementations in order for respondents to participate in a survey, or to use the Survey Administration console.

See Also

- Conditional Dependencies

3.2.2 Conditional Dependencies

The following conditional dependencies may apply, as appropriate:

- Script Author Dependencies
- Scripting Engine Agent User Interface Dependencies
- Survey Component Dependencies
- Java Development Kit Dependencies

See Also

- Mandatory Dependencies

3.2.2.1 Script Author Dependencies

If using custom Java code to write commands supporting the Scripting Engine in the agent interface, you must compile custom Java classes using a version of JDK that is compatible with the JRE used at runtime. For implementations using the Apache mid-tier architecture, this is the version of JDK used on the Apache Web server. For implementations using the caching architecture (supported for upgrades from release 11.5.4 only), this implies a version of JDK that is compatible with Oracle JInitiator on the agent client (at the same level or lower).

References

- Using Custom Java
- Java Development Kit Dependencies

See Also

- Scripting Engine Agent User Interface Dependencies
- Survey Component Dependencies
- Java Development Kit Dependencies

3.2.2.2 Scripting Engine Agent User Interface Dependencies

If referencing custom Java code in a script:

- The custom JAR must be appropriately packaged.
- You must compile custom Java classes using JDK 1.1.8.
- You must appropriately reference any custom Java methods in the script, and ensure the compiled and packaged code is available to the Scripting Engine at runtime.
  - When deploying custom Java using the Scripting Administration console, you must deploy scripts using the Script Author Java applet.
  - If storing your custom Java code on the applications server, you must reference the class path of any custom Java in the Apache Web server JSERV.PROPERTIES file.

References

- For information on packaging JAR files, see Packaging Java Bean Code Into a Java Archive File.
- For information on JDK 11.18 requirements, see Script Author release 11.5.6, see Compiling Custom Code Using JDK 1.1.8.
- For information on referencing the class path, see Adding Custom Java Class Path to JSERV.PROPERTIES File.

See Also

- Script Author Dependencies
- Survey Component Dependencies
- Java Development Kit Dependencies

3.2.2.3 Survey Component Dependencies

Targeted Survey Deployment Dependencies

If your implementation requires the use of targeted survey deployments, the following are dependencies:

- You must have Oracle One-to-One Fulfillment installed and implemented.
- You must have an appropriately configured fulfillment server.
- You must have Oracle Marketing installed and implemented.
- You must have access to a user with the Survey Administrator responsibility.
- In order to use HTTPS, you must have proxy servers appropriately configured, and you must set the appropriate system profiles (IES: Proxy Server Name, IES: Proxy Server Port and Apps Servlet Agent).

Self-Service Web Application Customization Dependencies

If your implementation requires the execution of Web scripts (scripts executed in a Web browser using the Scripting Engine Web interface from a self-service Web application such as Oracle iSupport), the following are dependencies:

- You must customize the self-service Web application to list survey URLs from existing valid non-list-based survey deployments.
- The customized page listing survey URL inks is recommended to be designated as the final page survey resource; at the completion of script execution in an

Oracle Applications 11*i*-certified web browser, the user is returned to the page from which they launched the Web script.

- You must use a hosted survey error page such as the seeded sample JSP for this purpose entitled IESSVYMENUBASEDTESTERROR.JSP.

References

- For information on Oracle One-to-One Fulfillment requirements, use, and implementation, see *Oracle One-to-One Fulfillment Implementation Guide*.
- For information on Oracle Marketing requirements and implementation, see *Oracle Marketing Implementation Guide*.
- For information on using and administering Oracle Marketing, see *Oracle Marketing Concepts and Procedures*.
- For information on the appropriate system profiles for using HTTPS over a proxy server, see Implementing the Scripting Engine Agent Interface.
- For information on customizing self-service Web applications for use with Oracle Scripting, see Oracle Self-Service Web Applications.

See Also

- Script Author Dependencies
- Scripting Engine Agent User Interface Dependencies
- Java Development Kit Dependencies

3.2.2.4 Java Development Kit Dependencies

This section includes the following topics:

- Compiling Custom Code Using JDK 1.1.8
- Packaging Java Bean Code Into a Java Archive File

See Also

- Script Author Dependencies
- Scripting Engine Agent User Interface Dependencies
- Survey Component Dependencies

3.2.2.4.1 Compiling Custom Code Using JDK 1.1.8 Oracle Scripting information renders in HTML using different HTML rendering engines. For scripts executed in HTML as a survey, the Oracle Applications 11*i*-certified Web browser provides HTML rendering and interpretation. Various browsers, and various versions of the same browser, may look substantially different from release to release.

For all Oracle Applications 11*i* releases to date, including release 11.5.9, scripts running in the Scripting Engine agent interface use the Java classes in the SWING library of JDK 1.1.8 to render the HTML.

For this reason, Oracle Corporation recommends:

1. Testing scripts in the lowest common denominator (the Scripting agent interface) to ensure expected interpretation of HTML elements.
2. Compiling custom Java code using JDK 1.1.8 to ensure compatibility for custom code when executing scripts in the Scripting Engine agent interface.
3. For Scripting implementations using the Caching Architecture, the Java Virtual Machine for executing Scripting is provided by Oracle JInitiator, which at the time of this writing is not supported with Oracle Scripting at versions above JDK 1.1.8.x. Thus, *all custom code* to be executed in this architecture *must be compiled using JDK 1.1.8* to ensure compatibility. This requirement will relax if JInitiator versions compiled against higher JDK versions than 1.1.8 are released and certified by Oracle Corporation.

   For this and other certification information, refer to Oracle*MetaLink* or contact your Oracle Support Services representative.

See Also

- Packaging Java Bean Code Into a Java Archive File

3.2.2.4.2 Packaging Java Bean Code Into a Java Archive File

For more information, see Packaging Java Bean or Custom Java Code Into a JAR File in *Oracle Scripting User's Guide*.

Oracle Scripting 11*i* interprets custom Java code when appropriately referenced in a script using a Script Author command. The two common uses of custom Java are to substitute a custom Java bean user interface in place of a standard Scripting panel (in the agent interface only), or (for any Scripting Engine interface) to execute commands in the script at runtime.

In order for a script referencing custom Java (a Java bean or a Java method) to supply that code to the Scripting Engine user interface at runtime, the Java code must first be written, tested, compiled, and finally packaged into a Java Archive (JAR) or WinZip (ZIP) file. Oracle Corporation recommends using JAR file format.

Due to a limitation in JDK 1.1.8, there are only two supported methods of packaging a JAR file appropriately. Use either of the following methods to appropriately package custom Java code for execution at runtime.

The two methods are:

1. Use the command-line "jar" utility from JDK 1.1.8 and specify no compression. For example:

   ```
   jar -cf0 TestBean.jar ...).
   ```

   For more information, refer to JDK documentation.

2. Create the custom JAR file in standard .ZIP or .JAR format (with any PC-standard compression utility) with or without compression. If that utility saves the resulting archive in .ZIP format, then simply rename the file extension from .ZIP to .JAR.

   This method of packaging a JAR file may differ based on the compression utility used. For specifics, see the compression utility manufacturer's documentation.

See Also

- Compiling Custom Code Using JDK 1.1.8

3.3 Minimum Software Requirements

The following are the minimum software requirements for using Oracle Scripting release 11.5.9:

- You must have Oracle Applications release 11*i* installed or updated to the appropriate patch level.
- You must have Oracle Scripting 11.5.6 or higher (IES MiniPack L or later) to execute scripts in Scripting Engine releases 11.5.6 and higher.
- You must have an Oracle Applications 11*i*-certified Web browser.
- You must have Oracle Scripting release 11.5.8 (Interaction Center Family Pack P) or later to deploy custom Java to the database using the Scripting Administration console. This is a requirement for replacing a script panel with a user interface Java bean at runtime and can be used to load Java for all custom Java methods.

- You must have custom Java compiled using a version of JDK that is compatible with the JRE used on the Apache Web server (the same level or lower) to execute custom Java using the Apache mid-tier architecture of Oracle Scripting. At this time, appropriate JDK versions may include JDK 1.3, 1.2, or 1.1.8.
- You must have custom Java compiled using a version of JDK that is compatible with Oracle JInitiator (the same level or lower) on the agent client to execute custom Java using the caching architecture of Oracle Scripting. At this time, appropriate versions may include 1.3 or 1.1.8.
- You must have custom Java compiled using a version of JDK that is compatible with Oracle JInitiator (the same level or lower) on the agent client to replace a script panel with a user interface Java bean at runtime. At this time, appropriate versions may include 1.3 or 1.1.8.

Certification information is constantly changing. For the latest information on software certifications or compatibility, refer to Oracle*MetaLink*.

See Also

- Related Product Documentation
- Dependencies
- Minimum Hardware Requirements
- Scalability and Performance Guidelines
- Installing Oracle Scripting
- Accessing Oracle Scripting

3.4 Minimum Hardware Requirements

Oracle Applications must be installed (using a Rapid Install) on an appropriate server-class UNIX or Windows NT server. The server must have suitable memory (RAM) and storage (hard drive space) for enterprise-specific tasks, as determined by enterprise system administrators. Guidance can be found for minimum requirements in *Installing Oracle Applications*.

See Also

- Related Product Documentation
- Dependencies
- Minimum Software Requirements

- Scalability and Performance Guidelines
- Installing Oracle Scripting
- Accessing Oracle Scripting

3.5 Scalability and Performance Guidelines

Oracle Scripting has the same scalability and performance guidelines as other Oracle Applications CRM release 11*i* products. For other scalability and performance guidelines, refer to *Installing Oracle Applications*.

Some implementations install the database tier of Oracle Applications on a separate physical host machine to improve performance. Technically, each tier in a Rapid Install may be installed on a separate host machine for additional performance gains, although this is a rare occurrence. Detectable performance gains for multi-server implementations are likely if load on the applications database is heavy. For medium to light load there should be little appreciable gain. Regardless of whether you have a single or multiple configuration, Rapid Install is a prerequisite. Consult with your systems administrator for more information.

Installation and Dependency Verification

You must verify an implementation using the procedures described in the Implementation Verification section. This includes:

For All Implementations

- Obtaining the regression test script from Oracle*MetaLink* to perform Scripting Engine testing in standalone mode.
- Opening, compiling, and deploying the regression test script to your applications database.
- Uploading custom JAR file to the database using the Scripting Administration console.
- For testing scripts in the agent interface, assigning the Scripting User or Scripting Agent responsibility to an existing or new Oracle Applications user.
- Successfully launching and executing the regression test script in the Scripting Engine.

For Survey Implementations

- Assigning the Survey Administrator responsibility to an existing or new Oracle Applications user.
- Creating or modifying survey resources and ensuring they are present in the $OA_HTML directory on the server.
- Defining survey resources, a survey campaign, cycle, and deployment in the Survey Administration console.
- Activating a deployment.
- Ensuring you can participate in a survey using a Web browser.
- If viewing responses, ensure answer collection is enabled in the appropriate script to ensure survey response data is collected for viewing.
- Assigning the iSurvey User responsibility to an existing or new Oracle Applications user, to administer concurrent programs.
- Ensuring you can schedule or perform concurrent programs to update deployment status and move survey data to survey summary tables.
- If using reporting, perform the Oracle Discoverer implementation steps described in Oracle Discoverer Workbooks.

See Also

- Related Product Documentation
- Dependencies
- Minimum Software Requirements
- Minimum Hardware Requirements
- Installing Oracle Scripting
- Accessing Oracle Scripting

3.6 Installing Oracle Scripting

You have the following options for installing Oracle Scripting:

Note: Prior to installation or upgrade, *always* review the readme and patch list for the Oracle Interaction Center Family Pack. The readme contains installation instructions, a summary of product changes, and a list of known issues for a family pack. The patch list contains a list of patches that are required or recommended for use with a family pack. The patch list for each Oracle Interaction Center Family Pack is available on Oracle*MetaLink* at http://metalink.oracle.com. Each patch list contains a hyperlink to the readme.

- Oracle Applications Rapid Install

  The Rapid Install is intended for customers who are installing Oracle Applications for the first time or upgrading to Release 11*i* from to Release 11.0 or Release 10.7. It contains the family packs or product minipacks for all products in Oracle Applications.

  The Rapid Install is provided on CD-ROMs and is available from Oracle Store at http://oraclestore.oracle.com. For information about installing Oracle Applications using Rapid Install, see *Installing Oracle Applications*. For information about upgrading Oracle Applications using Rapid Install, see *Upgrading Oracle Applications*.

- Oracle Applications Maintenance Pack

  The Maintenance Pack is intended for customers who have already installed Oracle Applications Release 11*i*. It contains the family packs or product minipacks for all products in Oracle Applications.

  The Maintenance Pack is provided as a patch and is available on Oracle*MetaLink* at http://metalink.oracle.com. For information about upgrading Oracle Applications Release 11i using the Maintenance Pack, see Maintenance Pack Release Instructions on Oracle*MetaLink* at http://metalink.oracle.com. (Perform an advanced search for Document ID 232834.1. Enter the document ID in the Search Field and select the Doc ID option.)

  When installing an Oracle Applications Maintenance Pack, you must also check the patch list for the corresponding Oracle Interaction Center Family Pack, for any new issues that have emerged since the Maintenance Pack was released. The patch list for each Oracle Interaction Center Family Pack is available on Oracle*MetaLink* at http://metalink.oracle.com. (Perform an advanced search for

Document ID 219238.1. Enter the document ID in the Search Field and select the Doc ID option.)

- Oracle Interaction Center Family Pack

  The Oracle Interaction Center Family Packs are intended for customers who have already installed or upgraded to Oracle Applications Release 11*i* and wish to upgrade Oracle Interaction Center, or are upgrading another Oracle Applications product family, such as Oracle Service, which uses Oracle Interaction Center components, such as Universal Work Queue. The family pack is cumulative and contains only the minipacks for products in the Oracle Interaction Center family.

  > **Note:** Oracle Interaction Center products integrate with other products in Oracle Applications. Therefore, you may have to install family packs, product minipacks, or individual product patches for *other* products before installing the Oracle Interaction Center Family Pack.

The Oracle Interaction Center Family Pack is provided as a patch and is available on OracleMetaLink at http://metalink.oracle.com. The patch list for each Oracle Interaction Center Family Pack is available on OracleMetaLink at http://metalink.oracle.com. (Perform an advanced search for Document ID 232834.1. Enter the document ID in the Search Field and select the Doc ID option.)

See Also

- Related Product Documentation
- Dependencies
- Minimum Software Requirements
- Minimum Hardware Requirements
- Scalability and Performance Guidelines
- Accessing Oracle Scripting

3.7 Accessing Oracle Scripting

This section includes the following topics:

- Logins, Technology Stacks, and Single Sign-On Authentication

- User Accounts
- Responsibilities

See Also

- Related Product Documentation
- Dependencies
- Minimum Software Requirements
- Minimum Hardware Requirements
- Scalability and Performance Guidelines
- Installing Oracle Scripting
- Accessing Oracle Scripting

3.7.1 Logins, Technology Stacks, and Single Sign-On Authentication

Logins and Technology Stacks

Oracle Applications uses three different technology stacks:

- Oracle Application Object Library (AOL)
- Oracle Applications Framework (OAF)
- Java Technology Foundation (JTF or JTT)

AOL applications are Oracle Forms developed using Oracle Developer and are usually referred to as Forms-based applications.

OAF applications are HTML-based applications that use a common framework and a single JavaServer Pages (JSP) template called OA.JSP. This is the currently favored HTML technology stack.

JTT applications are applications using sets of JSP pages developed using Oracle JDeveloper. These are usually referred to as HTML-based applications. Many applications in this technology stack are migrating in the near future to OAF (as the Survey Administration Console did in release 11.5.9, as of Interaction Center Family Pack P).

Each type of application accesses the same database and can share information with the other.

The product interfaces are accessed by providing the Uniform Resource Locator (URL) for the environment in an Oracle Applications 11*i*-compatible Web browser and navigating to the hyperlink for the login page for the specific technology stack. You can also provide the URL for the specific login page. This URL is referred to as your login URL.

Single Sign-On URL See Single Sign-On below.

Oracle Applications URL Use this URL to navigate to the Personal Home Page (PHP) URL or the CRM Home page URL.

```
http://<host>:<port>/
```

- To navigate to the Personal Home Page URL, choose **Apps Logon Links** > **Personal Home Page**.
- To navigate to the CRM Home Page URL, choose **Apps Logon Links** > **CRM Home Page**.

CRM Home Page URL This URL is sometimes referred to as the Apache or JTF login URL. Use this URL to open the login page for HTML-based applications.

```
http://<host>:<port>/OA_HTML/jtflogin.jsp
```

Personal Home Page URL: This URL is sometimes referred to as the Self-Service Web Applications or SSWA login URL. Use this URL to open the login window for Oracle Applications from the Personal Home Page. You can access Forms-based or HTML-based applications from the Personal Home Page.

```
http://<host>:<port>/OA_HTML/US/ICXINDEX.htm
```

Forms URL Use this URL to open the login page for Forms-based applications. This login URL is typically used by system administrators, not end users. *Use of this login with the Scripting Engine agent interface is not supported and could cause errors or unexpected behavior.*

```
http://<host>:<port>/dev60cgi/f60cgi
```

Single Sign-On

An optional feature available in Oracle Applications for this release is the implementation of Single Sign-On (SSO). For all Oracle E-Business Suite Applications, if SSO is implemented, you can access a single login screen, provide a valid Oracle Applications user account with the appropriate responsibility assigned, and begin an Oracle Applications session.

If SSO is implemented, this is the only login available. SSO uses a specific authentication method enabling the logged-in user to access all functions associated with the user in any technology stack.

Oracle Scripting can access SSO, if implemented. As the single exception, if an Oracle Forms session timeout occurs during a script transaction in which a Forms command is invoked, then the user must re-login to Oracle Applications (using the personal homepage) before the Forms command will be able to execute.

See Also

- User Accounts
- Responsibilities

3.7.2 User Accounts

An application user is an authorized user of Oracle Applications and is uniquely identified by a username. After the user account has been defined, the application user can sign on to Oracle Applications at the CRM Home Page, Personal Home Page, or Forms login.

Note: Oracle Applications is installed with a system defined username and password.

- Username: sysadmin
- Password: sysadmin

An application user enters a username along with a password to sign on to Oracle Applications. The password assigned by the system administrator is temporary. When signing on for the first time, the application user will be prompted to change the password. Access to specific functionality and data will be determined by the responsibilities assigned to your user account.

See Also

- Logins, Technology Stacks, and Single Sign-On Authentication
- Responsibilities

3.7.3 Responsibilities

A system administrator assigns one or more responsibilities to an application user. A responsibility is a level of authority that allows a user to access specific functionality and data in Oracle Applications. Oracle Applications is installed with predefined responsibilities. A system administrator can modify a predefined responsibility or create custom responsibilities.

Various predefined responsibilities are required to implement Oracle Scripting, based on which components are required. These responsibilities are clearly defined in each relevant implementation section of this document.

In the Forms interface, if an application user has only one responsibility, then the related menu or application (if there is only one function in the menu) appears after the user signs on. If an application user has more than one responsibility, then a list of available responsibilities appears after the user signs on. To switch responsibilities, choose Switch Responsibility from the File menu.

In the HTML interface, an application user must select a default responsibility (even if the user has only one responsibility). The next time the application user signs on, the tabs related to the default responsibility appear. To switch responsibilities, go to Navigation Preferences in your profile (Profile icon). In the Switch Responsibilities section, select another responsibility from the Current Responsibility list.

See Also

- Logins, Technology Stacks, and Single Sign-On Authentication
- User Accounts

4 Implementation Tasks

This section describes all tasks that may be required to implement Oracle Scripting. This section includes the following topics:

- Implementation Task Sequence
- Understanding Users Required for Implementation
- Creating an Administrator for Oracle Scripting
- Creating an Agent
- Implementing the Scripting Engine Agent Interface
- Implementing the Scripting Engine Web Interface
- Implementing Administrative Interfaces

4.1 Implementation Task Sequence

You must have an understanding of which Oracle Scripting components, functions, and interfaces are required at the enterprise in order to implement successfully. Scripting components include Script Author, the Scripting Engine (which includes a Web interface and an agent interface), and two HTML user interfaces for administration: the Scripting Administration console and the Survey Administration console.

You must also have a clear definition of the users required for your implementation, based on the functions of Oracle Scripting required at the enterprise. All implementations require one or more administrative users which may have varying attributes. Required end users also vary; the Scripting Engine agent interface end user is an interaction center agent. Scripting Engine Web interface end users may be customers accessing scripts through a Web browser, or enterprise employees or self-service Web application customers accessing scripts through integrated Web applications such as Oracle iSupport or Oracle iStore. For detailed information, see Understanding Users Required for Implementation. Implementation requirements for each varies accordingly.

The table below indicates the tasks and sequence required to implement Oracle Scripting to meet your requirements.

Task	Task Description	Required for	Consult documented tasks...
1	Create and administer Oracle Scripting administrators.	All	Creating an Administrator for Oracle Scripting
2	Create and administer Oracle Scripting end users.	All	Creating an Agent
3	Implement the Scripting Engine agent interface.	Agent UI	Implementing the Scripting Engine Agent Interface
4	Implement the Scripting Engine Web interface.	Web UI	Implementing the Scripting Engine Web Interface
5	Implement the Oracle Scripting administrative interface.	All	Implementing the Scripting Engine Agent Interface
6	Implement the Survey administrative interface.	Web UI	Implementing Administrative Interfaces

See Also

- Understanding Users Required for Implementation
- Creating an Administrator for Oracle Scripting
- Creating an Agent
- Implementing the Scripting Engine Agent Interface
- Implementing the Scripting Engine Web Interface
- Implementing Administrative Interfaces

4.2 Understanding Users Required for Implementation

Prior to creating and administering users for Oracle Scripting, you must understand the types of users required, the applicable responsibilities to assign to the appropriate Oracle Applications login, and the specific tasks to accomplish.

From an implementation perspective, you need to create administrator users and agent users. The responsibilities that must be associated with each user type depends on how Oracle Scripting will be implemented at the enterprise, and which Oracle Applications functions will be required.

This section addresses which user account to use to create other users, and possible requirements for JTF roles for a specific user. It also describes a rationale for creating a super user for implementation purposes. This section then includes an implementation responsibility matrix, and a discussion regarding the approach to creating users for your Oracle Scripting implementation.

This section includes the following topics:

- SYSADMIN Versus Explicitly Created Administrator User Account
- JTF Role Requirements
- Administrator/Agent Super Users
- Implementation Responsibility Matrix
- Road Map to User Creation and Administration

References

- Additional specific information for creating administrators and for creating agents can be found, respectively, at Creating an Administrator for Oracle Scripting and Creating an Agent.
- Step-by-step procedures to accomplish user creation tasks are described in Implementation Administration Tasks > Creating and Administering Users.

See Also

- Implementation Task Sequence
- Creating an Administrator for Oracle Scripting
- Creating an Agent
- Implementing the Scripting Engine Agent Interface
- Implementing the Scripting Engine Web Interface
- Implementing Administrative Interfaces

4.2.1 SYSADMIN Versus Explicitly Created Administrator User Account

Implementors often access the seeded SYSADMIN Oracle Applications user account when administering or configuring Oracle Applications. For security purposes, some enterprises restrict access to this account. For this reason, this document assumes you will create (or have created for you) at least one distinct Oracle Applications user account, and provide this new account with all of the required roles, responsibilities and privileges required for implementation, administration, and testing. For planning purposes, if an enterprise has the SYSADMIN account locked down, you may need to request in advance that an administrator user account be created for you.

To create an administrator for Oracle Scripting, you must know all of the tasks the administrator will perform, and how Oracle Scripting will be used at the enterprise.

4.2.2 JTF Role Requirements

Some implementation or administration tasks require the administrator user to be granted JTF roles before you log into Oracle Applications.

For example:

- A Survey Administrator user must be granted the JTF_FM_ADMIN role in order to access the Invitations tab on the Survey Administration console, or to access any tab on the Fulfillment Administration console.
- If performing tasks involving modification of JTF advanced system properties (for example, removing the Quick Find menu from the Scripting Administration console), your Oracle Applications user account must be granted the JTF_SYSTEM_ADMIN_ROLE role.
- If using an administrator user to create other users with roles (such as the JTF_SYSTEM_ADMIN_ROLE or the JTF_FM_ADMIN role), the user account accessed to assign JTF roles to another user must already be granted those JTF roles.

If required, you can use the seeded SYSADMIN Oracle Applications user account. If you do not have access to this account, request that the specific JTF roles needed are assigned to your administrator user.

4.2.3 Administrator/Agent Super Users

For implementation purposes, the administrator may require only an Oracle Applications user account with the appropriate roles and responsibilities. However, your implementation may require you to define an administrator in the enterprise who is also a "superuser" (a user with administrator *and* agent privileges). A super user can be used to administer users and system profiles, but also can be used to perform work in CRM applications. For example, if you want to launch a script from Oracle TeleSales to verify your implementation, you must associate this Oracle

Applications account with an employee record in the enterprise database, and import the employee as a CRM resource. If this superuser must also perform campaign and list management using Oracle Marketing, then the CRM resource associated with this user must also be associated with a resource group with the appropriate roles and usages. For more information, see Creating an Agent.

Review the information below to determine which requirements apply to your administrator user(s). Then define users, assign responsibilities, and otherwise perform all steps as applicable.

> **Note:** Some tasks (such as granting roles for list and fulfillment administration, or adding JTF system-level properties) require existing JTF roles such as JTF_SYSTEM_ADMIN_ROLE and JTF_FM_ADMIN role. Your Oracle Applications login must already be granted JTF roles in order to assign them or to administer advanced system-level properties. For this purpose, if required, you can use the seeded SYSADMIN Oracle Applications user account.

4.2.4 Implementation Responsibility Matrix

A matrix of responsibilities that may be required for various users is included below. The matrix includes user type (administrator or agent), responsibility name, the function provided by the responsibility, and the specific component that may require the responsibility.

Not all responsibilities listed in the matrix for each user type is necessarily required. For example, if Oracle Human Resource Management Systems is not installed, the HRMS Manager responsibility is not required. Similarly, for survey implementations, Oracle Marketing and Oracle One-to-One Fulfillment administration responsibilities are only required for list-based survey implementations.

Review the specific requirements for your implementation. Each implementation or administration task in this document lists the responsibility required to perform that task.

User Type	Responsibility	Function	For Component(s)
Administrator	System Administrator	■ Create and administer Oracle Applications users ■ Establish or modify system profile settings ■ Establish the link between Oracle Applications user and an employee in the database	All
Administrator	HRMS Manager	■ Create employees in the enterprise human resources database, if Oracle Human Resource Management Systems (HRMS) is licensed and installed at the enterprise.	All
Administrator	CRM Resource Manager	■ Create employees in the enterprise human resources database, if HRMS is not installed at the enterprise. ■ Import employees from the enterprise human resources database as CRM resources. ■ Create and manage resource groups.	All
Administrator	Scripting Administrator	Access the Scripting Administration console. This is required to: ■ Launch Script Author Java applet ■ Deploy scripts associated with an Oracle Applications user name, so the script can be mapped to specific custom Java archives ■ View or remove scripts deployed using the Script Author Java applet ■ View, upload, update, or remove custom Java archives to the database ■ Map custom Java archives to specific scripts ■ Categorize custom Java archives as global to apply to all deployed scripts ■ Generate and view panel footprint reports	All

User Type	Responsibility	Function	For Component(s)
Administrator	Survey Administrator	Access the Survey Administration console. This is required to: ■ Define survey resources (header and footer sections, final pages, and error pages in JSP format) ■ Create survey campaigns ■ Define survey cycles ■ Define or deploy survey deployments ■ View specific responses provided by survey respondents to a previously or currently active deployment	Survey
Administrator	iSurvey User	Schedule or run concurrent programs.	Survey
Administrator	One-to-One Fulfillment Administrator	■ Access the Fulfillment Administration console. ■ Set up or change Fulfillment groups ■ Set up or change mail servers associated with Fulfillment groups ■ View failed Fulfillment requests after concurrent programs succeed in passing requests to the Fulfillment server.	Survey These fulfillment administration tasks are required for implementations using targeted (list-based) survey campaigns
Administrator	Oracle Marketing Super User	■ Access the Marketing Administration console. ■ Establish campaigns ■ Assign scripts to agents or scripts to campaigns ■ Create, import, modify lists	Scripting Engine for agent application If Oracle TeleSales or Oracle Collections agents must launch a specific script, the script must be associated with a campaign or campaign schedule. Survey For list-based survey campaigns, lists must be created and administered.

User Type	Responsibility	Function	For Component(s)
Administrator	Sales Online SuperUser	■ Access the Sales Online Administration console to: ■ Assign agents to campaign schedules ■ Assign campaign schedules to agents	Scripting Engine for agent application If Oracle TeleSales or Oracle Collections agents must launch a specific script, the script must be associated with a campaign or campaign schedule.
Administrator	CRM HTML Administration	Administer Oracle CRM HTML applications, including verifying or adding Guest User JTF system-level properties so that survey respondents using the non-hosted IESSVYMAIN.JSP template can start an Oracle applications session.	Survey
Agent	Scripting User or Scripting Agent	Launch scripts in "standalone mode" from Oracle Forms, typically to test deployed scripts.	Scripting Engine for agent interface
Agent	Customer Support	For agents accessing Oracle Scripting from within the integrated Oracle TeleService application.	Scripting Engine for agent application
Agent	TeleSales Agent	For agents accessing Oracle Scripting from within the integrated Oracle TeleSales application.	Scripting Engine for agent application
Agent	Collections Agent	For agents accessing Oracle Scripting from within the integrated Oracle Collections application.	Scripting Engine for agent application

See Also

- Road Map to User Creation and Administration

4.2.5 Road Map to User Creation and Administration

Order and Timing of User Type Creation

1. Create and administer one or more administrator user with the appropriate responsibilities for your implementation.

   Administrator users must be created first, and are used to accomplish all other implementation and administration steps. Use the Implementation Responsibility Matrix above as a guideline.

Note that at least one administrator user requires the JTF_SYSTEM_ADMIN_ROLE to perform some implementation administration steps. Any administrator user that requires access to Oracle One-to-One Fulfillment also requires the JTF_FM_ADMIN role. Any user that must grant roles to other users must also already possess the role in question. For more information, see Granting JTF Roles.

2. Optionally, *c*reate and administer one or more agent user with the appropriate responsibilities for your implementation.

   The order in which agent users are created is at your discretion. You may choose to create Oracle Applications users for each agent immediately following the creation of an administrative user, assuming you have all information required. Alternatively, agent users can be created immediately prior to use of an agent login for administration or testing purposes.

   Use the Implementation Responsibility Matrix above as a guideline. Specific steps are detailed in *Oracle Interaction Center Server Manager Implementation Guide Release 11i*.

Method of Creating and Administering Users

1. The first task for user creation and administration is to create employees in the employee database.

   a. If HRMS is installed, you must use Oracle HRMS to create and administer employees. This requires the HRMS Manager responsibility.

   b. If HRMS is *not* installed, you must use Oracle CRM Resource Manager to create and administer employees. This requires the CRM Resource Manager responsibility.

2. After creating employees in the database, import each employees as a CRM resource. This requires the CRM Resource Manager responsibility.

   a. If resource group membership is required for the user, create and administer groups first. Ensure you provide the appropriate role types, roles, and usages to the group. If the appropriate group already exists and has the appropriate attributes, skip this step.

   b. After resource group creation (if applicable), import the employee as a CRM resource. Ensure you provide the appropriate role type and role for the employee. Employees do not have usages.

   c. If required, query the appropriate group and associate the employee from within the group. Verify the person has the appropriate role types and roles.

3. Define Oracle Applications users. This requires the System Administrator responsibility.

   a. First select the user name and password. Verify the password.

   b. Assign the appropriate responsibilities for this user. Use the Implementation Responsibility Matrix above as a guideline.

   c. If an employee in the database, associate this Oracle Applications user account with the employee in the enterprise database.

See Also

- Implementation Responsibility Matrix

4.3 Creating an Administrator for Oracle Scripting

For the purposes of Oracle Scripting implementation and use, administrators fall into three categories: system administrators, script administrators, and survey campaign administrators. The first two categories are required for all implementations. Survey administrators are required for any implementation in which scripts are intended to be executed in a Web browser. You can establish the same individual with all administration privileges, or you can create two or more administrative users.

System Administrators

System administrators require access to functions that the typical interaction center agent will not, and administer users and applications across the broad spectrum. For example, one function of a system administrator is to create agent users. This necessitates the creation of an administrative user as the first step.

Script Administrators

Script administrators access the Scripting Administration console, an HTML user interface that provides access to the Script Author Java applet, allows administration of Oracle Scripting files, and provides access to agent interface report generation and analysis. Thus, individuals who develop scripts using the Script Author component require access to this administrative UI.

Survey Administrators

Survey administrators access the Survey Administration console, an HTML user interface that provides the ability to establish survey campaigns and to administer targeted survey campaign deployments using Oracle One-to-One Fulfillment and

Oracle Marketing functionality. This console also provides access to view individual responses to a Web-enabled survey campaign. Thus, individuals who monitor campaign status for ongoing active survey campaigns or who analyze past campaigns require access to this administrative UI.

To successfully implement Oracle Scripting, you must create at least one administrator user. If you do not want one user account to have access to multiple Oracle applications, you can create several administrator users and assign each the appropriate responsibility.

Administrator users can be created by any Oracle Applications user account with the System Administrator responsibility, with one exception. If implementing the survey component, the administrator user must be granted the JTF_SYSTEM_ADMIN_ROLE and may require the JTF_FM_ADMIN role. Your Oracle Applications login must already be granted JTF roles in order to assign them. For this purpose, you can use the seeded SYSADMIN Oracle Applications user account.

For implementation purposes, the administrator may require only an Oracle Applications user account. However, if the administrator user is an enterprise super user that will perform work in CRM applications in addition to performing administrative tasks with the same login, or if the administrator user must access business applications for which campaign administration is required, you must perform all three user creation and administration tasks for the administrator.

Assign Administrative Responsibilities to Trusted Users Only

The Scripting Administrator and Survey Administrator responsibilities are required to implement Oracle Scripting and administer Oracle Scripting for use at an enterprise, as described below.

Survey administrators have the ability to modify the properties of active survey campaigns, including changing active survey campaign status (cancelling or closing the campaign). If performed inadvertently, this could destroy the integrity of survey data.

In the same manner, Scripting administrators can modify, deploy, and delete deployed scripts, delete custom Java, and change specific Java mappings. If performed inadvertently, any of these tasks could cause loss of data and script functionality. Additionally, scripts with SQL code can access sensitive database tables. Resulting information, if misused, can introduce liability issues for the enterprise.

For these reasons, Oracle Corporation strongly recommends that only trusted users be provided with Scripting Administrator or Survey Administrator responsibilities.

Scripting Administrators

Since Oracle Scripting implementations typically require a customized script, and since the new Script Author Java applet and custom Java archive deployment and mapping capabilities are only accessible from the Scripting Administration console, all new or upgraded implementations require a Scripting administrator. Therefore, all administrators should be assigned the Scripting Administrator responsibility. See Assigning Scripting Administrator Responsibilities to a User.

Survey Administrators

Implementations requiring execution of a script in a Web browser must have an administrator user who is assigned the Survey Administrator responsibility. See Assigning Survey-Related Responsibilities to a User.

Fulfillment and List Management Administration

Implementations requiring execution of targeted survey deployments require additional roles, responsibilities, and must meet or bypass sales group membership requirements. See the following sections:

- Assigning Fulfillment Administrator Role to Survey Administrator
- Granting JTF Roles
- Bypassing the Sales Resource Group Membership Requirement
- Administering Lists in Oracle Marketing

Consult the table below to determine which administrator user administration steps are required for your implementation. If your administrator needs agent functions, also consult Creating an Agent. Then perform all steps as applicable.

If administrator needs to do this...	Perform these steps...	Consult these documented tasks...
Create or modify Oracle Applications users	■ Assign System Administrator responsibility ■ Access Forms-based applications as a system administrator	Creating and Administering Users

If administrator needs to do this...	Perform these steps...	Consult these documented tasks...
Create or modify an employee in the enterprise database	■ Assign appropriate HRMS responsibility (based on country) or CRM Resource Manager responsibility (if HRMS not installed) ■ Access HRMS or CRM Resource Manager as appropriate	Creating an Employee in the Enterprise Database
Create a resource group in CRM Resource Manager	■ Assign CRM Resource Manager responsibility ■ Access CRM Resource Manager to create group and assign membership roles and members	Creating a Resource Group
Import an employee as a CRM resource	■ Assign CRM Resource Manager responsibility ■ Access CRM Resource Manager to import employee	Importing a CRM Resource
Access the Scripting Administration console to perform any of the following: ■ Launch the Script Author Java applet. ■ Create, modify, or deploy custom scripts using Script Author Java applet. ■ View or remove deployed scripts. ■ Deploy, update, or remove custom Java archives to the Applications database. ■ Define uploaded Java archives as global to apply to all deployed scripts. ■ Map uploaded Java archives to one or more specific scripts. ■ View and generate panel footprint reports.	■ Assign Scripting Administrator responsibility ■ Ensure prerequisite JTT system profiles are established.	Assigning Scripting Administrator Responsibilities to a User

If administrator needs to do this...	Perform these steps...	Consult these documented tasks...
Access the Survey Administration console to administer survey resources	■ Assign Survey Administrator responsibility ■ Provide FTP access to applications server to deploy JSP resources to $OA_HTML and images to $OA_MEDIA as appropriate	Assigning Survey-Related Responsibilities to a User
Access the Survey Administration console to perform any of the following: ■ Create survey campaigns and cycles ■ View or modify survey campaign details ■ Create additional cycles ■ Rename cycles ■ Define deployments ■ View or modify deployment details ■ Change or view deployment status	■ Assign Survey Administrator responsibility	Assigning Survey-Related Responsibilities to a User
Manage survey campaign responses, including: ■ View individual Web interface responses ■ Send reminders manually to list members	■ Assign Survey Administrator responsibility	Assigning Survey-Related Responsibilities to a User
Generate summarized Web interface reports or panel footprint reports	■ Assigning iSurvey User responsibility to access Concurrent Manager	Assigning iSurvey User Responsibility for Administering Concurrent Programs for Survey Execution

If administrator needs to do this...	Perform these steps...	Consult these documented tasks...
View, generate, or modify lists for list-based survey campaigns	- Assign Oracle Marketing Super User responsibility - Assign sales group membership and roles *or* bypass group membership requirements	- Assigning Oracle Marketing Responsibilities to a User - Assigning Sales Group Membership and Roles to a User *or* Bypassing the Sales Resource Group Membership Requirement - Administering Lists in Oracle Marketing
Access the Invitations tab from the Survey Administration console to administer fulfillment documents for list-based survey campaigns	- Assign JTF_SYSTEM_ADMIN_ROLE role - Assign JTF_FM_ADMIN role	- Assigning Fulfillment Administrator Role to Survey Administrator - Granting JTF Roles

See Also

- Implementation Task Sequence
- Understanding Users Required for Implementation
- Creating an Agent
- Implementing the Scripting Engine Agent Interface
- Implementing the Scripting Engine Web Interface
- Implementing Administrative Interfaces

4.4 Creating an Agent

Many CRM applications require users to be associated with CRM resource identification codes. This requires users to be defined as employees in the enterprise database, and to be imported as CRM resources. Oracle Corporation recommends that these steps be performed for all agent users as the first steps in agent creation.

You must define agent users that can log into an Oracle Applications session. Each agent user must be assigned the appropriate responsibilities to perform a particular set of functions. To define an agent for Oracle Scripting, you must know all tasks the agent will perform, and how Oracle Scripting will be used at the enterprise.

Additionally, applications that rely on campaign information (such as Oracle TeleSales, Oracle Collections, and Oracle Marketing) may require agents to have specific group membership and roles assigned to them.

Execution-level requirements also exist. For example, in order to launch a script from Oracle Collections, the script must be associated with a campaign. Thus, campaign management and administration is a prerequisite of using Oracle Scripting with Oracle Collections.

Review the information below to determine which requirements apply to your agent user(s). Then define users, assign responsibilities, and otherwise perform all steps as applicable.

The process of defining and administering users requires you to use an administrator user. For more information, see Creating an Administrator for Oracle Scripting.

Consult the table below to determine which agent user administration steps are required for your implementation. Then perform all steps as applicable.

If agent needs to do this...	Perform these steps...	Consult these tasks...
Use the Agent interface to execute scripts in standalone mode	■ Assign Scripting Agent or Scripting User responsibility	■ Defining Oracle Applications Users
Use the Agent interface to execute scripts from Customer Support component of Oracle TeleService	■ Assign Customer Support responsibility ■ Optionally, create a relationship plan	■ Defining Oracle Applications Users
Use the Agent interface to execute scripts from business applications requiring campaign information (Oracle TeleSales, Oracle Collections)	■ Assign Scripting Administrator responsibility ■ Access CRM Resource Manager to provide agent with Sales and TeleSales roles ■ Access CRM Resource Manager to add agent to sales group ■ Access Oracle Marketing to create a campaign and a campaign schedule, associate a script to a campaign schedule, assign agents to a campaign schedule or vice versa	■ Assigning Scripting Administrator Responsibilities to a User ■ Creating a Campaign ■ Creating a Campaign Schedule ■ Associating a Script to a Campaign Schedule ■ Associating a Script to a Campaign ■ Assigning Agents to a Campaign Schedule ■ Assigning Campaign Schedules to Agents

If agent needs to do this...	Perform these steps...	Consult these tasks...
Use the Web interface to execute scripts from a self-service Web application in a Web browser	■ Assign self-service Web application responsibility	■ Assigning Survey-Related Responsibilities to a User
Use the Web interface to execute surveys in a Web browser	■ Access an active survey URL	■ Scripting Engine Users
Use the Web interface to execute list-based surveys in a Web browser	■ Access an active survey URL from an e-mail message invitation or reminder	■ Scripting Engine Users

Oracle Scripting has varying requirements for administration, based on the components required for your implementation.

- Since Oracle Scripting implementations typically require a customized script, and since the new Script Author Java applet and custom Java archive deployment and mapping capabilities are only accessible from the Scripting Administration console, all new or upgraded implementations require a Scripting administrator.
- Implementations requiring execution of a script in a Web browser require a survey administrator.
- Implementations requiring execution of list-based surveys require additional roles, responsibilities, and group membership requirements.
- Implementations requiring customization of self-service Web applications need administrators of the appropriate application to customize the JSP pages to include survey URL links.
- Implementations using Oracle Applications on the UNIX platform have additional configuration steps based on platform and patch level.
- Implementations requiring multiple Apache listeners require Apache Web server administrators intimately familiar with Oracle Applications administration.

This section includes the following topics:

- Assigning Scripting Administrator Responsibilities to a User
- Assigning Survey-Related Responsibilities to a User

See Also

- Implementation Task Sequence

- Understanding Users Required for Implementation
- Creating an Administrator for Oracle Scripting
- Implementing the Scripting Engine Agent Interface
- Implementing the Scripting Engine Web Interface
- Implementing Administrative Interfaces

4.4.1 Assigning Scripting Administrator Responsibilities to a User

The **Scripting Administrator** responsibility is required for a user to log into the Scripting Administration console, from which the Script Author Java applet can be launched, deployed scripts can be viewed and deleted, Java archives can be viewed, deleted, or mapped to deployed scripts, and agent interface reports can be generated and analyzed.

Use this procedure to create a Scripting administrator:

Assign the **Scripting Administrator** responsibility to users of the Scripting Administration console, as described in Defining Oracle Applications Users.

For testing purposes, you may want to use an agent user who can execute scripts after they are created and deployed. In this case, you will also want the **Scripting Agent, Vision Enterprises**, or **Scripting User** responsibility as well. Alternatively, you can provide the Scripting administrator with a responsibility to launch a business application such as Oracle TeleSales, Oracle Collections or the Customer Care component of Oracle TeleService.

Alternatively, if executing scripts in the Scripting Engine Web interface, you may want the same administrator user to have access to survey administration. In this case, follow all additional recommendations in Assigning Survey-Related Responsibilities to a User.

For a full list of responsibilities that may be required for Oracle Scripting, see Implementation Responsibility Matrix.

See Also

- Assigning Survey-Related Responsibilities to a User

4.4.2 Assigning Survey-Related Responsibilities to a User

Assign the appropriate Survey-related responsibilities to appropriate administrator users as described in Defining Oracle Applications Users.

Following is a summary of Oracle Applications responsibilities for using Survey:

- The **Survey Administrator** responsibility is required for a user to log into the Survey Administration console, from which survey campaigns are created, deployed, and managed.
- The **iSurvey User** responsibility is required to run Concurrent Manager for processing and scheduling of concurrent programs required to support survey operations.
- Optionally, you can assign the **One-to-One Fulfillment Administrator** responsibility, required to access Oracle One-to-One Fulfillment, to an Oracle Applications user with the Survey Administrator responsibility. In this way, the same user would have the ability to administer all aspects of Oracle One-to-One Fulfillment required for list-based survey operations.

> **Note:** While assigning the One-to-One Fulfillment Administrator to a survey administrator user is not required, logging into Oracle Applications as a fulfillment administrator *is* required in order to perform the implementation step described in Adding Survey Administrator User to a Fulfillment Group.

For a full list of responsibilities that may be required for Oracle Scripting, see Implementation Responsibility Matrix.

See Also

- Assigning Scripting Administrator Responsibilities to a User

4.5 Implementing the Scripting Engine Agent Interface

The implementation tasks detailed in this section are required only for Scripting Engine operations for the Java-based interaction center agent application.

In addition to these tasks, you will also need to set up the Scripting Administration console as described in Setting Up the Scripting Administrative Interface.

This section includes the following topics:

- Assigning Scripting Engine-Related Responsibilities to a User
- Establishing Profile Settings for the Agent Interface
- Setting Session Idle Timeout Values for Apache JServ

See Also

- Implementation Task Sequence
- Understanding Users Required for Implementation
- Creating an Administrator for Oracle Scripting
- Creating an Agent
- Implementing the Scripting Engine Web Interface
- Implementing Administrative Interfaces

4.5.1 Assigning Scripting Engine-Related Responsibilities to a User

Assign the appropriate Scripting Engine-related responsibilities to appropriate agent and administrator users as described in Defining Oracle Applications Users.

The agent interface for the Scripting Engine is a Java bean that runs in an Oracle form. A script can be executed in the agent application by any Oracle Applications user with the appropriate responsibility.

Scripts can be launched in the agent application either in standalone mode (recommended for testing), or from an integrated application. In either case, you must have the appropriate responsibility for that application assigned to your Oracle Applications login.

Following is a summary of Oracle Applications responsibilities for using the Scripting Engine in the agent application:

- The **Scripting User** or **Scripting Agent** responsibility is required for a user to launch scripts in standalone mode.
- The **Customer Support** responsibility is required to run the Oracle Support component of Oracle TeleService, from which a user can launch a script in the agent interface.
- The **TeleSales Agent** responsibility is required to run Oracle TeleSales, from which a user can launch a script in the agent interface.
- The **Collections Agent** responsibility is required for a user to launch Oracle Collections, from which a user can launch a script in the agent interface.

For a full list of responsibilities that may be required for Oracle Scripting, see Implementation Responsibility Matrix.

See Also

- Establishing Profile Settings for the Agent Interface
- Setting Session Idle Timeout Values for Apache JServ

4.5.2 Establishing Profile Settings for the Agent Interface

Setting system profiles determines the behavior of Oracle Applications. To implement the Scripting Engine agent interface, follow the procedure described in Setting System Profile Values for the following system profile options.

See the Guidelines section below for information on these settings.

Step	Option	Level	Value
1	IES : Architecture Type	Site	Apache Mid Tier/Servlet Architecture
3	IES : Display Suspend Button on Script Frame	Site, Application, Responsibility, or User	False *or* True
4	IES : Initial Script Frame Size	Site, Application, Responsibility, or User	<Desired frame size, based on agent desktop resolution or overall requirements>
5	IES : Scripting Panel Display Mode	Site	Display Multiple Panels at a time in Scripting window *or* Display Single Panel at a time in Scripting window
6	IES: Proxy Server Name	Site or Application	<Name of Proxy server if used for Scripting Engine agent UI>
7	IES: Proxy Server Port	Site or Application	<Port of Proxy server if used for Scripting Engine agent UI>
8	Apps Servlet Agent	Site or Application[1]	<URL for the Servlet zone of the enterprise Apache Web server>

[1] If your implementation requires a separate Apache JServ listener established for Oracle Scripting, select the **Application** option. This may be required to avoid session timeout errors, as described in Setting Session Idle Timeout Values for Apache JServ.

A full list of profile option settings required to implement Oracle Scripting is included in Oracle Scripting Profile Options. For more information, refer to this appendix.

Guidelines

Author Debug: Turning Author Debug Mode on will log additional informational messages regarding Script Author operations. When enabled, this profile option provides additional informational and error messages for Script Author debugging, generated in the Apache server logs. Note that you must administer (configure) your Apache Jserv to send console output to the Apache error log.

Architecture Type: The only supported architecture type for new Scripting Engine 11.5.8 implementations is the Apache Mid-Tier/Servlet Architecture.

> **Note:** If Apache Mid Tier/ Servlet Architecture value is not in the list of values, and instead you see Two Tier Mode and Three Tier Mode, stop your work and see your systems administrator. This means your environment has not been upgraded beyond Oracle Applications 11.5.5. *Do not select Three Tier Mode.*

> **Note:** New Scripting Engine implementations with this profile option configured for the Caching Architecture *will not be supported.*

Debug Mode: Turning Debug Mode on will log additional informational messages regarding Scripting Engine execution and other operations. Message logs for debugging or informational purposes are accessible from OAF technology stack applications such as the Survey Administration console in which the Diagnostics global button appears. This button is displayed in the UI only when the FND: Diagnostics button is set to Yes at the site, application, responsibility, or user level.

Display Suspend Button on Script Frame: Deployed scripts which have the global Suspendable script property option checked (set to **True**) can suspend Oracle Scripting transactions when this option is set to **True**. If enabled, a Suspend button displays on the bottom of the script frame (next to the Disconnect button). This button is hidden if the profile is set to **False** (or remains null). Oracle Corporation recommends setting this option to **True** only (a) if the agent interface is used in standalone mode, or (b) when applications integrated with Oracle Scripting are updated to include the ability to resume the script from the business application. This can be set at the site, application, responsibility, or user level, as required.

Initial Script Frame Size: This profile option controls the initial size of the Scripting Engine frame window in the Forms/Java agent user interface. Default size is 800 pixels in width and 600 pixels in height. Administrators can now select other options including 1024 X 768, 1280 X 1024, or Maximized. This can be set at the site, application, responsibility, or user level, as required.

Scripting Panel Display Mode: There are two lookup values from which to choose: Display Multiple Panels at a time in Scripting window and Display Single Panel at a time in Scripting window. Using single-panel mode, only the current (active) panel will appear on the screen at any given time in the agent UI. Under multi-panel display mode, the active panel will have focus, but panels previously visited will appear in the UI, scrolling out of site. Panels that are not active are de-emphasized (they appear smaller and gray). Active panels display unformatted, spoken, instructional, and custom text in black, blue, magenta, and custom colors as designed. For more information, see the section "The Panel Display Area" in the **Understanding Oracle Scripting > Understanding the Scripting Engine** section of the *Oracle Scripting User Guide* and online documentation.

Proxy Server Name: This references the name of any proxy server required for using Oracle Scripting, and enables the use of HTTPS. This is only required for Scripting Engine agent UI implementations using a Proxy server.

Proxy Server Port: This references the port number of the appropriate proxy server required for using Oracle Scripting, and enables the use of HTTPS. This is only required for Scripting Engine agent UI implementations using a Proxy server.

Apps Servlet Agent: Type the URL for the Servlet zone of the Apache JServ listener intended for Oracle Scripting in this field.

Syntax:

```
http://<servername>.<domain>:<Apache Web server port>/<servlet_zone>
```

Example:

```
http://server1.yourcompany.com:7777/oa_servlets
```

This setting is required for Apache mid-tier architecture implementations of Oracle Scripting. This profile is populated with the servlet zone of the Apache Web server. If your implementation requires only one Apache Web server listener, this profile is set at the site level and is used for all applications. Some environments require multiple Apache Web server listeners. For example, if some applications require short session idle timeout settings and others (like Oracle Scripting) require longer idle timeout settings, additional listeners may be configured. In such cases, the URL of the alternate listener is set at the application level for this profile. The specified applications will then reference the alternate Apache JServ instance.

Reference

- Oracle Scripting Profile Options
- Setting System Profile Values

See Also

- Assigning Scripting Engine-Related Responsibilities to a User
- Setting Session Idle Timeout Values for Apache JServ

4.5.3 Setting Session Idle Timeout Values for Apache JServ

In the Oracle Scripting Apache mid-tier architecture, the Scripting servlet runs in an Apache JServ. Every Apache JServ has a session idle timeout setting that controls how long a servlet session can be idle before it is automatically timed out (expired and terminated) by the Apache JServ. This behavior ensures that abandoned sessions in the servlet are not left to consume resources unnecessarily. Sessions in the Apache JServ can be abandoned due to a variety of causes (end user walked away from the computer without logging out, network failure, etc.).

When a servlet session is active in the JServ, but is idle (has received no client requests) for longer than the specified idle timeout setting, the JServ will automatically time out or expire the session. In Scripting, the servlet session stores all of the state information of a running script (panels accessed, answers given, etc.). Therefore, a timeout of a Scripting servlet session while an agent is running a script is a fatal error, causing the agent to lose all data that was collected in that session of the script. When this occurs, the agent must close the Scripting window and start a new session of the script, recapturing all information. To avoid this problem, you should verify that the ZONE.PROPERTIES file for the appropriate Apache JServ instance configured for Oracle Scripting has an appropriate timeout setting, and modify this setting if required.

Oracle Scripting Timeout Requirements

Oracle Scripting requires longer Apache JServ session idle timeout settings than other types of applications using Apache JServ. Interaction centers generally use a relatively small number of dedicated Oracle Scripting agents, who will typically remain logged in throughout the course of the day. In between script transactions, Scripting agents may be idle (from an application perspective) for long periods of time (possibly hours). During an active script transaction, when the agent is conversing on the telephone with customers, agents will be generally more active, but will still have some idle periods, possibly lasting several minutes. *It is critical to ensure the Scripting session does not expire during a script transaction,* as all data and history collected during the script transaction is lost when a session times out. Due to the usage pattern for Scripting and the importance of keeping the session intact throughout an active script transaction, session timeout values for an Apache JServ instance that is running Scripting sessions should be verified and may need to be extended.

The session idle timeout setting for Apache JServ must be set to a large enough value that it will not cause session time-outs during a script transaction. Naturally, this value will vary based on your specific business needs, as well as agent behavior and training, and the use of other applications relying on Apache JServ. In general, a value of 30 minutes is reasonable for typical Oracle Scripting configurations.

Oracle iStore Timeout Requirements

Oracle applications using Apache JServ that have many users and who log in for short periods of time throughout the day have different idle timeout requirements. For applications that follow this model (for example, Oracle iStore), the JServ should time-out idle sessions quickly (after a few minutes) so that resources are freed and available for other users.

Single Apache JServ Listeners Serving Multiple Application Types

Generally, it is not recommended to have the Apache JServ session idle timeout value set longer than 30 minutes if you have an application using the Apache JServ in which many users log in for short periods of time throughout the day, as well as other users who log in for extensive periods of time and have long idle periods.

Multiple Apache JServ Listeners

If your Apache is currently serving both types of applications, Oracle Corporation recommends configuring one Apache listener for each application type. Each Apache port/listener will have its own respective ZONE.PROPERTIES file, and can therefore support different session idle timeout settings per Apache listener. Thus, for example, if using Oracle iStore and Oracle Scripting, the ZONE.PROPERTIES file for the Apache listener configured for Oracle iStore should be set for a short session idle timeout, whereas the other should be set for a longer timeout for Scripting operations.

Using multiple Apache listeners may require changing the Apps Servlet Agent profile to specify a different Apache servlet zone at the Application level specifically for Oracle Scripting.

Use the following guidance to verify or set the session idle timeout to a value that will prevent any Scripting servlet session from timing out while an agent is in the middle of a script transaction.

Prerequisites

- This procedure is only required for Oracle Scripting release 11.5.6 or later implementations using the Scripting Engine agent interface in the Apache mid-tier architecture. This problem does not affect the Survey component of Oracle Scripting.
- At least one ZONE.PROPERTIES file must exist from appropriate installation of Apache Web server software installed during Oracle Applications implementation.
- As with all customization, you should back up the ZONE.PROPERTIES file before modification in the event of data failure.
- Modifications should occur at low-traffic times so the Web server can be taken offline momentarily.
- Stop the Apache Web server prior to modification.

Steps

1. Change to the directory where your Apache Web server is installed.

   Refer to Apache documentation or consult your Apache Web server administrator if you are not sure where to find this.

2. Change to the directory in which the ZONE.PROPERTIES file resides.

   This file is generally in the location <ORAHTTP_TOP>/Jserv/etc/ but may vary, depending on your specific Apache JServ configuration.

3. Open the ZONE.PROPERTIES file and locate the section of the file containing the session.timeout parameter.

   This is preceded by several commented lines (prepended with "#"). The section of the ZONE.PROPERTIES file should appear similar to the following:

   ```
   # Set the number of milliseconds to wait before
   # invalidating an unused session.
   # Syntax: session.timeout=(long)>0
   # Default: 1800000 (30 mins)
   session.timeout=1800000
   ```

4. Verify that the value (in milliseconds) meets requirements for this Apache listener. If not, change the value after "session.timeout=" (adjust it higher or lower accordingly).

> **Note:** There are 60,000 milliseconds in one minute.

5. Save your work.
6. Restart the Web server.

See Also

- Assigning Scripting Engine-Related Responsibilities to a User
- Establishing Profile Settings for the Agent Interface

4.6 Implementing the Scripting Engine Web Interface

Using the Scripting Engine Web interface, you can execute scripts in an Oracle Applications 11*i*-certified Web browser, either from a self-service Web application, or as an information-gathering survey.

This section includes the following topics:

- Implementation Considerations
- Assigning Web Interface Responsibilities to a User
- Setting Display Server for UNIX Environments

See Also

- Implementation Task Sequence
- Understanding Users Required for Implementation
- Creating an Administrator for Oracle Scripting
- Creating an Agent
- Implementing the Scripting Engine Agent Interface
- Implementing Administrative Interfaces

4.6.1 Implementation Considerations

The following requirements are all factors in implementing the Scripting Engine Web interface:

- Active survey campaign requirements corresponding to a survey URL
- Appropriate survey administrator responsibilities, roles and requirements
- A valid Oracle Applications session
- Appropriate script end-user responsibility
- Oracle Applications platform and patch level

Active Survey Campaign and Survey URL Requirements

Executing a script using the Scripting Engine Web interface requires appropriate survey campaign information, which must be administered in the Survey Administration console.

To execute a script in a Web browser or to participate in a Web-based survey, the Scripting Engine Web interface end user typically clicks a hyperlink to access a survey URL. The survey URL references a specific active survey deployment (identified by the dID parameter) on the surveying enterprise's Apache Web server listener port. This may be the same Apache listener port established for all Oracle Applications at the enterprise, or it may reference an Apache JServ instance specifically set up for Oracle Scripting operations.

Survey Administration Responsibility, Roles and Requirements

The survey URL accessed by the script end user in the Web interface is generated when a survey administrator completes all requirements for a survey campaign in the Survey Administration console and makes the deployment active.

Administering survey campaign information is performed by an Oracle Applications user with the Survey Administrator responsibility. This administrator must have access to existing survey resources, survey campaign requirements (including start and end dates), and a valid, tested deployed script.

This administrator must have access to Oracle One-to-One Fulfillment master documents and lists created with Oracle Marketing. To access the Invitations tab from the Survey Administration console, this administrator user must be added to the appropriate fulfillment group, and requires two JTF roles (JTF_SYSTEM_ADMIN_ROLE and JTF_FM_ADMIN). This user will also require the One-to-One Fulfillment Administrator responsibility if the same Oracle Applications user will also be the administrator for Oracle One-to-One Fulfillment.

Valid Oracle Applications Sessions

Accessing the script or survey in a Web browser associated with the survey URL requires a valid, authenticated Oracle Applications session by an Oracle Applications user typically associated with one or more responsibilities.

For individuals executing a script in a Web browser from an integrated application such as Oracle iSupport, the Oracle Applications login information for the current valid Oracle Applications session is used. The individual selects a link from the customized self-service Web application interface (e.g., Oracle iSupport) to take the survey, which starts immediately using current login authentication.

For individuals executing a script in a Web browser who are not authenticated users of Oracle Application session, a guest user Oracle Applications account is accessed behind the scenes. This guest user account (seeded with Oracle Applications) provides access to an Oracle Applications session with privileges only to execute the designated script. The session starts upon accessing the survey URL, and is terminated upon completion of the script.

Script End User Responsibilities

Individuals executing a script in a Web browser from a self-service Web application such as Oracle iSupport will use the appropriate responsibility for that application.

Individuals executing a script in a Web browser as a survey use a guest user login. No responsibilities are typically associated with a guest user. As of this release, there are no setup steps required for the guest user (a common Oracle Applications guest user is now used; in previous releases, a guest user for the JTT technology stack was used and sometimes required administration).

Oracle Applications Platform and Patch-Level Requirements

- Display server setup for UNIX servers
- Manual configuration of XML SQL utility

References

- Assigning Web Interface Responsibilities to a User
- Setting Display Server for UNIX Environments

See Also

- Assigning Web Interface Responsibilities to a User

4.6.2 Assigning Web Interface Responsibilities to a User

Accessing the script or survey in a Web browser associated with the survey URL requires a valid, authenticated Oracle Applications session by an Oracle Applications user typically associated with one or more responsibilities.

For individuals executing a script in a Web browser from an integrated application such as Oracle iSupport, the Oracle Applications login information for the current valid Oracle Applications session is used, including appropriate iSupport responsibility. The individual selects a link from iSupport to take the survey, which starts immediately using current login authentication.

For individuals executing a script in a Web browser as a survey, an Oracle Applications session is initiated using the Oracle Applications guest user login. No responsibilities are typically associated with a guest user.

See Also

- Implementation Considerations

4.7 Implementing Administrative Interfaces

Oracle Scripting has two administrative interfaces: the Scripting Administration console and the Survey Administration console. While both are HTML consoles accessed by administrators, they are built on different Oracle Applications technology stacks.

Some setup steps common to both administrator interfaces, some are specific to one interface, and others relate to the installation, patch level, or platform for Oracle Applications.

The Scripting Administration console is built on the JTT technology stack. This console provides the user interface with which script developers can launch Script Author as a Java applet, and script administrators can administer Oracle Scripting files (deployed scripts and Java archives), as well as generate, view and analyze agent interface reports.

The Survey Administration console is built on the Oracle Applications framework. This console provides the user interface with which survey campaign administrators can administer survey resources, create and administer standard and targeted (list-based) survey campaign requirements, and monitor individual responses.

Which Interfaces Do You Need?

- As of Oracle Scripting release 11.5.8 (Interaction Center Family Pack P and later), all implementations require the Scripting Administration console.
- Any implementation making use of the Scripting Engine Web interface to execute a script in a Web browser will also require the Survey Administration console.

System Profile Settings

Users of either administrative interface must have Oracle Applications profile settings appropriately established. The Scripting Administration console additionally requires the administration of some system profiles used by that technology stack. Thus, while not required by the Survey Administration console, all implementations inherit the requirement to establish or verify JTF system profiles, as described below.

Reporting Implementation Requirements

To generate and view reports for analysis of survey information using Oracle Discoverer, you must perform certain post-installation implementation and administration steps, as outlined in Appendix C, Oracle Discoverer Workbooks.

Additionally, to generate and view the panel footprint report accessible from the Oracle Scripting Administration console, the following apply:

- Users of Oracle Applications installed between releases 11.5.1 and 11.5.4 must configure the Oracle XML SQL utility if this is not already configured.
- Users of Oracle Applications installed on a UNIX platform must set up a display X server.

Follow the procedures below to implement one or more administrative interfaces to meet the requirements of your implementation.

This section includes the following topics:

- Establishing Profile Settings
- Setting Up the Scripting Administrative Interface
- Setting Up the Survey Administrative Interface
- Configuring the Oracle XML SQL Utility
- Setting Display Server for UNIX Environments

See Also

- Implementation Task Sequence
- Understanding Users Required for Implementation
- Creating an Administrator for Oracle Scripting
- Creating an Agent
- Implementing the Scripting Engine Agent Interface
- Implementing the Scripting Engine Web Interface

4.7.1 Establishing Profile Settings

Setting system profiles determines the behavior of Oracle Applications. When implementing Oracle Scripting, follow the procedure described in Setting System Profile Values for the following system profile options. These include six JTF profiles establishing default behavior of JTT HTML-based applications and one ICX profile establishing the default language.

See the Guidelines section below for information on these settings. Set the following profiles as indicated. Order below is recommended only; you can set these options in any sequence.

The settings below are provided as recommendations. Some may not apply; for example, you may not wish for Oracle Scripting to be established as the default for JTT applications, or you may wish to display a different number of rows. ***If there are business rules or requirements in your environment for other settings, let those requirements take precedence,*** or set profile settings at lower levels for the specified users.

Step	Option	Level	Value
1	JTF_PROFILE_DEFAULT_APPLICATION	Site or User	519 (see note below)
2	JTF_PROFILE_DEFAULT_BLANK_ROWS	Site	3
3	JTF_PROFILE_DEFAULT_CSS	Site	JTFUCSS.CSS
4	JTF_PROFILE_DEFAULT_CURRENCY	Site	<Applicable currency. For example, USD for U.S. Dollars>
5	JTF_PROFILE_DEFAULT_NUM_ROWS	Site	10
6	JTF_PROFILE_DEFAULT_RESPONSIBILITY	Site or User	(see note below)

Step	Option	Level	Value
7	ICX: Language	User	As required. Example: American English

Guidelines

Default Application: This setting assumes this Survey administrator user does not already have other functions in Oracle CRM technology foundation (JTT) HTML-based applications. Should other responsibilities exist for this user, confer with your systems administrator or the specific user to determine whether to modify the existing setting, or set this at the user level. 519 as indicated above is the product code for Oracle Scripting. *This setting is not environment-specific.*

Default Responsibility: This setting determines the responsibility that is displayed when a user logs into CRM HTML-based applications. If no other responsibilities existed for this user, select Survey Administrator or Scripting Administrator, based on the requirements of your administrative users.If this user already has other responsibilities, it may be a requirement to leave this setting, or set this at the user level. The value for this setting may be specific to each environment. For more information, see Oracle CRM Technology Foundation (JTT) Profile Options.

ICX Language: This setting establishes language for the Internet Cartridge Exchange for Web-based transaction settings. This value is selected from the appropriate language from the FND_LANGUAGES table, and is typically set at the user level.

References

- Oracle CRM Technology Foundation (JTT) Profile Options
- Setting System Profile Values

See Also

- Setting Up the Scripting Administrative Interface
- Setting Up the Survey Administrative Interface
- Configuring the Oracle XML SQL Utility
- Setting Display Server for UNIX Environments

4.7.2 Setting Up the Scripting Administrative Interface

The Scripting Administration console provides users who are assigned the Scripting Administration responsibility with the user interface from which Script Author can be launched as a Java applet. Additionally, from this user interface, you can view and remove scripts deployed to the database; you can view, upload, and map custom Java archives to any script deployed to the database; and you can generate, view and analyze panel footprint reports.

Prior to this release, the Survey Administration console included a Quick Find menu feature to locate survey campaigns, cycles, or deployments. For users upgrading Oracle Applications, this menu continues to display in any CRM Technology Foundation (JTT technology stack) application associated with product code IES (Oracle Scripting), even though it is obsolete. (This menu will not appear in new implementations.) Thus, for upgrading implementations prior to Interaction Center Family Pack Q or release 11.5.9, this menu will appear in the Scripting Administration console. You can remove the Quick Find menu from Scripting applications manually, after which it will not appear. See Removing the Quick Find Menu.

Other than performing installation, patch level, or platform-specific setup steps, no other tasks are required to implement the Scripting Administrative Interface.

See Also

- Establishing Profile Settings
- Setting Up the Survey Administrative Interface
- Configuring the Oracle XML SQL Utility
- Setting Display Server for UNIX Environments

4.7.3 Setting Up the Survey Administrative Interface

The Survey Administration console provides users assigned the Survey Administration responsibility with the user interface from which survey campaigns are created and administered. Active survey campaigns are required in order to execute a script in a Web browser using the Scripting Engine Web interface.

From this user interface, you can administer survey resources, define or change survey campaign information and status, access Oracle Marketing and Oracle One-to-One Fulfillment functionality required for targeted (list-based) survey deployments, and view responses received from users who executed the survey campaign script in a Web browser.

Other than performing installation, patch level, or platform-specific setup steps, no other tasks are required to implement the Survey Administrative interface for use with standard (non-list-based) survey campaigns.

Implementations using targeted deployments require the fulfillment administrator role to be assigned to the survey administrator, and require the survey administrator to be a member of a fulfillment group.

Tasks

Tasks include:

- Assigning Fulfillment Administrator Role to Survey Administrator
- Adding Survey Administrator User to a Fulfillment Group

See Also

- Establishing Profile Settings
- Setting Up the Scripting Administrative Interface
- Configuring the Oracle XML SQL Utility
- Setting Display Server for UNIX Environments

4.7.3.1 Assigning Fulfillment Administrator Role to Survey Administrator

For list-based survey implementations, a user with the One-to-One Fulfillment Administrator and Survey Administrator responsibilities must also be assigned the Fulfillment Administrative role (JTF_FM_ADMIN). This role provides access to Oracle One-to-One Fulfillment's user interface from within the Survey Administration console.

To perform some Oracle HTML-based applications administration, including this task, your Oracle Applications user must be granted the JTF_SYSTEM_ADMIN_ROLE. If required, you can use the seeded SYSADMIN Oracle Applications user account. For more information, see Granting JTF Roles.

This step is only required for implementations requiring targeted (list-based) survey deployments, to allow access to the Invitations tab. If you only wish to access Oracle One-to-One Fulfillment from within that application's user interface (and not from the Invitations tab in the Survey Administration console), then perform this step for the Oracle Applications user assigned the One-to-One Fulfillment Administrator responsibility.

> **Note:** This step is not required for standard (non-list-based) survey campaign implementations, which will not be leveraging Oracle One-to-One Fulfillment capabilities.

Guidelines

- The user with the Survey Administrator and One-to-One Fulfillment Administrator responsibilities must be an employee in the enterprise database, must be available as a CRM resource, and must be associated with an Oracle Applications login.
- The Oracle Applications login must have the CRM HTML Administrator responsibility assigned to provide access to CRM HTML-based applications as an administrator with full privileges. This includes establishing the appropriate set of system profiles for site and user levels as discussed in previous implementation tasks.
- Oracle One-to-One Fulfillment does not need to be implemented at the enterprise to perform this step. Note, however, that Oracle One-to-One Fulfillment must be implemented in order to *use* the survey administration component of Oracle Scripting for list-based implementations, *including for testing*.

See Also

- Adding Survey Administrator User to a Fulfillment Group

4.7.3.2 Adding Survey Administrator User to a Fulfillment Group

This task is only required for list-based implementations, and should be performed once per Survey administrator user. However, this step is not required for non-list-based survey campaign implementations, which will not be leveraging Oracle One-to-One Fulfillment capabilities.

Use this procedure to add the Survey Administrator user to a Fulfillment group.

Prerequisites

- An Oracle HRMS employee must already exist in the enterprise, must be available as a CRM resource, and must be associated with an Oracle Applications login.
- An Oracle Applications user account must already be assigned both the Survey Administrator responsibility and the One-to-One Fulfillment Administrator responsibility.
- An Oracle One-to-One Fulfillment group must exist in order to perform this step.
- Oracle One-to-One Fulfillment does not need to be fully implemented at the enterprise to perform this step, as long as a Fulfillment group has been created.

Note, however, that Oracle One-to-One Fulfillment must be implemented in order to *use* the survey component of Oracle Scripting for list-based implementations, *including for testing*.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

One-to-One Fulfillment Administrator

Steps

1. From the Fulfillment Administration console, click the Group tab.

   The Groups page appears. A hyperlinked list of defined group names (if any) appear in a table.

   > **Note:** If no group names appear in the list, you must first create a valid Fulfillment group as documented in *Oracle Oracle One-to-One Fulfillment Implementation Guide*.

2. If you want to add the survey administrator user to an existing fulfillment group, do the following:

   a. Click on the appropriate Group Name hyperlink.

      The Group Detail page appears.

   b. In the Name text field of the Agents table, add the Oracle Applications user name of the agent (survey administrator) whom you wish to add to the Fulfillment group, and click **Go**.

      The Select Agent page appears.

   c. Click the appropriate hyperlinked user name in the Agent User Name column.

      If necessary, refine your search for the appropriate user name in the Select Agent text field and click **Search**. You can narrow your criteria using the % wildcard character.

      The Group Detail page refreshes. The user name you selected now appears in the Agents list.

d. Click **Update** to confirm your selection.

   The Groups page refreshes.

   Perform the remaining steps to verify that this implementation task was successful.

e. Select the same Group Name hyperlink from the Groups page.

   The Group Detail page appears.

f. Verify that the survey administrator user name appears in the list of agents.

3. If you want to add the survey administrator user to a new fulfillment group, do the following:

   a. Click **Create**.

      The Create Group page appears.

   b. In the Group Name text field, enter an appropriate unique group name.

   c. Optionally, in the Description text area, enter a description of this group.

   d. From the Server Name list, select an appropriate server on which this Fulfillment group will be created.

   e. In the Name field under the Agents area, enter search criteria for the first agent whom you wish to add to this Fulfillment group and click **Go**.

   - The name criteria should specify the Oracle Applications user name.
   - Search criteria is case-sensitive.
   - You can use the % wildcard character.

      The Select Agent page appears.

   f. Click the appropriate hyperlinked user name in the Agent User Name column.

      If necessary, refine your search for the appropriate user name in the Select Agent text field and click **Search**. You can narrow your criteria using the % wildcard character.

      The Create Group page refreshes. The user name you selected now appears in the Agents list.

   g. Repeat steps e and f until you have added all agents whom you wish to add to this Fulfillment group.

h. Click **Create** to save this group definition with the group members specified.

   Perform the remaining steps to verify that this implementation task was successful.

i. Select the same Group Name hyperlink from the Groups page.

   The Group Detail page appears.

j. Verify that the specified user names appear in the list of agents.

4. Continue your work, or click **Sign Out** to exit Oracle Applications.

See Also

- Assigning Fulfillment Administrator Role to Survey Administrator

4.7.4 Configuring the Oracle XML SQL Utility

Enterprises that implemented Oracle Applications using Rapid Install 11.5.1 through 11.5.6 that wish to use reporting in the Scripting or Survey Administration consoles must manually configure the Oracle XML SQL utility. Oracle Applications implementations using Rapid Install 11.5.7 or later include automatic configuration of the Oracle XML SQL utility and do not require this task to be performed.

Use this procedure to obtain the appropriate files from your Web server, locate them appropriately, and modify the classpath in the JSERV.PROPERTIES file used by the servlet engine in Oracle HTTP Server.

> **Note:** There is more than one way to specify a class path. *If the method described below is not supported by your enterprise implementation,* consult with your Apache Web server administrator to determine the appropriate method for your environment. This may include identifying the classpath in a control file or another configuration.

Login and Responsibility

You must have physical or Telnet access to the Web server, and have applmgr or sysadmin privileges. Use the appropriate login for these privileges in your environment.

Prerequisites

- Your web server must be configured to use JDK 1.2.2 or above. If you are using a JDK version below JDK 1.2.2, you can upgrade to the most recently certified JDK version with Oracle Applications using Oracle*MetaLink* Note 130091.1, titled *Upgrading to JDK 1.3 with Oracle Applications 11i*.
- You must retrieve the necessary Java libraries from the $COMMON_TOP/util directory of your web server to configure your web server properly. Obtain Xsu12.zip (for 11.5.1 installations) or XSU12_ver1_2_1.zip (for 11.5.2 through 11.5.5 installations).

Steps

1. Connect using Telnet to the enterprise system.
2. Change to the directory where your Oracle HTTP Server is installed. Refer to the Appendix of your Oracle Applications Installation manual if you are not sure where to find this.
3. Create a directory called **xsu**.
4. If the Java library you obtained is xsu12.zip, extract this file using the command:

   ```
   unzip $COMMON_TOP/util/Xsu12.zip OracleXSU/lib/oraclexmlsql.jar
   ```

   If the Java library you obtained is XSU12_ver1_2_1.zip, extract this file using the command:

   ```
   unzip $COMMON_TOP/XSU12_ver1_2_1.zip OracleXSU12/lib/xsu12.jar
   ```

5. Copy the file you just extracted to the new xsu directory.

   For example, on UNIX, you would execute one of the following commands:

   ```
   cp OracleXSU/lib/oraclexmlsql.jar xsu/
   ```

   or

   ```
   cp OracleXSU12/lib/xsu12.jar xsu/
   ```

6. Remove the OracleXSU or OracleXSU12 directory created by the unzip command.
7. Change to the directory in which the JSERV.PROPERTIES file resides.

Note: The actual name of this file may include an underscore and the Apache JServ port (for example, JSERV.PROPERTIES_9404). For the purposes of this document, the Apache Web server configuration file will be referred to generically as the JSERV.PROPERTIES file. If you have any difficulties locating the file, consult with your Apache Web server administrator.

The location of this file is environmentally dependent. It *may* be in one of the following paths:

<ORAHTTP_TOP>/Jserv/etc/jserv.properties

<ORAHTTP_TOP>/apache39/conf

The ORAHTTP_TOP variable may not be set for your environment, in which case the path will be fully qualified to identify the entire physical path in the file system. If your configuration is different, consult with your Apache Web administrator to locate this directory.

8. Load the JSERV.PROPERTIES file into a text editor.
9. Look in the JSERV.PROPERTIES file for several lines beginning with **wrapper.classpath=**.

   This is the section in the file that sets the CLASSPATH used by the web server.
10. If the Java library you obtained is **xsu12.zip**, add the following line to the JSERV.PROPERTIES file to include the XML SQL Utility libraries in your CLASSPATH:

    ```
    wrapper.classpath=[ORAHTTP_TOP]/xsu/oraclexmlsql.jar
    ```

    If the Java library you obtained is **XSU12_ver1_2_1.zip**, add the following line to the JSERV.PROPERTIES file to include the XML SQL Utility libraries in your CLASSPATH:

    ```
    wrapper.classpath=[ORAHTTP_TOP]/xsu/xsu12.jar
    ```

    Note: Replace [ORAHTTP_TOP] with the physical path where your Oracle HTTP Server is installed.

    ---
11. Save the JSERV.PROPERTIES file, and exit the text editor.

Your web server is now configured to use the Oracle XML SQL Utility.

Guidelines

- Back up your JSERV.PROPERTIES file prior to customization. After configuring and testing, also back up your customized JSERV.PROPERTIES file in a separate location.

> **Note:** When you apply a patch, any customizations you have made to the JSERV.PROPERTIES file *may be lost*. In this case you will need to restore customizations to this file after patching.

See Also

- Establishing Profile Settings
- Setting Up the Scripting Administrative Interface
- Setting Up the Survey Administrative Interface
- Setting Display Server for UNIX Environments

4.7.5 Setting Display Server for UNIX Environments

This procedure applies to Oracle Applications built on the Oracle CRM Data Model (JTF) technology stack using Unix operating systems on the applications tier. From an Oracle Scripting perspective, this applies to Oracle Scripting implementations (on the Unix platform) that require generation and analysis of panel footprint reports from the Scripting Administration console.

Oracle Applications can dynamically generate and cache images with embedded text at runtime. These dynamic images are used, for example, to create buttons and tab menu bars displayed throughout various HTML-based transaction pages.

JFC/SWING and AWT (graphics libraries that are standard components within Java) require access to the native library of the operating system to process graphics (e.g., to display a window, paint a picture, generate a GIF image, and so on).

The reporting functionality for Oracle Scripting (the panel footprint report available from the Reports tab of the Scripting Administration console) employs JFC/SWING and AWT on the middle tier, where GIF images are processed in support of dynamically generated reports with graphics, and to render a script in a Web browser using HTTP.

Java AWT on Windows servers (Windows NT, Windows 2000) does not need a display server to render the graphics dynamically. However, in order for Java AWT to perform dynamic image generation on UNIX, you must specify an X Server that will be used as a display server to generate dynamic images. Images are generated and cached on the file system. Once these images have been cached, they need not be generated again. For this reason, the display server does not need to be a high-powered machine, nor is it required to be a dedicated server. However, it must be accessible by the Apache server that will call it.

To ensure graphics display functions in all UNIX environments, there are two processes you should perform in a UNIX environment:

- Modify JSERV.PROPERTIES to implement the Display Server identifier
- Reference the X Server in the apachectl or jservctl file (whichever is used in your environment to start the JServ).

Prerequisites

You must have an X Server up and running, on the network and accessible from the Apache JServ machine.

Steps

1. Access the JSERV.PROPERTIES file.

   This is typically located at <ORAHTTP_TOP>/Jserv/etc/jserv.properties on the Web server.

2. Make the following changes to implement the Display Server identifier.

   This can be done through xhost +, or through a more secure xauth UNIX command.

   k. Add the following DISPLAY parameter immediately below the wrapper.bin variable.

   ```
   wrapper.env=DISPLAY=<xserver-hostname>
   <xserver-displayport>
   ```

   l. Replace <xserver-hostname> and <xserver-displayport> with the machine name and port number where the X Server is running.

   Example:

   ```
   wrapper.env=DISPLAY=myxserver.mycorp.com:0.0
   ```

Note: You can also reference a Windows NT machine that is running a UNIX emulator such as Exceed. See the man (UNIX manual) pages for xhost and xauth for more information.

3. Reference the X Server in the apachectl or jservctl file:

   Example:

   ```
   export DISPLAY=myxserver.mycorp.com:0.0
   ```

   Note: Whichever file is used to start the JServ for your environment (apachectl or jservctl) should be modified to reference the X server.

4. For your changes to take effect, fully stop and then restart the Apache Web server services.

Guidelines

- Establishing an X Server as a display server is no longer required for UNIX systems in order for survey respondents or self-service Web application users to participate in a survey or script using a Web browser.
- Establishing an X Server as a display server is no longer required for UNIX systems using the Survey Administration console.
- This step is still required in order to view reports from the Scripting Administration console.

References

- Testing Display Server for UNIX Environments

See Also

- Establishing Profile Settings
- Setting Up the Scripting Administrative Interface
- Setting Up the Survey Administrative Interface
- Configuring the Oracle XML SQL Utility

5 Implementation Administration Tasks

Tasks in this section address administration that may be required for implementation of Oracle Scripting.

Tasks

You can perform the following tasks:

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Finding Default Application from a Responsibility ID Value
- Finding the FND User ID Value of a Defined User
- Removing the Quick Find Menu

5.1 Creating and Administering Users

This section discusses the tasks required to create and administer Oracle Applications users.

This section includes the following topics:

- Creating an Employee in the Enterprise Database

- Creating a Resource Group
- Importing a CRM Resource
- Defining Oracle Applications Users

See Also

- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Finding Default Application from a Responsibility ID Value
- Finding the FND User ID Value of a Defined User
- Removing the Quick Find Menu

5.1.1 Creating an Employee in the Enterprise Database

Enterprises using Oracle Human Resource Management Systems (HRMS) must create employees with this application. For enterprises not using HRMS as their human resources application, employees must be created using CRM Resource Manager.

Use this procedure to create an employee in Oracle Human Resource Management Systems.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

HRMS Manager or CRM Resource Manager

Steps

1. If using HRMS, from the Navigator, select **People > Enter and Maintain** and click **Open**.

1. The Find Person window appears.

2. If using CRM Resource Manager, from the Navigator, select **Maintain Employee > Employee > Enter Person** and click **Open**.

3. Click **New**.

   The People window appears.

4. Enter the information for the new person.

   The following fields are required:

   - Last (Name)
   - Gender
   - Type (select **Employee**)
   - Date of Birth (if using Payroll)
   - Social Security Number (if using US HRMS)
   - Employee (Number) - If the Employee field is inactive, then Oracle HRMS is set up to automatically generate the employee number when the record is saved.

   Other fields may be required depending on how your enterprise has set up Oracle HRMS or CRM Resource Manager. In addition, when you save the record, you may receive one or more messages that explain the consequences of leaving certain fields blank.

5. From the File menu, click **Save**.

   In the status line, confirmation appears that a record has been applied and saved.

6. Continue your work or log out.

Guidelines

In a test environment, the Survey Administrator responsibility can be assigned to any Oracle Applications user. In a production environment, however, this user must be an existing HRMS employee. This employee must also be designated as a CRM Resource. Steps to create employees in the enterprise database, import them as

CRM resources, define Oracle Applications users, assign them responsibilities, and associate them with employees in the enterprise database are discussed in Creating and Administering Users.

HRMS Employee

The HRMS employee is typically created in Oracle Human Resource Management Systems (HRMS), if licensed by an enterprise. For enterprises that have *not* licensed HRMS, the ability to create employees is *also* a scaled-down function of Oracle CRM Resource Manager.

If HRMS *is* installed, Oracle Applications enforces the business rule of *requiring* the user to create a person and designate them as the person type "employee" within HRMS. If you attempt to create an employee using CRM Resource Manager while HRMS is installed, you will experience an error.

Creating an employee in HRMS is accomplished by a user that has been assigned the appropriate HRMS responsibility (in the United States, for example, the US HRMS Manager responsibility).

If HRMS is not installed and licensed by an enterprise, the employee must be created using the CRM Resource Manager responsibility.

CRM Resource

After an employee is created, additional information (not required by HRMS) is required in order for CRM applications at the enterprise to appropriately reference this individual. If the employee is created using CRM Resource Manager, this information is part of the definition process.

If the employee is created using HRMS, employee information must subsequently be identified as a CRM resource. (By "resource," the concept of a person in the enterprise is inferred.) To do so, import the HRMS employee into CRM Resource Manager using the CRM Resource Manager responsibility. This process includes creation of a Resource Number and a corresponding Resource ID for this individual which is required by CRM applications.

Regardless of which method is used to create the employee (using HRMS or CRM Resource Manager), this document refers to this employee as an HRMS employee.

The other assumption that is a prerequisite for all Survey implementation steps is that this employee is identified as a CRM resource.

See Also

- Creating a Resource Group

- Importing a CRM Resource
- Defining Oracle Applications Users

5.1.2 Creating a Resource Group

Use this procedure to set up resource groups in CRM Resource Manager. Resource groups allow the assignment of role types, roles, and usages to one group that may contain one or many members.

An employee in a group with the role of manager automatically becomes the manager of the other employees in that group and of the employees in the groups below in the hierarchy.

> **Note:** You must not assign more than one employee with the role of Manager per group. Doing so will impact reporting accuracy.

Use this procedure to create a resource group.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

CRM Resource Manager

Steps

1. From the Navigator, select **Maintain Resources > Groups** and click **Open**.

   The Define Groups page appears.

2. In the Group Name field, type a name for your resource group.
3. Click the Roles tab.
4. From the list in the Role Type field, select the appropriate role type.

For example, select **TeleSales** in the first row and **Collections** in the second row if this is a resource group for Oracle Collections agents, or select **TeleSales** if this is a resource group for Oracle TeleSales agents.

5. From the list in the Role field, select the appropriate role for the selected role type.

   For example, select **TeleSales Agent** and **Collections Agent** for use with Oracle Collections, or select **TeleSales Agent** for use with Oracle TeleSales.

6. Click the Usages tab.

7. From the list in the Usage field, select the appropriate usages for the group.

   For example, select **Sales and TeleSales** and **Collections** if this is a resource group for Oracle Collections agents, or select **Sales and TeleSales** if this is a resource group for Oracle TeleSales agents.

8. Click the **Save** icon.

9. Continue your work or log out.

See Also

- Creating an Employee in the Enterprise Database
- Importing a CRM Resource
- Defining Oracle Applications Users

5.1.3 Importing a CRM Resource

Use the following procedure to import a CRM resource.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

CRM Resource Manager

Steps

1. From the Navigator, select **Maintain Resources > Import Resources** and click **Open**.

   The Selection Criteria window appears.

2. In the Resource Category field, select **Employee**.

3. Enter any additional selection criteria.

   For example, in the Name field, select the name of an employee.

4. Click **Search**.

   Employees that meet the search criteria are listed in the Search Results area.

5. Refine your search if necessary. Select the appropriate employee and click **OK**.

   The Selection Criteria window refreshes with the criteria you confirmed.

6. When satisfied with your selection criteria, click **Search**.

   The Selection Criteria window refreshes. Employees that meet the specified search criteria are listed in the Search Results area. The Select checkbox for the matching employees are automatically selected.

7. Clear the Select checkbox for each employee for whom you do not want to create a CRM resource.

8. Click **Create Resource**.

   The Default Values window appears.

   Note: Do not make any changes or add roles from this window. You can add or modify this information in the resource details later.

9. Click **OK** to accept the defaults.

   The Selected Resources window appears. The Comments field indicates whether the resource is a new record, a duplicate record, or a duplicate record with a new role definition. The Select checkbox is automatically selected.

10. Clear the Select checkbox for each employee for whom you do not want to save as a resource.

11. To save the resources, click **Save Resource**.

A transaction number appears in the Transaction Number field. The transaction number is associated with each resource created during this transaction. More than one resource can have the same transaction number.

12. Click **Details**.

    The Resource window appears. Verify the name of the resource (Name), the name of the employee (Source Name), and the user name (User Name) and note the resource number (Number).

13. From the Resource window, click the Roles tab (if it is not already displayed).

14. In the Role Type field, select the appropriate roles for this imported resource.

    For example, if you want this user to access the Administration tab in Oracle Marketing, select **Sales** from the list of values. If you want this user to access Oracle TeleSales, select **Telesales** from the list of values. If you want this user to access Oracle Collections, select **Collections** in the first row and **Telesales** in the second row.

15. In the Role field, select the appropriate role for each role type.

    For example, if you want this user to access the Administration tab in Oracle Marketing, select **Sales Manager** from the list of values. If you want this user to access Oracle TeleSales, select **Telesales Agent** from the list of values. If you want this user to access Oracle Collections, select **Collections Agent** in the first row and **Telesales Agent** in the second row.

16. Click the **Save** icon.

17. From the Resource window, click the Groups tab.

18. In the Name field under Groups, select the group name you created in the Creating a Resource Group topic.

19. In the Group Member Roles area, click in the Name field.

    This field auto-populates with the appropriate role.

20. Click the **Save** icon.

21. Continue your work or log out.

See Also

- Creating an Employee in the Enterprise Database
- Creating a Resource Group
- Defining Oracle Applications Users

5.1.4 Defining Oracle Applications Users

Use the following procedure to define Oracle Applications user accounts (user name and password), assign responsibilities to users, and associate an Oracle Applications user with an employee in the enterprise database.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Security > User > Define** and click **Open**.

   The Users window appears.

2. In the User Name field, type the user name you want to define and press the tab key.

   User names should be between 6 and 64 characters in length and should not contain spaces. Avoid using special characters such as %, slash, or backslash. Hyphen and underscore characters are supported.

3. Optionally, in the Description field, type a description of this user.

4. In the Password field, type the password for this user and press the tab key.

   The first time the user logs in, this password must be changed.

   Passwords should be between 5 and 64 characters in length and should not contain spaces. Avoid using special characters such as %, slash, or backslash. Hyphen and underscore characters are supported.

5. In the Password field, retype the same password for confirmation, and press the tab key.

6. Optionally, in the Password Expiration area, select criteria for the expiration of this password.

7. In the Responsibilities tab, click in the first row of the Responsibility column.

8. Type the appropriate responsibility name and press the Tab key. Optionally, you can search for the appropriate responsibility name by selecting the ellipsis points (...).

   - You can enter the full responsibility name, or you can enter partial criteria followed by the percent wildcard. Oracle Applications will attempt to automatically validate and autofill based on the criteria you entered. If an exact match results, the remaining columns will be populated.
   - If an exact match cannot be determined, a Responsibilities window appears on top of the Users window. Your search criteria appears in the Find tab field, with all results that meet your criteria appearing in a list below.
   - If many choices appear in the list, you may scroll through the list to find the appropriate responsibility. You may also refine your search criteria in the Find field and click **Find**. Highlight the appropriate choice in the list, and click **OK**. The Find window closes, and the appropriate criteria for all columns populates in the current row.

   Responsibility information appears for the selected responsibility. By default, the Effective Date (From field) is populated with the SYSDATE.

9. Change the date in the Effective Date From field to a prior date (yesterday's date, the beginning of the current month, or other appropriate choice).

   While an effective date starting with the SYSDATE may be effective immediately in some environments, best practices dictate that you ensure that the effective date you designate will take effect immediately in all environments.

10. Optionally, enter a date beyond which the responsibility will no longer be effective in the To field. This date must be later than the SYSDATE.

11. Select **File > Save and Proceed** to save your work.

12. In a new row in the Responsibility table, repeat steps 8 through 11 for each responsibility you want to assign to this user.

13. In the Person field, search for the name of the person (employee in the enterprise database) for whom you want to associate this user account.

    - Enter full or partial criteria, in syntax Last Name, Title, First Name.
    - You can use the % wildcard character.

    If an exact match is found, the employee name will populate in this field. Otherwise, the Person Names window results.

14. If required, refine your search in the Person Names window and click **OK**.
15. Select **File > Save and Proceed** to save your work.
16. Continue your work or log out.

See Also

- Creating an Employee in the Enterprise Database
- Creating a Resource Group
- Importing a CRM Resource

5.2 Setting Up Debugging and Message Logging in Oracle Scripting

There are various methods to debug problems or view informational or error messages when using Oracle Scripting. Each method provides information for functionality or problems with different components.

- Problems with Script Author graphical scripts are displayed in the Syntax tab of the Script Author Java applet. Clicking on any specific error displays an error message in the Debug pane, which appears immediately below the graphical Script Author canvas (the visual work area). This is accessed by a user with the Scripting Administration responsibility. *No setup is required to view Script Author errors in the Debug pane.*

- When launching Script Author from the Scripting Administration console, status messages regarding the launching of the Java applet are visible in the Java console. If any errors in communication with the mid-tier occur, these are displayed in the Oracle JInitiator Java console. To enable this console, see Enabling the Oracle JInitiator Java Console. No other setup steps are required.

- Client-side message logs for scripts executed using the Scripting Engine component of Oracle Scripting are controlled by the IES : Debug Mode profile. The default setting for this is **Debug off**, and this is generally the recommended setting. Set this profile to **Debug on** at the user level *only when remediating problems.* When debug mode is enabled, these messages are written to the Oracle JInitiator Java console. To enable this console, see Enabling the Oracle JInitiator Java Console. For more information on profile settings, see Oracle Scripting Profile Options > Current IES Profile Options.

- Server-side message logs are controlled by FND profiles. FND system profile settings are detailed in Oracle Scripting Profile Options > Information, Error and Debug Logging Profile Options. These logs are accessed when the user selects the **Diagnostics** global button or link from an OAF application. The FND: Diagnostics profile must be set to **Yes** for the logged-in user to display this control. To view message logs, see Viewing Server Message Logs.

This section includes the following topics:

- Setting Up Applications Common Logging
- Enabling Logging for the Guest User

See Also

- Creating and Administering Users
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Finding Default Application from a Responsibility ID Value
- Removing the Quick Find Menu

5.2.1 Setting Up Applications Common Logging

As of this release, Oracle Scripting has migrated to the Applications Common Logging Infrastructure for those aspects of error and informational message logging supported by that technology stack. Those aspects include:

- Status and error information about Scripting Engine mid-tier operations
- The Scripting Engine HTML client
- PL/SQL information passed between the database and the Scripting Engine
- Scripts executed in the Scripting Engine Web interface
- Transactions in the Scripting Administration console and the Survey Administration console

To record errors or informational messages, set the FND system profile options. These profile options, the seeded lookup values for each, the recommended level to set each, and a detailed description of each profile is included in Oracle Scripting Profile Options > Information, Error and Debug Logging Profile Options.

In brief, the following profile options are relevant:

- FND: Diagnostics must be set to **Yes** for an administrator to view message logs. This enables the **Diagnostics** global link and icon, which appears at the top right side of the HTML user interface for Oracle HTML-based applications using the Oracle Applications Framework. Oracle Corporation recommends setting this at the user level to provide access to information in logs only to appropriate trusted individuals.
- FND: Debug Log Enabled must be set to yes in order to capture information.
- FND: Debug Log Level determines the amount and type of information captured.
- FND: Debug Log Module specifies the module for which messages are logged.

References

- To view message logs, see Viewing Server Message Logs

See Also

- Enabling Logging for the Guest User

5.2.2 Enabling Logging for the Guest User

For implementations using guest user authentication for scripts executed in the Scripting Engine Web interface, you must explicitly enable message logging if you want to view logs generated for this user.

For example, perform this task if you want to view message logs for users of the Scripting Engine Web interface who execute scripts in a Web browser through targeted surveys and Web scripts that are *not* hosted in Oracle self-service Web applications.

As a prerequisite, set the profiles as detailed in Oracle Scripting Profile Options > Information, Error and Debug Logging Profile Options.

At the User level, for the guest user, establish system profile options accordingly. You *must* ensure that the guest user name and password are registered in the system profile option value for the Guest User Password profile.

For example, for a guest user named GUEST with a password of ORACLE, set the Guest User Password profile to GUEST/ORACLE.

Note: The applications-level guest user is seeded with Oracle Applications. For security purposes, you may want to establish a more secure password for this account.

References

- To view message logs, see Viewing Server Message Logs

See Also

- Setting Up Applications Common Logging

5.3 Enabling the Oracle JInitiator Java Console

Oracle JInitiator is a Java applet used for Oracle Applications. It serves many purposes, including ensuring that the correct version of Java is executed on the client destktop. This applet launches when an Oracle Applications session is started.

The Oracle JInitiator Java applet includes the ability to view status messages using a Java console. From an Oracle Scripting perspective, you may want to view the Oracle JInitiator Java console for this applet to determine the outcome of certain events or monitor status.

For example, if debug mode is enabled for client-side message logs, the logs are sent to this console.

Also, when launching Script Author, status messages regarding the launching of the Java applet are visible in the Java console. If any errors in communication with the mid-tier occur, these are displayed in the Oracle JInitiator Java console.

Use this procedure to enable the Oracle JInitiator Java console.

Prerequisites

Oracle JInitiator must be installed on your client workstation.

Login

None

Responsibility

None

Steps

1. From the Start menu, select **Programs** > **JInitiator Control Panel**.

   The Oracle JInitiator Properties window appears.

2. In the Basic tab, select Show Java Console.
3. Click **Apply**.
4. Close the Oracle Jinitiator Properties window.

See Also

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Finding Default Application from a Responsibility ID Value
- Finding the FND User ID Value of a Defined User
- Removing the Quick Find Menu

5.4 Bypassing the Sales Resource Group Membership Requirement

To access the Audience tab in the Survey Administration Console, the Oracle Applications user account must have a sales group ID or a sales role. Administration of resource groups is accomplished in Oracle CRM Resource Manager.

By setting a profile option, you can bypass this requirement for the Survey Administrator responsibility as described below. This profile can also be set at the application level.

Sales Groups, Roles and Usages

Sales groups, roles and usages are administered in Oracle CRM Resource Manager. These are used by various applications, including Oracle Marketing, Oracle TeleSales and Oracle Collections.

You must define a sales group and assign the group roles and usages prior to assigning any members to the group. If a sales group already exists, you can add members to the group. Each group member must have a CRM resource ID, which is created upon importing an employee into CRM Resource Manager.

Some applications (such as Oracle Marketing) require steps to be performed in a particular order. Thus, the recommended order is as follows:

- Creating an employee using Oracle HRMS (if installed) or CRM Resource Manager (if HRMS is not installed).
- Creating an Oracle Applications user account with the appropriate responsibilities.
- Associating the Oracle Applications user account with the employee.
- Creating a resource group and assigning roles and usages to the group.
- Importing a CRM resource and assigning roles and usages to the resource.
- Adding a defined resource as a member of an existing group.

Prerequisites

None

Responsibility

System Administrator

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Steps

1. From the Navigator, select **Profile > System** and click **Open**.

   The Find System Profile Values window appears above the unpopulated System Profile Values window. In the Display area, the Site and Profile with No Values options are selected.

2. In the Display area, select **Responsibility**.

   The field changes color, and a set of ellipsis points appears.

   In the Responsibility field, type the appropriate responsibility for which you want to bypass the requirement of group membership, and press the tab key.

For example, type **Survey Administrator** and press the tab key.

3. Leave the **Profiles with No Values** option selected.

   This allows the query to return profiles that meet your search criteria but have no values associated with them yet (blank profiles).

4. In the Profile field, enter **Bypass Group Validation**.
5. Click **Find**.

   The System Profile Values window appears.

6. In the Bypass Group Validation row, click once in the Responsibility column.

   A set of ellipsis points appears.

7. Click the ellipsis points.

   The Bypass Group Validation window appears.

8. From the list of values, select **Yes** and click **OK**.
9. Save your work and exit.

See Also

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Granting JTF Roles
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Finding Default Application from a Responsibility ID Value
- Removing the Quick Find Menu

5.5 Granting JTF Roles

To perform certain JTF tasks, a user must be assigned a specific JTF role. For example, the JTF_SYSTEM_ADMIN_ROLE role provides certain administrative capabilities to a user. From an Oracle Scripting perspective, survey administrators using targeted deployments must be assigned the JTF_FM_ADMIN role to access the Invitations tab from the Survey Administration console.

Additionally, from an implementation perspective, your Oracle Applications user account must already have certain roles in order to grant those roles to other users.

Use the procedure below to grant JTF roles to Oracle Applications accounts (users).

Prerequisites

- To perform some Oracle HTML-based applications administration, including this task, your Oracle Applications user must be granted the JTF_SYSTEM_ADMIN_ROLE. If required, you can use the seeded SYSADMIN Oracle Applications user account.
- Your Oracle Applications user must be granted a JTF role before you can assign that JTF role to another user. Thus, this user must be granted the JTF_FM_ADMIN role. This role is available to the seeded SYSADMIN Oracle Applications user account.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

CRM HTML Administration

Steps

1. From the CRM Applications Administration console, click the Users tab.
2. Click the Registration subtab.

   The Introduction page appears.
3. From the side panel, click **User Maintenance**.

   The Users page appears.
4. In the search criteria field, enter the appropriate search criteria to find the appropriate user name, and click **Go**.

   > **Note:** You must enter at least three characters. In addition to those three characters, you can use the % wildcard search character in your criteria.

The Users page refreshes, displaying a summary list table containing the set of records matching your criteria.

5. If necessary, refine your search criteria and click **Go**.

6. From the resulting list of users, click the appropriate entry in the Username column.

   The User Details page appears.

7. In the User Details area, click **Roles**.

   The User-Role Mapping page appears. Two lists are displayed (Available Roles and Assigned Roles).

8. From the Available Roles list, select each role you want to assign.

   You can select multiple records from the list by pressing and holding the CTRL key (use the Option key for Macintosh computers) as you make each selection.

9. When all desired available roles are selected, click **> Move** to move it to the Assigned Roles list.

   For example, select JTF_SYSTEM_ADMIN_ROLE and JTF_FM_ADMIN and move them to the Assigned Roles list.

10. When satisfied, click **Update**.

    The User-Role Mapping page refreshes. Two lists are displayed (Available Roles and Assigned Roles). The roles you just assigned for this user should appear in the Assigned Roles list.

11. Continue your work or sign out of Oracle Applications.

See Also

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Finding Default Application from a Responsibility ID Value
- Removing the Quick Find Menu

5.6 Setting System Profile Values

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Profile > System** and click **Open**.

   The System Profile Values window appears. Above this, the Find System Profile Values window appears and has focus.

2. If you want to display and set profile options by site, in the Display area of the Find System Profile Values window, ensure **Site** is selected. If you do not wish to view or set profiles at the site level, clear the **Site** option.

3. You can display and set profile options by Application, Responsibility, or User. For each selection:

   a. Mark the appropriate option.

   b. In the text field next to the selected Application, Responsibility, or User option, type search criteria and click tab.

      If you typed full or partial search criteria with one match, the appropriate value populates the field. Otherwise, a window appears where you can narrow your search criteria.

   c. Refine your search if required, select the appropriate option, and click **OK**.

Note: Site is the highest and broadest level of Oracle Applications profile settings, and affects all Oracle Applications users unless a specific lower level is specified. By changing the search criteria from Site to any of the other values (Application, Responsibility, or User), you can query and set profiles for lower and more granular levels. A lower level profile setting, if populated, always takes precedence and supersedes any value set at a higher level.

4. Ensure the **Profiles with No Values** option is selected.

   This allows the query to return profiles that meet your search criteria but have no values currently associated with them (blank profiles).

5. In the Profile field, enter your search criteria. You may include the % wildcard character.

   For example, enter IES%. IES is the product code for Oracle Scripting. Profile values associated with the survey component are included in this category.

6. Click **Find**.

   The System Profile Values window populates, displaying all profile values for the requested search criteria at the requested levels.

7. Set each profile option as required by your implementation task. You can set these options in any sequence.

8. Select **File > Save and Proceed** to save your work.

   The status line will indicate that records has been saved.

9. If you do not need to set additional profile options at this time, then proceed to step 13.

10. If you want to set additional profile options with different search criteria, then from the **View** menu, select **Find**.

    The Find System Profile Values window appears.

11. Repeat steps 2 through 8 above.

12. When satisfied, select **File > Save and Proceed** to save your work.

    The status line will indicate that records have been saved.

13. Select **File > Close Form** to close this window.

14. Continue your work or log out of Oracle Applications by selecting **File > Exit Oracle Applications**.

See Also

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Finding Application and Responsibility ID Values
- Finding Default Application from a Responsibility ID Value
- Removing the Quick Find Menu

5.7 Finding Application and Responsibility ID Values

You must know the APPLICATION_ID value for your application and the RESPONSIBILITY_ID value for a specified responsibility before you can set the required Oracle CRM Technology Foundation (JTT) profile options appropriately.

Based on how Oracle Applications was installed in your environment, a responsibility may have different names. For example, in a Vision Database installation, the responsibility to access the Scripting Engine agent interface in standalone mode is Scripting Agent, Vision Enterprises. In a fresh installation, this responsibility is referred to as Scripting User. Nevertheless, both will have a single responsibility identification.

> **Note:** The APPLICATION_ID value for Oracle Scripting is typically 519. The RESPONSIBILITY_ID value for Survey Administrator is typically 21685. The RESPONSIBILITY_ID for Scripting Administrator is typically 23723. However, in some scenarios, the responsibility identification may differ, based on environmental factors.

Use the following procedure to find the APPLICATION_ID value and RESPONSIBILITY_ID value of a known responsibility.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Security > Responsibility > Define** and click **Open**.

   The Responsibilities form appears.

2. Choose **View > Find**.

   The Responsibilities search window appears.

3. Search for the appropriate responsibility, highlight it, and click **OK** in the search window.

   The Responsibilities form is populated with the record for the selected responsibility.

4. With your cursor in any field of the record, select **Help > Diagnostics > Examine.**

   If the Enable Diagnostics window appears, enter the apps level database password and click **OK**.

   The Examine Field and Variable Values window appears.

5. If you want to determine the Application ID, in the Examine Field and Variable Values window, from the Field list, select **APPLICATION_ID** and click **OK**.

   The Value field in the Examine Field and Variable Values window is populated with the value of APPLICATION_ID.

6. If you want to determine the Responsibility ID, in the Examine Field and Variable Values window, from the Field list, select **RESPONSIBILITY_ID** and click **OK**.

   The Value field in the Examine Field and Variable Values window is populated with the value of RESPONSIBILITY_ID.

7. If you want to determine the Responsibility Key, check the value of the selected responsibility in the **Responsibility Key** field.

See Also

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Setting System Profile Values
- Finding Default Application from a Responsibility ID Value
- Removing the Quick Find Menu

5.8 Finding Default Application from a Responsibility ID Value

If a default application is already established for JTT applications, and you see only an application ID number in the JTF_PROFILE_DEFAULT_APPLICATION system profile, you must know which application is assigned, to determine whether to override this value with 519 (for Oracle Scripting).

Use the following procedure to determine (from a known application ID) the name of the application.

Prerequisites

You must have the application ID value from the JTF_PROFILE_DEFAULT_APPLICATION system profile.

Login

Log into SQL*Plus or another SQL application as a user with read privileges to the apps schema

Responsibility

None

Steps

1. Launch SQL*Plus or another SQL application.

2. Log into the appropriate applications instance.
3. Perform the following query to determine (from a known application ID) the current default HTML JTF application for this environment.

```
SELECT
   A.APPLICATION_ID, B.APPLICATION_NAME
FROM
   FND_APPLICATION A, FND_APPLICATION_TL B
WHERE
   A.APPLICATION_ID = B.APPLICATION_ID AND
   A.APPLICATION_ID = <KNOWN_APPLICATION_ID>;
```

Guidelines

Replace <KNOWN_APPLICATION_ID> with the value of the application ID in your SQL statement.

See Also

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Finding the FND User ID Value of a Defined User
- Removing the Quick Find Menu

5.9 Finding the FND User ID Value of a Defined User

When viewing log messages for Oracle Applications using the OA Framework, a User ID automatically populates in the User ID field, with the value applicable to the current Oracle Applications user account. This value is from the USER_ID column of the FND_USER table in the Oracle Applications schema.

If you want to view log messages for a specific user other than your current user, you must first obtain this user ID.

This section includes the following topics:

- Obtaining the FND USER_ID Value Using SQL
- Obtaining the FND USER_ID Value from the Users Window

See Also

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Removing the Quick Find Menu

5.9.1 Obtaining the FND USER_ID Value Using SQL

Use this procedure to determine the FND_USER_ID value for a defined user from a SQL application such as SQL*Plus.

Prerequisites

- You must know the Oracle Applications user account.
- You must have apps level access to the Oracle Applications database schema using a SQL application such as SQL*Plus.

Login

Log into SQL*Plus or another SQL application as a user with read privileges to the apps schema

Responsibility

None

Steps

1. Launch SQL*Plus or another SQL application.
2. Log into the appropriate applications instance.
3. Perform the following query to determine (with a known Oracle Applications user name, in upper case) the FND USER_ID value for this user.

```
SELECT
   USER_ID, USER_NAME
FROM
   FND_USER
WHERE
   USER_NAME = 'NAME';
```

Guidelines

Replace <NAME> in your SQL statement with the value of the Oracle Applications user name for the appropriate user.

See Also

- Obtaining the FND USER_ID Value from the Users Window

5.9.2 Obtaining the FND USER_ID Value from the Users Window

Use this procedure to determine the FND_USER_ID value for a defined Oracle Applications user account from the system administrator Users window.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Security > User > Define** and click **Open**.

   The Users window appears.

2. Place your cursor in the User Name field.

3. Enter query mode by selecting **View > Query By Example > Enter** or by selecting **F11** on your function keys.

   For example, to search for the seeded applications guest user, type **GUEST**.

4. In the User Name field, type the name of the Oracle Applications user account for which you wish to obtain the FND User ID.

5. Execute the query by selecting **View > Query By Example > Run** or by selecting **CTRL-F11** on your keyboard.

6. If the query is successful, the Users window refreshes. If the query is unsuccessful, in the User Name field, type other criteria, and run the query again.

7. With your cursor in the User Name field, select **Help > Diagnostics > Examine**.

   If the Enable Diagnostics window appears, enter the apps level database password and click **OK**.

   The Examine Field and Variable Values window appears.

8. To determine the FND user ID value for this user, then in the Examine Field and Variable Values window, from the Field list, select **USER_ID** and click **OK**.

   The Value field in the Examine Field and Variable Values window is populated with the value of the FND user ID for this user.

See Also

- Obtaining the FND USER_ID Value Using SQL

5.10 Removing the Quick Find Menu

When upgrading Oracle Applications, the obsolete Quick Find menu remains associated with product code IES, causing this menu to appear in the Scripting Administration console, where it has no function. This menu will not appear in the Scripting administrative interface for new implementations. For upgrade implementations, you must modify the JTT properties for product IES (Oracle Scripting) to remove the association between this menu and the Scripting product.

Use the procedure below to remove the Quick Find menu from its association with Oracle Scripting (product code IES).

Prerequisites

- You must log in as a user with JTF_SYSTEM_ADMIN_ROLE. For example, you can use the seeded SYSADMIN Oracle Applications user account.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

CRM HTML Administration

Steps

1. From the CRM Applications Administration console, click the Settings tab.
2. Click the System subtab**.**

   The Introduction page appears.
3. From the side panel, select **Properties > Advanced.**

   The Advanced - Properties page appears.

   > **Note:** If this page does not appear, your login does not have the JTF_SYSTEM_ADMIN_ROLE role. Log out and log in with an appropriate user.

4. From the View menu, select **IES**.

   The Advanced - Properties page refreshes. The records displayed show advanced properties for product code IES (Oracle Scripting).

   The following keys for IES advanced properties must be modified:

   a. search.factories

   b. service.oracle.apps.ies.survey.search.server.IESSurveySearchFactory.categories

   c. service.oracle.apps.ies.survey.search.server.IESSurveySearchFactory.desc

5. For each of the keys listed above, append a period and the responsibility ID for the Survey Administrator responsibility, as indicated below:

> **Note:** The responsibility ID for the Survey Administrator is typically 21685, but this may differ in your environment. To determine the appropriate ID for your environment, see Finding Application and Responsibility ID Values.

   a. In the Advanced - Properties table, from the Key column, click on the appropriate property.

      The Key Details page appears.

   b. In the Key field, append a period and the responsibility ID.

      For example, for the search.factories key, modify the entry to read:

      ```
      search.factories.21685
      ```

   c. Click **Update**.

      The Advanced - Properties page refreshes. The modified key is displayed.

   d. Repeat steps a through c for the remaining keys to be modified.

   e. In the Key field, append a period and the responsibility ID.

6. If the changes do not immediately result in the removal of the Quick Find menu in the Scripting Administration console, stop and restart the Apache Web server to see the changes take effect.

7. Continue your work or log out of Oracle Applications by selecting **Sign Out**.

See Also

- Creating and Administering Users
- Setting Up Debugging and Message Logging in Oracle Scripting
- Enabling the Oracle JInitiator Java Console
- Bypassing the Sales Resource Group Membership Requirement
- Granting JTF Roles
- Setting System Profile Values
- Finding Application and Responsibility ID Values

- Finding Default Application from a Responsibility ID Value
- Finding the FND User ID Value of a Defined User

6 Scripting Administration Tasks

The Scripting Administration console provides the user interface with which script developers can launch Script Author as a Java applet, and script administrators can administer Oracle Scripting files, as well as generate, view and analyze agent interface reports.

This console is accessed by logging into Oracle HTML-based applications using an Oracle Applications user account with the Scripting Administrator responsibility.

This section includes the following topics:

- Scripting Home
- Administering Oracle Scripting Files
- Generating and Viewing Panel Footprint Reports

6.1 Scripting Home

Upon logging into the Scripting Administration console, the Home tab appears.

Tasks

You can perform the following tasks:

- Launching the Script Author Java Applet

Concepts

For more information, see the following concepts:

- Scripting Administration Console
- Scripting Administration Console View List

- Agent Interface Reports

See Also

- Administering Oracle Scripting Files
- Generating and Viewing Panel Footprint Reports

6.1.1 Launching the Script Author Java Applet

Using Script Author, you can create, modify, and deploy scripts to the applications database.

Use this procedure to launch Script Author as a Java applet.

Prerequisites

None

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration console, click the Home tab.
2. Click **Launch Script Author**.

   A separate browser window appears, in which Oracle JInitiator launches, using authentication information from the current Oracle Applications session. Presently, a separate Java applet window appears with the Script Author user interface.

6.2 Administering Oracle Scripting Files

From the Administration tab of the Scripting Administration console, you can monitor and administer files supporting Oracle Scripting, including scripts deployed with Script Author, and custom Java archive files.

Tasks

You can perform the following tasks:

- Viewing and Removing Deployed Scripts
- Viewing and Administering Custom Java Archive Files

Concepts

For more information, see the following concepts:

- Using Custom Java

See Also

- Scripting Home
- Generating and Viewing Panel Footprint Reports

6.2.1 Viewing and Removing Deployed Scripts

Use the Deployed Scripts subtab to view or remove scripts deployed to the applications database from Script Author. Removing a script deletes the script and all related metadata from the database tables.

Use this procedure to view or remove deployed scripts.

Prerequisites

At least one script must be deployed to the applications database from Script Author.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.
2. From the Administration tab, click the Deployed Scripts subtab.

The Deployed Scripts page appears. A summary list of all deployed scripts meeting the filter criteria in the View list appears in the Deployed Scripts table.

3. Optionally, to change the set of records displayed in this list, perform any of the following:

   - From the View list, select a different filter option to display more or fewer deployed scripts based on active status and which user created the script.
   - To display subsequent records in the found set, click **Next** or the right arrow. To view previous records in the found set, click **Previous** or the left arrow.
   - To navigate through the found set of records in groups of ten, from the records range list between the Previous and Next links, select the appropriate range of records (for example, select **11 to 20 of *n***, where n is the number of records in the found set).

   After any selection above, the page refreshes, displaying the requested records.

4. Optionally, to remove scripts from the applications database, in the Select column, mark the **Select** option for each record you want to delete, and click **Remove**.

   > **Note:** Oracle Corporation recommends that you back up any script you may want to reuse to a local or network file system before deleting it from the database.

Guidelines

- The value in the Locked column is a read-only property indicating whether the listed script can be modified and deployed from Script Author (if value is **No**), or whether changes to the script metadata are prohibited (if value is **Yes**).
- Deployed scripts with ACTIVE_STATUS set to "0" are retained in IES_DEPLOYED_SCRIPTS so that existing footprinting and answer collection data can maintain valid references. Only active deployed scripts (those scripts with an ACTIVE_STATUS set to "1") can be executed.

See Also

- Viewing and Administering Custom Java Archive Files

6.2.2 Viewing and Administering Custom Java Archive Files

At runtime, scripts that contain commands referencing custom Java methods can execute each method as specified by the Script Author command parameters. Customized scripts can also use custom user interface Java beans to provide functionality at runtime. The source code for custom Java supporting Oracle Scripting must be compiled into executable class files and packaged into Java archives as appropriate.

Using the Scripting Administration console, you can upload custom Java archive files to the IES_JARFILES table in the applications database. Management of these files is accomplished from the Jar Listings subtab of the Administration tab.

Custom code loaded from the Scripting Administration console to the applications database need not be added to the class path of any configuration file.

Conversely, for custom code that is deployed manually to the APPL_TOP (the model available before this console was available), the corresponding class path must still be specified (in the JSERV.PROPERTIES file for enterprises using the Apache mid-tier architecture, or in the APPSWEB.CFG file for enterprises using the caching architecture). Manually deployed Java archives will not appear in the list of Java archive files on the **Administration > Jar Listings** page, nor can they be removed using this user interface.

Tasks

You can perform the following tasks:

- Viewing and Removing Uploaded Java Archive Files
- Uploading New Java Archive Files
- Overwriting Existing Java Archive Files with Updated Archives
- Changing the Global Property of Uploaded Java Archive Files
- Creating, Removing, and Viewing Java Archive Mapping Definitions
- Specifying Load Order of Java Archives Files

See Also

- Viewing and Removing Deployed Scripts

6.2.2.1 Viewing and Removing Uploaded Java Archive Files

When a Java archive is deployed to the database using the Scripting Administration console, a corresponding record is created in the IES_JARFILES table.

Use this procedure to view and remove custom Java code archives (JAR and ZIP files) uploaded to the applications database from the Scripting Administration console. Removing a Java archive deletes the JAR or ZIP file and all related metadata from the database tables.

Prerequisites

- At least one Java archive file must be uploaded to the applications database using the Scripting Administration Console.
- If a Java archive you want to remove is mapped to a script, delete the mapping prior to performing this task. You cannot remove a Java archive deployed to the database until any mappings associated with the archive are also removed.

> **Note:** Manually deployed Java archives will not appear in the list of JAR files on the Administration > Jar Listings page.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.
2. From the Administration tab, click the Jar Listings subtab.

   The Jar Files page appears. A summary list of all uploaded Java code archive files meeting the filter criteria in the View list appears in the Jar Files table.
3. If required, change the set of records displayed in the Jar Files table list as appropriate to display the desired code archives by performing any of the following:
   - From the View list, select a different filter option to display more or fewer Java archive files based on the Oracle Applications login used when the files were uploaded to the database.

- To display subsequent records in the found set, click **Next** or the right arrow. To view previous records in the found set, click **Previous** or the left arrow.
- To navigate through the found set of records in groups of ten, from the records range list between the Previous and Next links, select the appropriate range of records (for example, select **11 to 20 of *n***, where n is the number of records in the found set).

After any selection above, the page refreshes, displaying the requested records.

4. Optionally, to remove uploaded Java archive files from the server, in the Select column, mark the **Select** option for each Java archive you want to delete, and click **Remove**.

Guidelines

You cannot remove a Java archive deployed to the database until any mappings associated with the archive are also removed.

See Also

- Uploading New Java Archive Files
- Overwriting Existing Java Archive Files with Updated Archives
- Changing the Global Property of Uploaded Java Archive Files
- Creating, Removing, and Viewing Java Archive Mapping Definitions
- Specifying Load Order of Java Archives Files

6.2.2.2 Uploading New Java Archive Files

As discussed in Using Custom Java, custom Java archives can be used with Oracle Scripting to support the execution of custom commands in any script, or the use of Java beans in the agent interface. Using the Scripting Administration console, you can upload Java archive files from the Jar Listings subtab of the Administration tab.

When a Java archive is deployed to the database using the Scripting Administration console, a corresponding record is created in the IES_JARFILES table.

Custom code loaded from the Scripting Administration console to the applications database need not be added to the class path of any configuration file.

Use this procedure to upload new Java archive files for execution in a script at runtime.

Prerequisites

- Write the Java bean or Java method, compile into a class file, and package into a Java archive file.
- Java beans deployed to the database must be for use with the Scripting Engine agent interface only. The Web interface does not support the use of Java beans.
- To execute a method, a script must appropriately reference the method using a Script Author command.
- The script must be deployed to the applications database from Script Author.
- You must compile your custom code using a JDK level compatible with the Java Runtime Environment (JRE) used for your Scripting implementation.
- For all Java archive files deployed to the applications database from the Scripting Administration console for execution in scripts, the following limitations apply:

    1. Java archive files must be created using the "jar" utility of the JDK and specifying no compression (e.g., "jar -cf0...."), or by creating a ZIP or JAR file using a compression and archiving utility such as WinZip (with or without compression).
    2. Java archive files must be smaller than 1 megabyte.
    3. JAR file format is recommended over ZIP file format.
    4. You must meet the compilation requirements discussed in Using Custom Java > Java Compilation and Oracle JInitiator Dependencies.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.
2. From the Administration tab, click the Jar Listings subtab.

    The Jar Files page appears.

3. To begin uploading a Java archive file, click **Upload New Jar**.

The Upload File page appears.

4. In the **Name** field, type the name you want to associate with this Java archive file in the database.

   You can use any unique name to associate with this file. If you leave this field blank, the file name of the specified JAR or ZIP file will be used. You cannot use a name already assigned to another Java archive.

   If the Java archive is a Java bean, this name will be referenced by the Jar File Name field of the Script Author panel properties window.

5. Optionally, if you want this Java archive to be available globally (to all scripts deployed to this environment), select the **Global** flag option.

6. In the File field, click **Browse...** to begin locating and uploading the Java archive file you want to upload.

   The Choose File window appears.

7. Navigate to the appropriate location on your local file system or network, highlight the file you want to upload, and click **Open**.

   Note: You can only upload ZIP or JAR files. Other files will not execute appropriately using either Scripting Engine interface, and are therefore not allowed to be uploaded in this user interface.

   The Choose File window closes, and the File window is populated with the path and filename for the specified Java archive file.

8. To clear this selection and specify another file, click **Reset**.

9. To complete uploading the selected file, click **Upload**.

   The Jar Files page refreshes. An informational message indicates the logical name of the file you successfully uploaded.

Guidelines

- You must compile your custom code using a JDK level compatible with the Java Runtime Environment (JRE) used for your Scripting implementation. For more information, see Using Custom Java.
- JAR or ZIP files loaded from the Scripting Administration Console must be smaller than 1 megabyte.

- You cannot upload files of any type other than JAR or ZIP.
- When specifying the name of the Java archive file to upload, if you leave the Name field blank, the file name of the specified file will be used.
- Once uploaded, Java archives intended to provide custom commands to scripts must either be designated as global or must be mapped to one or more scripts to be accessible to the Scripting Engine at runtime.
- Java archives uploaded to replace panels with Java beans in the Scripting Engine agent interface do not require JAR mappings.

See Also

- Viewing and Removing Uploaded Java Archive Files
- Overwriting Existing Java Archive Files with Updated Archives
- Changing the Global Property of Uploaded Java Archive Files
- Creating, Removing, and Viewing Java Archive Mapping Definitions
- Specifying Load Order of Java Archives Files

6.2.2.3 Overwriting Existing Java Archive Files with Updated Archives

When custom code supporting the use of Oracle Scripting is updated, you must recompile the code, package it appropriately as a JAR or ZIP file, and upload the modified code to the database. This overwrites the previously loaded Java archive file.

Use this procedure to overwrite an existing Java archive file.

Prerequisites

At least one Java archive file must be uploaded to the applications database using the Scripting Administration Console.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.
2. From the Administration tab, click the Jar Listings subtab.

   The Jar Files page appears. A summary list of all uploaded JAR files meeting the filter criteria in the View list appears in the Jar Files table.
3. If required, change the set of records displayed in the Jar Files table list as appropriate to display the desired code archives.
4. Locate the JAR file in the Jar Files table that you wish to overwrite.
5. In the Select column, mark the **Select** option for the Java archive file you want to overwrite, and click **Overwrite Existing Jar**.

   The Upload File page appears.
6. Click **Browse...** to identify the Java archive you wish to upload in place of the existing file.

   The File Upload window appears.
7. In the File Upload window:

   a. Locate the appropriate JAR or ZIP file from your local file system or network.

   b. Highlight the file name.

   c. Click **Open**.

      The Upload File page refreshes. The FILE field contains the path and file name of the designated file.
8. To clear this selection and specify another file, click **Reset**.
9. To complete uploading the selected file, click **Upload**.

   The Jar Files page appears. An informational message indicates the logical name of the file you successfully uploaded.

See Also

- Viewing and Removing Uploaded Java Archive Files
- Uploading New Java Archive Files
- Changing the Global Property of Uploaded Java Archive Files
- Creating, Removing, and Viewing Java Archive Mapping Definitions

- Specifying Load Order of Java Archives Files

6.2.2.4 Changing the Global Property of Uploaded Java Archive Files

Java archive files uploaded using the Scripting Administration console have a global attribute. The data type for this attribute is Boolean, and defaults to **false** or null. To enable or disable this attribute for a selected code archive, you must explicitly select (setting to **true**) or clear the selected global attribute (setting to **false**), respectively. Code archives specified as global are loaded into the class loader for all scripts executed in the Scripting Engine.

Java code archives designated as global are loaded into the class loader for all scripts executed in the Scripting Engine in the specified environment, regardless of whether the scripts were deployed from the Script Author applet or from a standalone version of the Script Author application (use release Script Author 11.5.6 or later for compatibility with an 11.5.6 or later Scripting Engine).

Methods referenced in a global code archive need not be mapped to a specific script in order to execute at runtime. Each method in a global code archive is automatically accessible for execution in the Scripting Engine in any interface. You can change the global property at any time from the Scripting Administration console.

Use this procedure to select or clear the global property for one or more code archives.

Prerequisites

- Each Java archive file for which you want to change the global property must be uploaded to the applications database using the Scripting Administration Console.
- Each global Java archive file for which you want to clear the global property must be set as global using the Scripting Administration Console.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.
2. From the Administration tab, click the Jar Listings subtab.

   The Jar Files page appears. A summary list of all uploaded JAR files meeting the filter criteria in the View list appears in the Jar Files table.
3. If required, change the set of records displayed in the Jar Files table list as appropriate to display the desired code archives.
4. If you want to set the global property:
   1. For each Java code archive file displayed on the page that you want to set as global, in the Select column, click the Select option.
   2. For each Java code archive file displayed on the page that you want to set as global, in the Global column, select the **Global** option.

      A check mark indicates that the option is selected.
5. If you want to clear the global property:
   1. For each global Java code archive file displayed on the page for which you want to clear the global property, in the Select column, click the Select option.
   2. For each appropriate global Java code archive file displayed on the page, in the Global column, clear the **Global** option.

      An empty check box indicates that the option is cleared.
6. To complete the change to the global property of each selected code archive, click **Update**.

   The Jar Files page refreshes. For each code archive you selected (and any code archive previously set as global), the Global option is selected. For each code archive previously marked as global that you cleared, the Global option is cleared.
7. To change the global property for other archives, repeat steps 3 through 6 above.

See Also

- Viewing and Removing Uploaded Java Archive Files
- Uploading New Java Archive Files

- Overwriting Existing Java Archive Files with Updated Archives
- Creating, Removing, and Viewing Java Archive Mapping Definitions
- Specifying Load Order of Java Archives Files

6.2.2.5 Creating, Removing, and Viewing Java Archive Mapping Definitions

Java archive files supporting custom Java methods can be mapped to specific scripts, or specified as applying globally to all Oracle Scripting scripts. Methods in mapped files are only loaded into the class loader by the Scripting Engine if it identifies an appropriate script mapping. Code archives specified as global are loaded into the class loader for all scripts.

An uploaded Java archive can be mapped to zero, one, or many deployed scripts. When a Java archive is mapped to one or more deployed script, a corresponding record is created in the IES_SCRIPT_JARFILES table tracking this relationship. When a mapping definition is removed, the mapping record is deleted from this database table and no longer appears in the mapping list. An existing mapping definition must be deleted before the mapped Java archive can be removed.

Java archive files supporting custom Java beans do not need to be mapped to any specific script. The Java beans will be available to run as required in each script containing Java beans.

Use the Jar Mapping subtab to map uploaded Java archive files to apply to specific scripts, to remove mapping definitions, to view defined mappings, and to specify load order.

Prerequisites

- At least one Java archive file must be uploaded to the applications database using the Scripting Administration Console.
- At least one script must be deployed to the applications database from Script Author.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.
2. From the Administration tab, click the Jar Mapping subtab.

   The Script Jar Files Mapping page appears, containing a Script Jar Files Mapping area and a Create Mapping area. A summary list of all existing mappings meeting the filter criteria in the View list appears in the Script Jar Files Mapping table.
3. Optionally, request additional Java archive mappings to be listed by changing the View By selection or by clicking the First, Previous, Next, or Last navigation hyperlinks at the bottom of the page.
4. To create a new mapping, in the Create Mapping area, do the following:

   a. From the **Select Script** list, select a deployed script.

   b. From the **Select Jar File** list, select the Java archive you want to map to the designated script.

   c. In the Load Order field, enter an integer representing the load order of this Java archive.

   d. Click **Create Mapping**.

   The Script Jar Files Mapping page refreshes, displaying the new mapping definition in the table.
5. To remove one or more existing mappings, in the Script Jar Files Mapping area, do the following:

   a. In the Select column, select each record representing a mapping you want to delete.

   A checkmark in the Select column indicates the record is selected.

   b. Click **Remove**.

   The Script Jar Files Mapping page refreshes. The records you selected for deletion are no longer listed in the table, and the corresponding information has been deleted from the database.

See Also

- Viewing and Removing Uploaded Java Archive Files
- Uploading New Java Archive Files

- Overwriting Existing Java Archive Files with Updated Archives
- Changing the Global Property of Uploaded Java Archive Files
- Specifying Load Order of Java Archives Files

6.2.2.6 Specifying Load Order of Java Archives Files

Load order refers to the order in which the Scripting Engine loads mapped Java archive files. If a Java method of the same name is included in two or more Java archives, and you want to control which method is used, specify the desired archive as having a higher load order. This is chiefly relevant only when methods of the same name have differences in the code.

There is currently no verification to ensure that specified load order of Java archives do not overlap. Thus, the Scripting Engine will load Java archives of the same load order level in the order they are returned to the Scripting Engine from its query of the database.

Use this procedure to create or modify the load order for uploaded Java archives mapped to specific scripts.

Prerequisites

- At least one Java archive file must be uploaded to the applications database using the Scripting Administration console.
- At least one script must be deployed to the applications database from Script Author.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.
2. From the Administration tab, click the Jar Mapping subtab.

   The Script Jar Files Mapping page appears, containing a Script Jar Files Mapping area and a Create Mapping area.

3. To modify an existing load order, from the Script Jar Files Mapping list, change the value in the Load Order field to a different integer and click **Update**.

   The Script Jar Files Mapping page refreshes. The modified load order is listed accordingly.

4. To create a load order, you must first specify mapping information. In the Create Mapping area, from the **Select Script** list, select the Java archive file for which you want to specify load order.

5. In the **Load Order** field, type a whole number indicating the intended load order, where the lowest number indicates earlier load order.

6. To save this mapping, including load order, click **Create Mapping**.

   The Script Jar Files Mapping page refreshes. Any new mappings are included in the Script Jar Files Mapping list. The load order is a modifiable field.

Guidelines

- Load order is only relevant for mapped Java methods.
- Load order is only relevant when Java methods of the same name may contain different code.

See Also

- Viewing and Removing Uploaded Java Archive Files
- Uploading New Java Archive Files
- Overwriting Existing Java Archive Files with Updated Archives
- Changing the Global Property of Uploaded Java Archive Files
- Creating, Removing, and Viewing Java Archive Mapping Definitions

6.3 Generating and Viewing Panel Footprint Reports

Use the Reports tab of the Scripting Administration console to generate, view and analyze the panel footprint report.

Other reports supporting Oracle Scripting are available through Oracle Business Intelligence.

Tasks

You can perform the following tasks:

- Generating a Panel Footprint Report

Concepts

See Also

- Scripting Home
- Administering Oracle Scripting Files

6.3.1 Generating a Panel Footprint Report

The panel footprint report indicates, by panel per script, what panels were visited and the duration of time (in milliseconds) spent in each panel. The business objective of running this report is to see how effective a script is at leading an agent through a dialog with a customer (for the interaction center) or determining which questions may have been the most confusing (for survey respondents). Essentially, this report helps in tuning a script for maximum clarity and streamlined flow. It is typically of most use in the interaction center to reduce talk time and improve a script's efficiency. Note that if this report is not used, then footprinting should be disabled at the script level to conserve system resources.

Use this procedure to generate a panel footprint report.

Prerequisites

- You must know the script name, reporting period start date, and reporting period end date to run this report.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration console, click the Reports tab.

   The Panel Footprint Summary page appears.

2. In the Select a Script field, type search criteria for the specific script for which you want to run the report and click **Go**.

- If required, then you can further refine your search and click **Go**.
- You can use partial search criteria and the % wildcard character if desired. To see a list of every available script, just type % and click **Go**.

A hypertext linked list of records which meet the specified search criteria, if any, will appear.

3. Click the appropriate hypertext-linked script name.

   The Panel Footprint Summary page refreshes. The selected script is listed in the Select a Script field.

4. In the Start Date field, using the calendar control, select the start date of the reporting period for which you want to generate a panel footprint report.
5. In the End Date field, using the calendar control, select the end date of the reporting period for which you want to generate a panel footprint report.
6. Click **Run Report**.

   The report generates, appearing on the refreshed Panel Footprint Summary page. The report lists which panels were visited and the duration of time (in milliseconds) spent in each panel.

7. Exit the window or run another report as desired.

   Reports are dynamically generated. There is no need to save your work.

See Also

- Scripting Home
- Administering Oracle Scripting Files

7

Survey Resources Administration Tasks

7.1 Administering Survey Resources

Survey resources are objects defined as part of a survey campaign that display for scripts at runtime to users of the Scripting Engine Web interface.

Survey resources consist of two *display types:* section resources and page resources.

Section resources include the header section and the footer section. When a script is executed in a Web browser, each panel is represented by a separate HTML page. For each panel at runtime, header and footer sections (if associated with a survey campaign) appear just above or below panel contents, respectively.

Page resources include the error page and the final page. These are full HTML pages that display when a script is executed in a Web browser, but do not contain panel content (nor do they include header and footer sections). Error page resources are displayed when an error condition occurs during script execution in a Web browser. (If no error occurs, this page is not displayed.) Final page resources are displayed for every script transaction for which they are associated, after the Scripting Engine processes the last panel in the flow of a script.

Thus, each survey campaign can include up to four survey resources (header section, footer section, error page, and final page resources).

Survey resources are comprised of two *components:* the survey resource *definition*, and the *physical* survey resource file itself. The survey resource definition serves as a logical pointer to the physical survey resource. Survey resources are defined (and HTML resources uploaded to the database) from the Survey Resources tab of the Survey Administration console.

As for physical survey resources, no provisions are made to create HTML or JSP files within the context of the Oracle Scripting product. Nonetheless, in order to display survey resources when you execute a script in the Web interface, physical survey resource HTML or JSP files must be created and uploaded appropriately. Creation of survey resources is required customization. For more information on creating physical survey resources, see Survey Resources in the Oracle Scripting Survey Concepts section.

As of this release, there are three *resource types:* an HTML file, a JavaServer Page (JSP) file, and a URL. (When listed in the survey administration UI, these are labeled HTML File Upload, Deprecated - JSP, and URL For Redirect, respectively.) Their applicability is described below:

- HTML files can be used for any display type (section or page survey resource) in the OAF technology stack only.
- JSP survey resources must be used for any survey resource displayed in a survey campaign using the deprecated JTT base technology stack only.
- URLs can be used for page resource display types (the error page or the final page) for survey campaigns using the OAF technology stack only.

Physical HTML files corresponding to a survey resource definition are stored in the applications database. These are uploaded when defining an HTML survey resource, explaining the label "HTML File Upload."

JSP files must be uploaded to the $OA_HTML directory on the applications server. This is conducted manually, as there is no method to accomplish this from within the context of Oracle Scripting.

URL resources must refer to an absolute URL accessible to the Apache Web server at runtime. Assuming the page to which users are being redirected exists, there is no physical file required to be uploaded.

Although survey resources differ in many ways based on the technology stack to be used at runtime, the following are attributes of all survey resources:

- Each survey resource you want to use in a survey campaign must first be defined in the Survey Administration console, to provide a logical pointer to its corresponding physical file or URL redirect.
- Once created, survey resource definitions persist; the same resources can be used by any number of survey campaigns (of the same technology stack) without limit.
- You can change which survey resources are associated with an open or active survey campaign after survey campaign creation.
- Physical survey resource files must be created by certified, knowledgeable HTML or JavaServer Page developers.

- Survey resource creation is outside the scope of Oracle Applications.

Compatibility Statement

Oracle Scripting supports two technology stacks for executing a script in the Scripting Engine web interface. These technology stacks employ different underlying technologies during the execution of a script in a Web browser at runtime. As a result of these differences, the deprecated JTT technology stack uses JSP survey resources, whereas the OAF technology stack uses HTML survey resources or URLs. Survey resources of both types are created and administered from the Survey Resources tab.

- After defining a base technology stack for a survey campaign, you can only assign the resources applicable to that technology stack.
- The OAF technology stack is recommended for all new survey campaigns.
- ***Use of the JTT technology stack for new survey campaigns is not recommended***. This technology stack is deprecated in this release, and will be obsolete in the next major release of Oracle Scripting. Survey operations using the JTT technology stack is extended through this release to support enterprises with ongoing survey campaigns, to provide backward compatibility and uniformity of reporting.
- The differences in resources for each type is only one aspect that is affected by the base tech stack of a survey campaign. You should mention that they actually run on different underlying technologies as well.

Tasks

You can perform the following tasks:

- Creating Survey Resources
- Creating Survey Resource Definitions
- Viewing or Deleting Survey Resource Definitions
- Uploading JSP Survey Resources

Concepts

For more information, see the following concepts:

- Survey Resources Differentiated by Technology Stack
- Detailed Description of Survey Resources

- Creating Survey Resources
- Seeded JSP Survey Resources

References

- Survey Campaign Administration Tasks

7.1.1 Creating Survey Resources

No provisions are made to create HTML or JSP files within the context of the Oracle Scripting product. Nonetheless, physical survey resource HTML or JSP files must be created and uploaded, and subsequently defined in the Survey Administration console, in order to use survey resources when you execute a script in the Web interface. Creation of survey resources is required customization.

Including Graphics in Survey Resources

Survey resources, like any other HTML or JSP page, may include images and hyperlinks. Survey resources for JTT survey campaigns must be located in the $OA_HTML directory on the applications server to use at runtime. Survey resources for OAF survey campaigns are loaded into the database when defining the survey resources.

Regardless of technology stack, any objects (such as GIF or JPG images) referenced in a survey resource page must also be available to the application server at runtime. Thus, ensure you reference these objects accordingly when creating survey resources, and ensure the graphics are loaded to the appropriate directory accessible to the Apache Web server at runtime. For more information, see Uploading JSP Survey Resources.

Seeded JTT Test Survey Resources

For JTT survey campaigns, test survey resource JSP files ship with Oracle Applications, seeded in the appropriate directory ($OA_HTML) on the applications server. You can use the provided test resources to test your implementation and to serve as building blocks for creating your own resources. Ensure you retain the seeded resources for testing.

No seeded footer survey resource sample is included with Oracle Applications. For test purposes, you can reference a header resource in the Footer Section field, to ensure it displays as appropriate at runtime.

For more information, see Oracle Scripting Survey Concepts > Survey Resources > Test Survey Resources.

Note: Files with a JSP file extension must be tested on an existing Web server to view appropriately.

You must have the requisite knowledge and resources to create or modify HTML and JSP files in order to administer survey resources for use with Oracle Scripting to execute scripts in a Web browser.

References

- Oracle Scripting Survey Concepts > Survey Resources > Test Survey Resources.
- Oracle Scripting Survey Concepts > Survey Resources > Creating Survey Resources

See Also

- Creating Survey Resource Definitions
- Viewing or Deleting Survey Resource Definitions
- Uploading JSP Survey Resources

7.1.2 Creating Survey Resource Definitions

Creating a survey resource definition writes a record to the database that serves as a logical pointer to a physical survey resource file intended to display at runtime. Survey resources must be defined before they can be associated to a survey campaign.

Once defined, survey resource definitions persist; the same resources can be used by any number of survey campaigns (of the same technology stack) without limit.

Different restrictions apply to defining survey resources, based on the technology stack for which they are intended to be used, and based on the function of the resource. For more information, see Guidelines below.

Use the following procedure to create survey resource definitions.

Prerequisites

- To create a survey resource definition for HTML File Upload survey resource types, you must have local or network access to the corresponding physical HTML file, since these resource types are uploaded at the time of definition.

- For survey resources of the Deprecated - JSP resource type, you must know the name of the corresponding physical JSP survey resource file. This file must be uploaded to $OA_HTML on the applications server in order for this survey definition to function at runtime.
- For survey resources of the URL For Redirect resource type, you must know the absolute URL of an HTML page that is accessible to the Apache Web server at runtime. You must use the hypertext transfer protocol when specifying the URL.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Resources tab.

   The Survey Resources page appears, displaying a list of any existing survey resources that match the View list criteria.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

   - To view survey resources defined by all users, select **All Survey Resources** from the View list and click **Go**.
   - To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. Click **Create** to begin defining a survey resource.

   The Define Survey Resources page appears.

4. In the Logical Name field, type a name to serve as a pointer to the physical survey resource file.

   This is the name by which the survey resources is typically identified in the Survey Administration console. The Logical Name is also the default filter type used for the Search and Select Resource page used when associating survey resources with a survey campaign.

You can use a logical name identical to the physical survey resource file or URL, or type a meaningful descriptive name that follows your enterprise or survey campaign requirements or naming conventions.

5. Optionally, in the **Description** text area, type a meaningful description.
6. From the Language list, select a language.

   This field reads and displays available languages from the Oracle RDBMS FND_LANGUAGES table. Select the appropriate language for your environment, if it is not already displayed.

   For example, for English in the United States, select **American English.**
7. From the Resource Type list, select the appropriate resource type.
   - HTML File Upload resource types can be designated for any survey resource of *any* display type (header sections, footer sections, error pages, or final pages) for survey campaigns executed using the OA Framework technology stack.
   - URL For Redirect resource types can be designated for *page display* survey resources (error pages or final pages) for survey campaigns executed using the OA Framework technology stack.
   - Deprecated - JSP resource types can be designated for any survey resource of *any* display type (header sections, footer sections, error pages, or final pages) for survey campaigns executed using the deprecated JTT technology stack.
8. If defining a resource type of **HTML File Upload**, then do the following:
   - If you know the name and path of the file you want to upload into the applications database, type this information.
   - To search for and identify the file from your local file system or network:
     * Click **Browse**

       The Choose file window appears.
     * Navigate to the file you want to upload, and select the file in the Choose file window.
     * Click **Open**.

       The Choose file window closes, and the name and path of the designated file appear in the File To Upload field.

   You can upload files with a file extension of HTM, HTML, or XML.

9. If defining a resource type of **Deprecated - JSP**, then in the Resource JSP field, type the name corresponding to a physical JavaServer Pages file.

   - The physical JSP file does not require dynamic content; it can be a static HTML file saved with a file extension of JSP.
   - For Oracle Applications implementations hosted on a UNIX server, this value is case-sensitive.
   - Although this file is not required to exist in $OA_HTML at the time of survey resource definition, it must be uploaded to that location prior to execution, or a *file not found* error will result at runtime, and the script will not be executable in a Web browser.

10. If defining a resource type of **URL For Redirect**, then in the Resource URL field, type the full, absolute URL of the page to which you want this survey resource to redirect the user as appropriate.

    - Include the hypertext transfer protocol.
    - This must be an absolute URL, such as http://www.oracle.com. You cannot use a relative path as the definition for this survey resource type.

11. When satisfied with the values for this resource, click **Apply**.

    The Define Survey Resources page closes.

    The Survey Resources page appears, displaying a list of any existing survey resources that match the View list criteria, including the resource you just defined.

Guidelines

- For survey campaigns executed using the deprecated JTT technology stack, three resources must be defined, and associated with a survey campaign: header section, error page, and final page. Footer section resources are optional.
- For HTML survey resources, the Define Survey Resources page includes a File to Upload field. The definition process for this resource type includes the uploading of the physical HTML file to the applications database.
- For URL survey resources, the Define Survey Resources page includes a Resource URL field. Only absolute URLs are accepted. The location of the URL designates the physical file that displays upon request.
- For JSP survey resources, the Define Survey Resources page includes a Resource JSP field, into which you enter <FILENAME>.JSP, the physical name of a JavaServer Pages file corresponding to the resource definition. The JSP file must

be uploaded to the $OA_HTML bin or directory on the applications tier by a system administrator or apps user with FTP or file-level access and write privileges to the APPL_TOP.

> **Note:** The JSP file must be uploaded prior to executing the survey deployment, or an error results. There is no validation process in place to verify that the JSP file referenced exists in the appropriate directory. Thus, it is not a requirement of survey resource *definition* to upload the file to $OA_HTML, but rather an execution prerequisite.

See Also

- Creating Survey Resources
- Viewing or Deleting Survey Resource Definitions
- Uploading JSP Survey Resources

7.1.3 Viewing or Deleting Survey Resource Definitions

A survey resource definition serves as a logical pointer to a physical survey resource file intended to display at runtime.

You can view or delete existing survey resource definitions from the Survey Resources page of the Survey Administration console.

For more information on deleting survey resources, see Guidelines below.

Use this procedure to view or delete survey resource definitions.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Resources tab.

   The Survey Resources page appears, displaying a list of any existing survey resources that match the View list criteria.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

   - To view survey resources defined by all users, select **All Survey Resources** from the View list and click **Go**.

   - To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. To delete one or more existing survey resource definitions, do the following:

   a. Enter or view list search criteria to display one more records in the survey resources definition list.

   b. For each survey resource definition you want to delete, in the Select column, mark the Select box.

   c. In the Select Object area, click **Delete**.

      A warning page appears, listing the selected survey resources, and asking you to confirm the deletion action.

   d. If you want to cancel the deletion of survey resource definitions, click **No**.

      The Survey Resources page appears. Data in the list is unchanged.

   e. If you want to delete the selected survey resource definitions, click **Yes**.

      The Survey Resources page appears. The selected records are deleted, and a confirmation message at the top of the page confirms the deletion.

Guidelines

- To delete a survey resource definition, it must not be associated with an active survey deployment. If you need to delete a survey resource that is currently in use, first define another suitable resource, and assign the new resource in its place for all active survey campaigns.

- When you delete a survey resource definition of the HTML File Upload resource type, you also cause the corresponding physical survey resource file to be deleted from the database.

- Since survey resource definitions of the other types do not upload a file automatically, deleting these definitions causes no change to physical files. Thus, if you wish to remove a file uploaded to $OA_HTML (for the deprecated JTT technology stack), you must manually delete the corresponding physical file.
- Deleting a survey resource definition of the URL For Redirect resource type does not affect the file located at that URL in any way.

  For survey campaigns executed using the deprecated JTT technology stack, three resources must be defined, and associated with a survey campaign: header section, error page, and final page. Footer section resources are optional.
- For HTML survey resources, the Define Survey Resources page includes a File to Upload field. The definition process for this resource type includes the uploading of the physical HTML file to the applications database.
- For URL survey resources, the Define Survey Resources page includes a Resource URL field. Only absolute URLs are accepted. The location of the URL designates the physical file that displays upon request.
- For JSP survey resources, the Define Survey Resources page includes a Resource JSP field, into which you enter <FILENAME>.JSP, the physical name of a JavaServer Pages file corresponding to the resource definition. The JSP file must be uploaded to the $OA_HTML bin or directory on the applications tier by a system administrator or apps user with FTP or file-level access and write privileges to the APPL_TOP.

> **Note:** The JSP file must be uploaded prior to executing the survey deployment, or an error results. There is no validation process in place to verify that the JSP file referenced exists in the appropriate directory. Thus, it is not a requirement of survey resource *definition* to upload the file to $OA_HTML, but rather an execution prerequisite.

See Also

- Creating Survey Resources
- Creating Survey Resource Definitions
- Uploading JSP Survey Resources

7.1.4 Uploading JSP Survey Resources

For HTML For Upload survey resources, the corresponding physical HTML or XML files are uploaded as part of the survey resource definition process.

For URL For Redirect survey resources, the corresponding physical HTML or XML files to which the survey resource definition points to be uploaded independently, as part of the maintenance of that Web sight by its administrator.

For Deprecated - JSP survey resources, the physical JSP survey resource files must be physically uploaded to the $OA_HTML directory on the applications server. This is required to make those files accessible to the Apache Web server at runtime, so the resources can display as appropriate.

Seeded Test JSP Survey Resources

Test JSP survey resources are already seeded in $OA_HTML.

Graphics and Survey Resources

If you employ graphics in custom survey resources, the appropriate physical image files (typically in GIF or JPG file format) must be uploaded to a server location accessible to the Apache Web server at runtime.

Since images supporting Oracle Applications are stored in the $OA_MEDIA directory on the server, this is one possible location to store graphics associated with survey resources. You can reference the graphic using its relative path. For example, if you store graphics in the $OA_MEDIA directory, reference your code to the appropriate path as in the following sample image source tag:

```
<IMG SRC="/OA_MEDIA/image_name.gif" WIDTH="50" HEIGHT="10" ALT="image
description" BORDER="0">
```

Consult with your system administrator to determine if there are different requirements for your environment.

Prerequisites

- To upload a physical JSP survey resource file, the file must first be created by a certified and knowledgeable JSP developer. The file should be tested on a Web server prior to deploying for production.
- You must have local or network access to the physical JSP file, and to any associated files (such as graphics referenced by the survey resource).

- You must have access and write privileges to the $OA_HTML directory on the APPL_TOP of the applications server. Consult your Apache Web server administrator or system administrator for access.
- If the resource contains graphics, you must have access to the directory to which these are stored. This location must be accessible to the Apache Web server at runtime (for example, $OA_MEDIA). Consult your Apache Web server administrator or system administrator for access.

> **Note:** $OA_HTML and $OA_MEDIA are variables which correspond to physical file locations on the Apache Web server. The physical directory structures may differ for each environment. For more information, consult your Apache Web server administrator or system administrator.

Login

Not applicable

Responsibility

Not applicable

Steps

1. On your local computer (or over a network), locate the <filename>.JSP files created for use as survey resources.
2. Connect to the applications server through an appropriate method such as TELNET.

   If you do not have permissions to access the appropriate environment, consult your system administrator.
3. Navigate to the $OA_HTML directory.

   $OA_HTML is a variable to point to a physical directory. JSP resources for Oracle Scripting must physically exist in this directory path in order to execute a script in a Web browser in the deprecated JTT technology stack.
4. Using an appropriate protocol such as FTP, "put" or copy the target JSP files to the $OA_HTML directory.

   Specify ASCII format for file transfer.

5. If any survey resource file includes references to graphics, navigate to the appropriate directory specified in the survey resource (for example, $OA_MEDIA).

   $OA_MEDIA is a variable to point to a physical directory in which media items for Oracle Applications are stored. The physical directory path may differ based on your specific environment or enterprise requirements for storing customized images in support of Oracle Applications.

6. Using an appropriate protocol such as FTP, "put" or copy the target graphic files to the appropriate directory (for example, $OA_MEDIA).

   The survey resources (and any supporting image files) are now available to Survey administrators through the Survey Administration console. Before these files can be used, resources must be defined in the Survey Administration console, and a survey campaign designated as using a base technology stack of **Deprecated - JTT** must be created and activated.

See Also

- Creating Survey Resources
- Creating Survey Resource Definitions
- Viewing or Deleting Survey Resource Definitions

8

Survey Campaign Administration Tasks

A *survey campaign* is a high-level set of instructions for executing a script in a Web browser (as a survey or Web script). All survey campaigns require, as a prerequisite, a script created using the Script Author component of Oracle Scripting. Each survey campaign has one or more processing units, called *cycles*. Each cycle, in turn, has one or more processing units called *deployments*. Deployments designate detailed execution requirements including, at minimum, start and end dates and times.

These requirements are entered into the Survey Administration console by Oracle Applications users with the Survey Administrator responsibility. As each level of requirements is created, the information is saved as a set of objects in the database. Cycles are child objects of the survey campaign (the parent object); deployments are children of the cycle object.

After defining a deployment, it must be explicitly activated. This generates a *survey URL*, a system-generated string of parameters to allow end users to execute the designated questionnaire script (a survey campaign parameter) in a Web browser.

The amount and depth of information required to administer survey campaigns, and the application tasks that are prerequisite, differ based on which of two types of survey campaigns are required.

Standard survey campaigns are quick and easy to administer. Once a deployment for a standard survey campaign is activated, it is executed when end users click a specified, static URL in a Web browser. There are no dependencies on other applications (other than the script). Standard survey campaigns are typically administered and deployed for use as surveys linked from a web site, or for Web scripts (in which self-service Web application users click a customized link in the application to execute the script in a Web browser).

Targeted survey campaigns are executed from an e-mail message. The recipients of the e-mail message are members of an Oracle Marketing list. The e-mail distribution (sometimes referred to as a *marketing blast*) is handled using Oracle One-to-One

Fulfillment. The list members that respond to the e-mail message by participating in the execution of the script are referred to as *respondents*.

Targeted survey campaigns take advantage of application integration, and require substantial planning. Administration using Oracle Marketing and Oracle One-to-One Fulfillment is required prior to defining a deployment for a targeted survey campaign.

To simplify administration for creating targeted deployments, the Survey Administration console hosts functionality from Oracle Marketing in the Audience tab (for list management functions) and functionality from Oracle One-to-One Fulfillment in the Invitations tab (to create invitations, reminders, and queries, and associate them with a template). These tabs are not required for administrators of standard survey campaigns. Certain implementation requirements apply to allow access to the functionality hosted using these tabs. These functions can also be performed from within each application’s own UI.

Once lists are created and invitations and reminders are suitably administered, survey administrators can define deployment details for a targeted survey campaign, which requires list and template information to be specified.

This document includes some task-based steps for setting up lists and templates in support of administering and executing targeted survey campaigns. For detailed information on the use of these applications, refer to the respective product documentation.

Tasks for administering Oracle Scripting survey campaigns, cycles, and deployments can be accomplished using the Survey Campaigns tab of the Survey Administration console.

Responses to an active survey campaign (saved when scripts are executed in a Web browser) can be viewed from the Survey Administration console. This information is accessible (by cycle and deployment) by selecting the Response View option from either the Update Survey Campaign page or the Update Deployment page.

Tasks for administering list management is considered Oracle Marketing functionality. These tasks can be accomplished using the Audience tab of the Survey Administration console, or from the Oracle Marketing user interface. For more information, refer to Oracle Marketing product documentation.

Tasks for administering master documents (HTML invitations and reminders) and their associated queries, and the template objects with which they are associated, is considered Oracle One-to-One Fulfillment functionality. These tasks can be accessed using the Audience tab of the Survey Administration console, or from the Oracle One-to-One Fulfillment user interface.

Tasks for administering concurrent programs supporting the Scripting Engine Web interface provide the most recent information for survey administration, targeted deployments, and reporting. These tasks are performed from Oracle Forms-based applications as an Oracle Applications user with the iSurvey User responsibility.

This section includes the following topics:

- Survey Home
- Administering Survey Campaigns
- Setting Up Invitations and Reminders
- Administering Concurrent Programs for Survey Execution

8.1 Survey Home

Upon logging into the Survey Administration console, the Home tab appears. At this time, the Survey home page contains only identifying information for the page. There are no navigable links or destinations specific to this category, and no functions associated with this page.

Tasks

You can perform the following tasks:

- Navigating the Survey Administration Console
- Setting Survey Administration Preferences

Concepts

See Also

- Administering Survey Campaigns
- Setting Up Invitations and Reminders
- Administering Concurrent Programs for Survey Execution

8.1.1 Navigating the Survey Administration Console

Tab Navigation

There are five tabs in the Survey Administration console user interface (UI). These tabs provide access to the following functions:

Tab	Scripting Function	Tasks You Can Accomplish
Home	Identify user interface	None
Survey Campaigns	Administer survey campaigns	■ Create survey campaigns and cycles ■ View and modify survey campaign details ■ Add additional cycles to existing survey campaigns or modify cycle details ■ Delete survey campaigns and cycles ■ Define deployments ■ View and modify deployment details ■ Activate deployments ■ Delete deployments ■ View responses (for deployments for which responses have been received)
Survey Resources	Administer survey resources	■ Create survey resource definitions ■ View and modify survey resource definitions ■ Upload HTML files for use as OA Framework survey resources ■ Delete survey resource definitions
Audience	Administer Oracle Marketing lists for targeted survey deployments	■ Create, import, or segment a list ■ Use Oracle Discoverer to build a list using the Discoverer workbook ■ Search lists ■ Administer the organization or person for list members
Invitations	Administer Oracle One-to-One Fulfillment invitations and reminders for targeted survey deployments	■ List Oracle One-to-One Fulfillment invitation and reminder master document details ■ Create and upload new master documents ■ Upload existing master documents ■ Create queries for master documents ■ Create a template identifying master documents ■ Verify the status of a fulfillment request ■ Check fulfillment request history.

The Audience and Invitations tabs provide access to user interfaces owned by other Oracle applications: Oracle Marketing and Oracle One-to-One Fulfillment, respectively. Oracle Marketing provides list management and campaign management functionality. Oracle One-to-One Fulfillment is used for creation, management, deployment and delivery of invitations and reminders. For survey administration, these two tabs are only required for targeted (list-based) survey deployments. As these products are documented elsewhere in appropriate product-specific documentation, only their functionality specific to using Oracle Scripting is addressed in this document.

The information contained in this document is the most current information available at the time of publication. Every effort is made to synchronize between the releases of Oracle Scripting and the various applications with which it integrates. Nonetheless, if the steps listed in this document do not match the user interface you see when you attempt to perform a task, please consult the product documentation appropriate to the release installed at your site.

See Also

- Setting Survey Administration Preferences

8.1.2 Setting Survey Administration Preferences

From the Preferences page of the Survey Administration console, you can affect the way HTML applications display.

The following preferences can be changed on this page:

Profile Category	Profile Option	Description	Example
General Preferences	Known As	First and last name of logged-in user, derived from Known As field from HRMS database.	For user Michael Smith, Mike Smith.
General Preferences	Language	Default language	American English
General Preferences	Territory	Country in which Oracle Applications is in operation.	United States
General Preferences	Timezone	Timezone, based in hours before or after Greenwich Mean Time (GMT)	(GMT -08:00) Pacific Time
General Preferences	Client Character Encoding	Appropriate character encoding for the client computer workstation of logged-in user.	Western European (Windows)

Profile Category	Profile Option	Description	Example
General Preferences	Accessibility Features	To provide additional accessibility to users of assisted technologies.	No
Notification	Notification Style	Electronic mail settings, enabled if logged-in user has default e-mail address defined.	Do not send me mail.
Formatting	Date Format	Defines date format for table listings, database queries, etc.	DD-MON-RRRRR (displays as 20-APR-2004)
Formatting	Number Format	Determines placeholder usage for thousands place based on standard country usage.	10,000.00
Password Settings	Old Password	Password used to log into current session.	OLDPASSWORD
Password Settings	New Password	New password intended to govern the remainder of this session and all future sessions until changed again.	NEWPASSWORD
Password Settings	Repeat Password	Repeat the new password entered in the previous field.	NEWPASSWORD

Use the procedure below to set or change general preferences for the logged-in user in the Survey Administration console.

Prerequisites

- To change display preferences, the new selections must be supported in your environment.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click **Preferences**.

   The **Preferences** page appears.

2. If you want to change any general, notification, formatting, or password preferences, enter the new setting in the appropriate field using the following guidelines:

   If you want to edit a value appearing in a text field, replace the existing contents of the text field with the desired value.

   If you want to change a value appearing in any list, select the new value from the list using your up and down arrow keys or by clicking the drop-down list control, scrolling to highlight the appropriate value, and releasing the mouse.

   If you want to change your current password, do the following:

   a. In the Old Password field, enter the exact value of the password you used to log into the current Oracle Applications session.

   b. In the New Password field, enter the exact value of the new password you want to use.

      Passwords should be between 5 and 64 characters in length and should not contain spaces. Avoid using special characters such as %, slash, backslash, or a space character. Hyphen and underscore characters are supported.

   c. In the Repeat Password field, enter again the exact value of the new password you want to use.

3. Review any modifications.

4. When satisfied, click **Apply**. To exit the application without saving any changes you have made to the fields on this page, click **Cancel**.

   When you apply changes, the page refreshes, and a confirmation appears indicating that your modifications were successfully completed.

   When you click **Cancel**, the page closes and the Navigator appears.

   When viewing subsequent HTML pages in this application, the appropriate settings will now take effect.

5. Continue your work or log out, as applicable.

See Also

- Navigating the Survey Administration Console

8.2 Administering Survey Campaigns

From an administration perspective, the survey campaign is a collection of top-level requirements for executing an information-gathering survey campaign by executing a script in a Web browser. These requirements are stored as a set of objects in the database. To achieve the goals of the campaign, you must create a survey campaign, and define for each survey campaign at least one cycle and at least one deployment. The first cycle for any survey campaign is created at the time of survey campaign creation. Additional cycles can be defined later (while the survey campaign status is open or active).

Defining deployments is a separate process that can be performed either immediately after survey campaign and cycle creation, or later (while the survey campaign status is open or active).

Use the Survey Campaigns tab to create, modify, or view survey campaigns, cycles, deployments, or to view responses for active or closed survey campaigns.

Tasks

You can perform the following tasks:

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

Concepts

For more information, see the following concepts:

- The Survey Questionnaire

- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes
- Survey URL

See Also

- Survey Home
- Setting Up Invitations and Reminders
- Administering Concurrent Programs for Survey Execution

8.2.1 Filtering the Display of Survey Campaigns

Each time you select the Survey Campaigns tab, the Survey Campaigns page appears. Listed on this page are all existing survey campaigns meeting the View list and Exclude Prototypes box criteria. These are context-sensitive controls which filter the survey campaigns that display on the page.

The View list defaults to the **My Survey Campaigns** option. This filters out any survey campaigns defined by Oracle Applications users other than the account you used to log into your current session. Selecting the **All Survey Campaigns** option displays all existing survey campaigns in the list, regardless of the Oracle Applications account used to create them in this environment.

The Exclude Prototypes box is cleared by default. Thus, by default, all survey campaigns (including those designated as prototypes) display on the page. When Exclude Prototypes box is selected, prototype survey campaigns are filtered out of the displayed list.

Use this procedure to change the view of existing survey campaigns that display on the Survey Campaigns page of the Survey Administration console.

Prerequisites

- To display survey campaigns using the My Survey Campaigns option, at least one survey campaign must already be defined in the Survey Administration console by you or by another individual using the same Oracle Applications user account that you used to log in.
- To display additional survey campaigns using the All Survey Campaigns option, a different Oracle Applications user with the Survey Administrator responsibility must first define at least one survey campaign.
- For selection of the Exclude Prototypes box to take noticeable effect, at least one survey campaign must be designated as a prototype.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, to view survey campaigns created by all users, from the View list, select **All Survey Campaigns**.

   - If using the mouse, select the appropriate value from the list and release the mouse button.
   - If using the keyboard, press the tab key until the View list has focus, and then use the up or down arrow keys to select the value from the list. When the appropriate choice is selected in the list, press the Tab key.

   The Survey Campaigns page refreshes, listing all survey campaigns (if any) created by any user for this environment.

3. Optionally, when viewing all survey campaigns, to filter out from the list any campaigns not created by you (or by another individual using the same Oracle Applications user account that you used to log in), from the View list, select **My Survey Campaigns**.

- If using the mouse, select the appropriate value from the list and release the mouse button.
- If using the keyboard, press the tab key until the View list has focus, and then use the up or down arrow keys to select the value from the list. When the appropriate choice is selected in the list, press the Tab key.

The Survey Campaigns page refreshes, listing only those survey campaigns (if any) created by the Oracle Applications account you used to log into your current session.

4. Optionally, to exclude from the list any survey campaigns designated as prototypes, select the Exclude Prototypes box.

   - If using the mouse, click the Exclude Prototypes box to select it.
   - If using the keyboard, press the tab key until the Exclude Prototypes box has focus, and then press the space bar to select it.

   The Survey Campaigns page refreshes, listing only those survey campaigns (if any) that are not designated as prototypes.

5. Optionally, if the Exclude Prototypes box is selected, to display all survey campaigns, including those designated as prototypes (if any), clear the Exclude Prototypes box.

   - If using the mouse, click the Exclude Prototypes box to clear it.
   - If using the keyboard, press the tab key until the Exclude Prototypes box has focus, and then press the space bar to clear it.

   The Survey Campaigns page refreshes, listing all survey campaigns, including those designated as prototypes (if any).

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments

- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.2 Creating Survey Campaigns and Cycles

The survey campaign is a structure that allows you to define the basic parameters for executing a script in a Web browser. The cycle is a child object of the survey campaign.

To execute a survey campaign, a number of validations occur. To meet execution requirements, you must designate a survey campaign name, identify a base technology stack in which the script will execute, provide a status (open), identify whether the survey is a prototype, identify the script to execute at runtime, and designate a cycle name. If you want specific survey resources to display, you must also identify those resources. Additionally, you must subsequently create a deployment (as a child to the cycle) and activate the deployment.

You can create and save a survey campaign with only minimal information: you must define a survey campaign name, select the base technology stack, provide a status, identify whether the survey is a prototype, and define a cycle name. Survey campaigns with this information cannot be executed until all required parameters are subsequently defined, as described in Viewing or Modifying Survey Campaign and Cycle Details. For more information, see Guidelines below.

Use this procedure to create a survey campaign and cycle in the Survey Administration console.

Prerequisites

In order to create a survey campaign in the Survey Administration console:

- When selecting a base technology stack***, Oracle Corporation recommends using the OAF technology stack*** for all new survey campaigns. Oracle Applications are standardizing on this technology stack, which is why Oracle Scripting is undergoing migration to this technology. The JTT technology stack is deprecated in this release, and will be obsolete in the next major release of Oracle Scripting.
- You must know in advance all required parameters of a survey campaign and a cycle before you begin this step.

Parameters may follow specific naming conventions or otherwise adhere to enterprise or campaign requirements. If in doubt, then consult with the individual responsible for determining survey campaign requirements.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Click **Create** to begin defining a survey campaign.

   The Create Survey Campaign page appears.

3. In the Create Survey Campaign area, in the Survey Campaign Name field, type the name you want to use for this survey campaign.

   This is the name by which this survey campaign will be referenced. This name should be specified by the campaign administrator in advance, and must follow enterprise requirements or project naming conventions, if applicable.

4. Optionally, in the Description field, type a description of this survey campaign.

   For example, enter meaningful information by which survey administrators can quickly discern the purpose of the campaign, its status, group ownership within the enterprise, or other identifying information.

5. From the Base Tech Stack list:

   - For survey campaigns intended for execution in the Oracle Applications Framework technology stack, select **OA Framework**.

     Oracle Corporation recommends using this technology stack for all new survey campaigns.

   - For survey campaigns intended for execution in the deprecated Java Technology Foundation (JTT) technology stack, select **Deprecated - JTT**.

     The JTT technology stack is deprecated in this release, and will be obsolete in the next major release of Oracle Scripting.

Caution: Once you specify the base technology stack in this field and save the survey campaign, this information cannot be changed.

6. From the Status list, verify that the survey campaign status is **Open**.

7. If you want this survey campaign to be defined as a prototype, select the **Prototype** box.

   Prototype survey campaigns are identical to standard survey campaigns, except that the script used as the survey campaign questionnaire is not locked. This allows survey campaign administrators more freedom to refine requirements for test survey campaigns, including modifications to the script for the designated survey campaign.

   Prototype survey campaigns and non-prototype survey campaigns differ in only one aspect: the survey questionnaire (the script) is locked when a non-prototype survey campaign is activated, but not in a prototype survey campaign. When a questionnaire script is locked, it cannot be modified again. This ensures data consistency for the results of the survey campaign across each respondent's answers. Designating a survey campaign as a prototype allows survey campaign administrators more freedom to refine requirements for test survey campaigns by modifying the properties of a script (such as questions and branching) and then allows them to deploy the final version to a non-prototype survey campaign, where it is assured that the data will be consistent.

8. In the Activation Requirements area, from the Questionnaire Script field, click the search icon to begin locating the specific deployed script to designate as the survey questionnaire script for this survey campaign.

   The Search and Select: Questionnaire Script page appears in a new window. This page contains a Search area and a Results area.

9. In the Search area, type specific search criteria in the Search field.

   The only criteria by which to search is the global name of the deployed script (Script Name). If you do not know the name of the script you want to use, you can use the % wildcard character.

10. When you are satisfied with the search criteria you have typed, click **Go**.

    The Search and Select: Questionnaire Script page refreshes, showing all records (if any) matching your search criteria.

- If the desired deployed script is not listed in the Results area, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.
- If necessary, use the **Previous** and **Next** navigation buttons to view additional records returned by the query.

11. In the Results list:

    a. Click **Cancel** to clear the selected list item.

    b. To confirm your selection and proceed, do one of the following:

    * From the Quick Select column, click the LOV Quick Select icon, or

    * From the Select column, choose the Select option to indicate the appropriate script, and at the bottom of the page, click **Select**.

      When you select a record, the Search and Select page closes.

12. Verify that the Questionnaire Script field on the Create Survey Campaign page contains the name of the appropriate script.

13. For OA Framework survey campaigns, skip to step 18.

14. In the Survey Resources area, to designate a header section survey resource, perform the following:

    a. From the Header Section field, click the search icon to begin locating the JSP survey resource to appear as a header section on each page in the script at runtime.

    Note: Survey resources must be defined before you can associate them with a survey campaign. For JTT survey campaigns, header section, error page, and final page resources are required. Footer section resources are optional. Physical files corresponding to the logically defined JSP survey resources are not required to create a survey campaign, but must be present in order to execute the campaign in a Web browser. For more information, see Administering Survey Resources.

    The Search and Select: Header Section page appears in a new window. This page contains a Search area and a Results area.

    b. In the Search area, filter the search criterion by selecting Logical Name (the default) or Physical Name.

Logical name is an optional name assigned to the resource at the time of survey resource definition. Physical name is the name of the corresponding physical file.

For survey resources that are JSP files, the physical name is the name of the physical JavaServer Page file that must be uploaded to the OA_HTML directory on the applications server prior to survey runtime. For JTT survey campaigns, only JSP resource types are displayed in the appropriate search pages.

c. Type specific search criteria in the Search field. If you do not know the name of the survey resource you want to use, you can use the % wildcard character.

d. When you are satisfied with the search criteria you have typed, click **Go**.

   The Search and Select: Header Section page refreshes, showing all records (if any) matching your search criteria.

   * If the desired survey resource is not listed in the Results area, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.

   * If necessary, use the **Previous** and **Next** navigation buttons to view additional records returned by the query.

   * Click **Cancel** to clear the selected list item.

e. In the Results list, to confirm your selection and proceed, do one of the following:

   * From the Quick Select column, click the LOV Quick Select icon, or

   * From the Select column, choose the Select option to indicate the appropriate survey resource, and at the bottom of the page, click **Select**.

   When you select a record, the Search and Select page closes. The Create Survey Campaign Page refreshes, with your recent selection appearing in the appropriate field.

15. Optionally, to designate a footer section survey resource, from the Footer Section field, repeat step 14, and search for the appropriate JSP resource in the Search and Select: Footer Section page.

16. Repeat step 14 once more from the Final Page field for the final page resource, and again from the Error Page field for the error page, searching for the appropriate JSP resource in the Search and Select: Final Page and the Search and Select: Error page, respectively.

17. For JTT survey campaigns, skip to step 20.

18. Optionally, for OA Framework survey campaigns, in the Survey Resources area, to designate a header section or a footer section survey resource, perform the following steps.

    Note: If header section or footer section resources are not provided, no header section or footer section will appear on the HTML page at runtime.

    Note: OA Framework survey campaigns do not require survey resources. However, any survey resources to be associated with an OA Framework survey campaign must first be defined. Physical files corresponding to the logically defined HTML survey resources are uploaded to the database at the time of survey resource definition. For more information, see Administering Survey Resources.

    a. If you do not want to define a header section or a footer section, skip to step 19.

    b. From the Header Section or Footer Section field, as appropriate, click the search icon to begin locating the HTML survey resource to appear as a header section or footer section on each page in the script at runtime.

       The Search and Select page for the header or footer section appears in a new window. This page contains a Search area and a Results area.

    c. In the Search area, filter the search criterion by selecting Logical Name (the default) or Physical Name.

       Logical name is an optional name assigned to the resource at the time of resource definition.

       For survey resources that are HTML files, the physical name is the name of the physical HTML file uploaded to the applications database at the time of resource definition.

       For OAF header and footer sections, only HTML resource types are displayed in the appropriate search pages.

d. Type specific search criteria in the Search field. If you do not know the name of the survey resource you want to use, you can use the % wildcard character.

e. When you are satisfied with the search criteria you have typed, click **Go**.

   The Search and Select page refreshes, showing all records (if any) matching your search criteria.

   * If the desired survey resource is not listed in the Results area, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.

   * If necessary, use the **Previous** and **Next** navigation buttons to view additional records returned by the query.

   * Click **Cancel** to clear the selected list item.

f. In the Results list, to confirm your selection and proceed, do one of the following:

   * From the Quick Select column, click the LOV Quick Select icon, or

   * From the Select column, choose the Select option to indicate the appropriate survey resource, and at the bottom of the page, click **Select**.

   When you select a record, the Search and Select page closes. The Create Survey Campaign Page refreshes, with your recent selection appearing in the appropriate field.

19. Optionally, for OA Framework survey campaigns, in the Survey Resources area, to designate a final page or error page survey resource, perform the following steps.

Note: If final page or error page resources are not provided, a simple default page for that function will be displayed by the system at the appropriate time.

Note: OA Framework survey campaigns do not require survey resources. However, any survey resources to be associated with an OA Framework survey campaign must first be defined. Physical files corresponding to the logically defined HTML survey resources are uploaded to the database at the time of survey resource definition. For URL survey resource types, no validation is performed on the specified URL. For more information, see Administering Survey Resources.

a. If you do not want to define a final page or an error page resource, skip to step 20.

b. From the Final Page or Error Page field, as appropriate, click the search icon to begin locating the HTML survey resource to appear as a final page or error page, respectively.

   The Search and Select page for the final page or error page appears in a new window. This page contains a Search area and a Results area.

c. In the Search area, filter the search criterion by selecting Logical Name (the default) or Physical Name.

   Logical name is an optional name assigned to the resource at the time of resource definition.

   For survey resources that are HTML files, the physical name is the name of the physical HTML file uploaded to the applications database at the time of resource definition.

   For survey resources designated as a URL for Redirect, the physical name is the specific URL associated with the survey resource.

   For OAF error page or final page sections, any defined HTML or URL for Redirect survey resource types are displayed in the appropriate search pages.

d. Type specific search criteria in the Search field. If you do not know the name of the survey resource you want to use, you can use the % wildcard character.

e. When you are satisfied with the search criteria you have typed, click **Go**.

   The Search and Select page refreshes, showing all records (if any) matching your search criteria.

- If the desired survey resource is not listed in the Results area, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.
- If necessary, use the **Previous** and **Next** navigation buttons to view additional records returned by the query.
- Click **Cancel** to clear the selected list item.

f. In the Results list, to confirm your selection and proceed, do one of the following:

- From the Quick Select column, click the LOV Quick Select icon, or
- From the Select column, choose the Select option to indicate the appropriate survey resource, and at the bottom of the page, click **Select**.

When you select a record, the Search and Select page closes. The Create Survey Campaign Page refreshes, with your recent selection appearing in the appropriate field.

g. Optionally, *for error pages using an HTML resource type only,* if you want error information to appear on the top of the designated error page (to help identify the nature of an error encountered at runtime), check the **Display Error on Top** box.

20. In the Create Cycle area, in the Cycle Name field, type an appropriate cycle name for this survey campaign.

    This is the name by which this cycle will be referenced. This name should be specified by the campaign administrator in advance, and must follow enterprise requirements or project naming conventions, if applicable.

21. Click **Apply**.

    The new survey campaign and cycle information are created, and all defined fields are saved to the database.

    The Update Survey Campaign page appears This page is similar to the Create Survey Campaign page, but in addition contains a Deployments area, from which you can define a deployment for the cycle you defined in step 20 above.

22. To continue from this point to define a deployment, proceed to the Defining Deployments procedure, step 5.

Guidelines

- The fields required to *create* and *save* a survey campaign and cycle include: Survey Campaign Name, Base Tech Stack, Status (required to be Open), Prototype status (selected or cleared), and Cycle Name.
- In order to *execute* a survey campaign using the JTT technology stack, the following additional fields are required: Questionnaire Script, Header Section, Error Page, and Final Page. OAF survey campaigns require the same fields, with the exclusion of the survey resources.
- You cannot change the base technology stack or the prototype status after a survey campaign is activated. In order to change this either of these fields, you will be required to recreate the entire survey campaign and all its child objects.
- The Prototype option is recommended only for test survey campaigns, since its script does not lock, and since an active Prototype survey campaign cannot be converted into a standard locking survey campaign type.
- An appropriate script serving as the survey questionnaire must be deployed to the applications database from Script Author before you can designate this script as the survey questionnaire.
- Any survey resources intended for use with the survey campaign must already be defined, as described in the section Administering Survey Resources.
  - Survey campaigns to be executed in the new OA Framework technology stack do not require any survey resources. However, if using them, the corresponding physical HTML files for defined survey resources must already be uploaded, or the URL to be used as a resource for error or final pages must already be identified. This process occurs at the time of survey resource definition.
  - Survey campaigns to be executed in the deprecated Java Technology Foundation (JTT) technology stack require, at minimum, a header section, error page, and final page resource defined. While the corresponding physical JSP files for defined survey resources need not exist on the server in order to *create* a survey campaign, they must reside in the appropriate directory in order to *execute* the survey questionnaire in a Web browser using the JTF technology stack.

See Also

- Filtering the Display of Survey Campaigns
- Viewing or Modifying Survey Campaign and Cycle Details

- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.3 Viewing or Modifying Survey Campaign and Cycle Details

Once a survey campaign is created, and before it contains any active deployments, you can view or modify any of its properties except the base technology stack.

Once a deployment has been activated, survey campaign and cycle details that you can change include the description, survey resources, and cycle name. If you change the deployment status from active, you can then change the survey campaign status.

For idle or closed survey campaigns, you can change the description, status, questionnaire script, survey resources, and cycle name fields.

Use this procedure to view or modify survey campaign and cycle details.

Prerequisites

A survey campaign must already be created and saved.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

   - When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.
   - To filter these records, see Filtering the Display of Survey Campaigns.

3. In the list of survey campaigns, for the appropriate survey campaign, click **Update**.

   The Update Survey Campaign page appears

4. View survey campaign details or make changes as desired.

   Note: If you want to change the status of an active survey campaign, you must first change the status of any of its active children deployments.

5. The only modifiable property for a cycle is the cycle name. For instructions on how to add or modify cycles, proceed to step 4 of the task Adding or Modifying a Cycle for an Existing Survey Campaign.

6. Click **Update** to save your work.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.4 Cancelling a Survey Campaign from the Survey Campaigns List

Survey campaigns cannot be deleted when their status is Open.

Use this procedure to change the status of an open survey campaign to Cancelled from the Survey Campaigns list.

Prerequisites

A survey campaign must already be created and saved, but not yet activated.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

   - When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.
   - To filter these records, see Filtering the Display of Survey Campaigns.

3. From the Select column of the survey campaigns list, select a status of **Cancelled**.

Guidelines

- Once a survey campaign is cancelled, its cycle is also cancelled.
- Once a survey campaign is cancelled, its status cannot be changed.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details

- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.5 Deleting Survey Campaigns and Cycles

Use this procedure to delete one or more survey campaigns and cycles from the Survey Administration console and from the applications database.

Prerequisites

- A survey campaign and cycle must exist in order to delete them. You must know the name of the survey campaign, or be able to determine its name from the list of survey campaigns.
- A survey campaign cannot be deleted if it contains one or more cycles with an active deployment.
- Only survey campaigns with a status of closed or cancelled can be deleted. You cannot delete open or idle survey campaigns.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

- When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.
- To filter these records, see Filtering the Display of Survey Campaigns.

3. In the Select column of the Survey Campaigns table, check each record that you wish to delete.

> **Note:** To delete multiple survey campaigns in one step, all survey campaigns you want to delete must appear on the page. If you cannot adjust search criteria to view all relevant survey campaigns, you must search for and delete survey campaigns separately.

4. Click **Delete**.

   The Survey Campaigns page refreshes.

5. Verify that the survey campaign you deleted no longer appears in the Survey Campaigns list.

   When a survey campaign is deleted, its child objects are also deleted.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.6 Adding or Modifying a Cycle for an Existing Survey Campaign

One cycle is defined for each survey campaign upon creating the survey campaign. Survey campaigns can contain any number of cycles.

Use this procedure to modify the cycle name, or to add a cycle to an existing survey campaign.

Prerequisites

A survey campaign and cycle must already be created and saved.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

   - When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.
   - To filter these records, see Filtering the Display of Survey Campaigns.

3. In the list of survey campaigns, for the appropriate survey campaign, click **Update**.

   The Update Survey Campaign page appears

4. To modify the name of an existing cycle, in the Cycles and Deployments area, do the following:

   a. From the View By filter, select **Deployment View**.

   b. If the name of the cycle you want to modify for this survey campaign is not already displayed, then from the Cycle Name list, select the appropriate cycle name.

   c. Click **Update Name**.

      The Update Cycle page appears.

   d. In the Cycle Name field, type the new name you want to assign to this cycle.

This is the name by which this cycle will be referenced. This name should be specified by the campaign administrator in advance, and must follow enterprise requirements or project naming conventions, if applicable.

e. Click **Apply**.

The Update Cycle page closes. The Update Survey Campaign page appears

f. In the Cycles and Deployments area, verify that the Cycle Name field is updated with the new name. If this survey campaign has more than one cycle, you can see other cycle names by viewing the Cycle Name list.

5. To add a cycle to this survey campaign, in the Cycles and Deployments area, do the following:

a. From the View By filter, select **Deployment View**.

b. Click **Add Cycle**.

The Add Cycle page appears. The parent survey campaign name is listed.

c. In the Cycle Name field, type an appropriate cycle name to add to this survey campaign.

This is the name by which this cycle will be referenced. This name should be specified by the campaign administrator in advance, and must follow enterprise requirements or project naming conventions, if applicable.

d. Click **Apply**.

The Add Cycle page closes. The Update Survey Campaign page appears

e. Verify that the Cycle Name field is updated with the new name. If this survey campaign has more than one cycle, you can see other cycle names by viewing the Cycle Name list.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Defining Deployments
- Viewing or Modifying Deployment Details

- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.7 Defining Deployments

A deployment is a detailed set of execution requirements for conducting a survey campaign or establishing a Web script. At least one deployment must be defined for each cycle for a survey campaign.

Cycles can contain any number of deployments.

After a deployment is created, it must be activated (deployed) before a script can be executed in the Scripting Engine Web interface.

Use this procedure to define a deployment for a cycle.

Prerequisites

A survey campaign and cycle must already be created.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

   - When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.
   - To filter these records, see Filtering the Display of Survey Campaigns.

3. In the list of survey campaigns, for the appropriate survey campaign, click **Update**.

   The Update Survey Campaign page appears

4. If more than one cycle is associated with this survey campaign, then in the Cycles and Deployments area, do the following:

   a. From the View By filter, select **Deployment View**.

   b. From the Cycle Name list, select the appropriate cycle.

      If a new cycle name is selected, the Update Survey Campaign page refreshes, showing the Deployments area specific to the selected cycle.

5. In the Deployments area, click **Create**.

   The Create Survey Deployment page appears.

6. Verify that the appropriate parent survey campaign name is listed in the Survey Campaign Name field.

7. Verify that the appropriate parent survey cycle name is listed in the Survey Cycle Name field.

8. In the Deployment Name field, type the name of the deployment you want to create.

   This is the name by which this deployment will be referenced. This name should be specified by the campaign administrator in advance, and must follow enterprise requirements or project naming conventions, if applicable.

9. Verify that the media type listed is **WEB**.

   At this time, no other media type is supported.

10. In the Status field, verify that the **Open** option is selected.

11. In the Deploy Date and Time field, click the calendar control widget.

    The Pick a Date page opens in a separate window.

12. If necessary, change the calendar settings in the Pick a Date page to display the month and year on which you want this deployment to start.

    To navigate to the correct month and year, click **Previous month** or **Next month**, as appropriate, or select the appropriate month and year from the month and year lists.

13. Click on the date for which you want this deployment to start.

    Note: The deployment start date and time must be equal to the SYSDATE or in the future.

The Pick a Date page closes. The Create Survey Deployment page appears. The deployment date you selected, and the time (in hours, minutes, and seconds), appear in the Deploy Date and Time field.

14. In the Response End Date and Time field, repeat steps 11 through 13 to set the response end date and time for this deployment.

15. In the Deployment Type field:

    - For standard deployments only, verify that the deployment type specified in the Deployment Type field is **Standard**. If so, skip to step 40. If not, select **Standard** and proceed to step 40.

      Standard deployments require no other deployment details. Standard deployments are sometimes referred to as "non-list-based" deployments.

    - For targeted deployments only, in the Deployment Type field, select **Targeted**, click **Go**, and proceed to the next step.

      The Create Survey Deployment page refreshes, displaying additional fields applicable to targeted deployments.

      Targeted deployments require detailed information to specify a targeted a list of possible survey respondents (functionality provided by Oracle Marketing) and details regarding invitations and reminders sent to the targeted list members (functionality provided by Oracle One-to-One Fulfillment). For this reason, targeted deployments are sometimes referred to as *list-based* deployments.

16. In the Hosting Options field, select a hosting option based on your survey campaign requirements, using the following guidelines:

    - **Standalone** deployments require guest user authentication. These are typically used for targeted survey campaigns, or for surveys or Web scripts that are not associated with self-service Web applications. Upon completion of the script, the Oracle Applications session is terminated. Unless the URL is otherwise customized, the URL for JTT survey campaigns uses the IESSVYMAIN.JSP template, and for OAF survey campaigns uses the OA.JSP template.

    - **Menu Based** hosting for targeted deployments using the OA Framework base technology stack is not supported.

    - **Menu Based** deployments require authentication from an existing Oracle Applications session. These are typically used for surveys or Web scripts executed when the user is already logged into an Oracle Applications session and using a self-service Web application; the authentication and database connection information for the current applications session are used to provide the privileges to execute the Scripting Engine Web interface. That application's the tab-based menus are hosted in the survey or Web script user interface. Upon completion of the script, the Oracle Applications session is maintained. Unless the URL is otherwise customized, this uses the IESSVYMENUBASED.JSP template.

17. Optionally, if you want to execute the survey using a different Apache Web server or Web server port, then in the Survey URL field, change the survey URL parameters as required.

    This field automatically populates with the Web server and port information for the current Oracle Applications session. If you need to point this to a second Apache Web server (for example, to execute a survey outside the corporate firewall), then type the name of the Web server hostname including domain, port, and reference to HTML bin, ending with a slash. For example, enter:

    ```
    http://server2.company.com:8888/OA_HTML/
    ```

    For more information, see the survey concepts topic Survey URL.

18. From the List Name field, click **Search for List Name** to begin locating the Oracle Marketing list that identifies the target audience.

    The Search and Select: List Name page appears in a new window. This page contains a Search area and a Results area.

19. In the Search area, type specific search criteria in the Search field.

    The only criteria by which to search is the Oracle Marketing list name (List). If you do not know the name of the list you want to use, you can use the % wildcard character.

20. When you are satisfied with the search criteria you have typed, click **Go**.

    The Search and Select: List Name page refreshes, showing all records (if any) matching your search criteria.

    - If the desired list is not included in the Results list, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.
    - If necessary, click **Previous** and **Next**, as appropriate, to view additional records returned by the query.

21. In the Results list:

a. Click **Cancel** to clear the selected list item.

b. To confirm your selection and proceed, do one of the following:

   * From the Quick Select column, click the LOV Quick Select icon, or

   * From the Select column, choose the Select option to indicate the appropriate list, and at the bottom of the page, click **Select**.

When you select a record, the Search and Select: List Name page closes. The Create Survey Deployment page refreshes.

22. In the Create Survey Deployment page, verify that the list you selected appears in the List Name field.

23. From the Invitation Template Name field, click **Search for Invitation Template Name** to begin locating the Oracle One-to-One Fulfillment template associated with the desired invitation master document.

    The Search and Select: Invitation Template Name page appears in a new window. This page contains a Search area and a Results area.

24. In the Search area, type specific search criteria in the Search field.

    The only criteria by which to search is the Oracle One-to-One Fulfillment template name (Template). A template identifies a master document name. In this case, you want to specify a template associated with the appropriate invitation master document (which in turn is associated with a query). If you do not know the name of the master document you want to use, you can use the % wildcard character.

25. When you are satisfied with the search criteria you have typed, click **Go**.

    The Search and Select: Invitation Template Name page refreshes, showing all records (if any) matching your search criteria.

    - If the desired template is not included in the Results list, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.
    - If necessary, click **Previous** and **Next**, as appropriate, to view additional records returned by the query.

26. In the Results list:

    a. Click **Cancel** to clear the selected list item.

    b. To confirm your selection and proceed, do one of the following:

       * From the Quick Select column, click the LOV Quick Select icon, or

  - * From the Select column, choose the Select option to indicate the appropriate template, and at the bottom of the page, click **Select**.

  When you select a record, the Search and Select: Invitation Template Name page closes. The Create Survey Deployment page refreshes.

27. In the Create Survey Deployment page, verify that the template you selected appears in the Invitation Template Name field.

28. In the Invitation E-mail Subject Heading field, type the name you want to appear as the subject line of the outgoing message created using the invitation master document.

    For example, type **Please participate in our survey.**

29. If no reminders are required, skip to step 38.

30. From the Reminder Template Name field, click the search icon to begin locating the Oracle One-to-One Fulfillment reminder master document.

    The Search and Select: Reminder Template Name page appears in a new window. This page contains a Search area and a Results area.

31. In the Search area, type specific search criteria in the Search field.

    The only criteria by which to search is the Oracle One-to-One Fulfillment template name (Template). A template identifies a master document (which in turn is associated with a query). In this case, you want to specify a template associated with the appropriate reminder master document. If survey campaign requirements dictate, this may be the same template used to specify the invitation. If you do not know the name of the master document you want to use, you can use the % wildcard character.

32. When you are satisfied with the search criteria you have typed, click **Go**.

    The Search and Select: Reminder Template Name page refreshes, showing all records (if any) matching your search criteria.

    - If the desired template is not included in the Results list, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.

    - If necessary, click **Previous** and **Next**, as appropriate, to view additional records returned by the query.

33. In the Results list:

    a. Click **Cancel** to clear the selected list item.

b. To confirm your selection and proceed, do one of the following:

   * From the Quick Select column, click the LOV Quick Select icon, or

   * From the Select column, choose the Select option to indicate the appropriate template, and at the bottom of the page, click **Select**.

   When you select a record, the Search and Select: Reminder Template Name page closes. The Create Survey Deployment page refreshes.

34. In the Create Survey Deployment page, verify that the template you selected appears in the Reminder Template Name field.

35. In the Reminder E-mail Subject Heading field, type the name you want to appear as the subject line of the outgoing message created using the reminder master document.

    For example, type **Reminder: survey closes last day of the month!**

36. In the Number of Reminders field, type an integer based on survey campaign requirements.

    This number determines how many reminders will be automatically sent to all list members. Upon request, additional reminders can be sent manually regardless of the value entered in this field.

37. In the Reminder Interval In Days field, type an integer.

    This number represents, in days, the span of time that will pass between reminders to be sent automatically to all list members.

    To determine this value, divide the number of reminders by the duration of the deployment (in days).

38. In the Maximum Responses Per Person field, if survey campaign requirements specify that list members can participate in a survey or execute a Web script for an established number of times, enter the appropriate integer.

    For example, if each list member must be restricted from executing the questionnaire script in a Web browser more than once, enter 1 in this field.

    This value determines how many times an individual (list member) can participate in a survey or execute a Web script. This restriction is based on the unique survey URL for a list member, which includes both the deployment identification code and the unique respondent identification code.

Note: This is a required field. Thus, if no maximum applies, type 0 in this field.

39. Optionally, in the Target Response Percent field, type an integer between 1 and 100.

    This value is an administrator-defined target percentage, used in survey reports accessible by users of Oracle Business Intelligence. Reports compare the target percentage indicated against the total number of responses received for a deployment. Thus, the value is provided as a report metric, and is not used to enforce any survey campaign-level business rules.

40. Review the deployment detail selections you entered for this deployment. When satisfied, click **Apply**.

    The new deployment information is created, and information in all completed fields is saved to the database. The Updated Deployment page appears, displaying details for the current deployment.

    To modify deployment details or activate a deployment, proceed to step 6 in the task Viewing or Modifying Deployment Details.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.8 Viewing or Modifying Deployment Details

Use this procedure to view or modify details for an existing deployment, or to activate a deployment.

If performing this procedure immediately after creating or updating a deployment, begin with step 6.

Prerequisites

A survey campaign, cycle, and deployment must already be created.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

   - When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.
   - To filter these records, see Filtering the Display of Survey Campaigns.

3. In the list of survey campaigns, for the appropriate survey campaign, click **Update**.

   The Update Survey Campaign page appears

4. In the Cycles and Deployments area, if the cycle with which the desired deployment is associated is not displayed, then from the Cycle Name list, select the appropriate cycle and click **Go**.

   The Update Survey Campaign page refreshes.

5. In the Deployments area, from the Deployment Name column, click the appropriate deployment name.

   The Update Deployment page appears.

6. In the Update Deployment page, select **Deployment View**.

   The Update Deployment page refreshes. This page has two views. In the deployment view, you can modify deployment information or activate this deployment. In the response view, you can see the results of any responses received for this deployment.

7. If you want to view or modify deployment details, then do the following:

   a. Optionally, to view deployment details specific to targeted deployments (if not already displayed), then from the Deployment Type field, select **Targeted** and click **Go**.

      The Update Deployment page refreshes, displaying additional fields applicable to targeted deployments.

   b. Make any changes as appropriate.

   c. When satisfied, click **Update**.

8. If you want to activate this deployment, click **Activate**.

   > **Note:** This option is only available if the deployment is not already active.

   The Update Deployment page refreshes. For an active deployment, the only editable fields include Status and Response End Date and Time.

   For active standard deployments, in the Hosting Options area, the Survey Web URL field contains an active hyperlink to the executable deployment, including the message **Please click here to take the survey.**

   - For active standard deployments, in the Hosting Options area, the Survey Web URL field contains an active hyperlink to the executable deployment, including the message **Please click here to take the survey.**
   - For active targeted deployments, the Survey Web URL field contains the URL of the current Apache Web server session and port, Oracle Applications directory, and the appropriate JSP page based on hosting option. Since each respondent has an identification code included as a parameter in their unique survey URL, targeted deployments cannot list a single survey URL.
   - For targeted deployments for which the SUBMIT GROUP FM REQUEST FROM IES concurrent program has not yet executed, only the concurrent

request ID appears. Once the fulfillment request is executed and the page is updated, the fulfillment request ID also appears.

- If the fulfillment request ID does not appear, and the Activate button reappears, the concurrent program to submit the fulfillment request has failed.

9. If you want to participate in this survey:

   For standard deployments, access the URL in the Survey Web URL field.

   > **Note:** If you access this link using the same Web browser (for example, to test this deployment), you will invalidate your current Oracle Applications session. This will require a subsequent login to view details in the Survey Administration console.

   For targeted deployments, access an invitation e-mail message sent from the fulfillment server and access the unique URL it contains.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Deleting Deployments
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.9 Deleting Deployments

Use this procedure to delete one or more deployments from the Survey Administration console and from the applications database. Deleting a deployment does not otherwise affect the parent cycle, or its parent object, the survey campaign.

Prerequisites

- A deployment must exist as a child to a parent survey campaign and cycle before you can delete the deployment.
- You must know the name of the survey campaign, cycle, and deployment in order to delete the deployment.
- The deployment must not be active.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

   - When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.
   - To filter these records, see Filtering the Display of Survey Campaigns.

3. In the list of survey campaigns, for the appropriate survey campaign, click **Update**.

   The Update Survey Campaign page appears

4. In the Cycles and Deployments area, if the cycle with which the desired deployment is associated is not displayed, then from the Cycle Name list, select the appropriate cycle and click **Go**.

   The Update Survey Campaign page refreshes.

5. In the Cycles and Deployments area, in the Select column, check each deployment that you wish to delete.

> **Note:** To delete multiple deployments in one step, all deployments you want to delete must appear on the page. If you cannot adjust search criteria to view all relevant deployments, you must search for and delete deployments separately.

6. Click **Delete**.

   The Update Survey Campaign page refreshes.

7. Verify that the deployments you deleted no longer appear in the Deployments list.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Viewing Responses Received
- Sending Reminders Manually to List Members

8.2.10 Viewing Responses Received

Use this procedure to view individual responses to a script executed in a Web browser.

Prerequisites

- A survey campaign, cycle, and deployment must already exist.
- At least one deployment must be active.

- At least one individual must already have executed the appropriate script in a Web browser. For targeted survey campaign deployments, at least one list member must have responded by taking a survey.
- You must know the specific survey campaign, cycle and deployment for which you want to review individual responses.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

   - When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.
   - To filter these records, see Filtering the Display of Survey Campaigns.

3. In the list of survey campaigns, for the appropriate survey campaign, click **Update**.

   The Update Survey Campaign page appears

4. In the Cycles and Deployments area, if the cycle containing the desired deployment *is not* displayed, then from the Cycle Name list, select the appropriate cycle.

   The Update Survey Campaign page refreshes.

5. Optionally, if the cycle containing the desired deployment *is* displayed, then to view the deployments for this cycle and the amount of responses received for each, in the Cycles and Deployments area, select **Response View**.

   The Update Survey Campaign page refreshes. In the Deployments area, each deployment for the selected cycle is listed, with response-related statistics displayed.

6. From the deployments view *or* the response view, to view responses for a specific deployment, then from the list of deployments, for the appropriate deployment, click **Update**.

   The Update Deployment page refreshes.

7. In the Deployment Details area, select **Response View**.

   The Update Deployment page refreshes. Individual responses received for this deployment (if any) are listed in the Responses table.

8. In the Responses area, from the Answer column, click the **View Answers** icon.

   The Survey Responses page appears for this respondent. Below information for the survey campaign, cycle, deployment and the response date, the Answers table appears. This table contains an expandable navigation tree for the selected respondent.

9. To expand or collapse all navigation trees for each panel or question for this respondent, click **Expand Al**l or **Collapse All** at any time, as appropriate.

   In the fully expanded view, subsequent navigation trees appear for each panel and question. In the fully collapsed view, only respondent information is visible.

10. When collapsed, click the triangle marked with a plus sign to view respondent answer information hierarchically. The hierarchy, from top to bottom, is respondent, panel name, question in panel, answer choice in question (if relevant), and respondent's selected answer.

11. For any panel or question, from the Focus column, click the focus icon to focus in on that aspect of the response.

    An uncluttered view of the panel or question appears. Navigation links at the top of the table allow you to navigate back to the parent question or panel.

12. To view all the questions in a panel, click the appropriate panel's navigation tree.

    Additional rows appear, one for each question in the panel. Each row contains its own expandable navigation tree to view any answer choices or lookup values included in the question.

13. To view the answer choices available and see which choice the respondent made, click the appropriate question's navigation tree.

    The answer choice selected by this respondent has an entry in the Answer column of the table.

Additional rows appear, one for each question in the panel. For each answer choice or lookup value provided in this question, another row appears. For questions with only one possible answer choice, only one row appears.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Sending Reminders Manually to List Members

8.2.11 Sending Reminders Manually to List Members

Use this procedure to manually send a reminder message to a specified list member as part of a survey campaign deployment. This procedure results in initiating a fulfillment request to send out the reminder, just as if the reminder were scheduled. This will not interfere with the schedule for any currently scheduled reminders.

Prerequisites

- A survey campaign, cycle, and deployment must already exist.
- At least one deployment must be active.
- The active deployment must be targeted (include valid list criteria).
- The active deployment must include reminder information in the deployment definition.
- You must know, or be able to determine, the list member for whom you want to send reminders.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of survey campaign records displayed.

   - When a specified number of records exist (typically more than ten), you can navigate through the list using **Previous** or **Next** links.

   - To filter these records, see Filtering the Display of Survey Campaigns.

3. In the list of survey campaigns, for the appropriate survey campaign, click **Update**.

   The Update Survey Campaign page appears

4. In the Cycles and Deployments area, if the cycle containing the desired deployment *is not* displayed, then from the Cycle Name list, select the appropriate cycle.

   The Update Survey Campaign page refreshes.

5. Optionally, if the cycle containing the desired deployment *is* displayed, then to view the deployments for this cycle and the amount of responses received for each, in the Cycles and Deployments area, select **Response View**.

   > The Update Survey Campaign page refreshes. In the Deployments area, each deployment for the selected cycle is listed, with response-related statistics displayed.

6. From the deployments view *or* the response view, to view responses for a specific deployment, then from the list of deployments, for the appropriate deployment, click **Update**.

   The Update Deployment page refreshes.

7. In the Deployment Details area, select **Response View**.

The Update Deployment page refreshes. Individual responses received for this deployment (if any) are listed in the Responses table.

8. From the Send Reminder column, click the **Send Reminder** icon for the appropriate list member.

   The Update Deployment page refreshes. If successful, a confirmation message appears.

Note: Reminders can only be sent manually for active targeted deployments for which reminder details are established.

Guidelines

- A confirmation message when manually sending a reminder does not indicate that the reminder was successfully sent to the list member. Instead, this indicates that the fulfillment request is successfully sent to the fulfillment server.
- To confirm delivery, sign into Oracle One-to-One Fulfillment as a fulfillment administrator.

See Also

- Filtering the Display of Survey Campaigns
- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign and Cycle Details
- Cancelling a Survey Campaign from the Survey Campaigns List
- Deleting Survey Campaigns and Cycles
- Adding or Modifying a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Sending Reminders Manually to List Members

8.3 Setting Up Invitations and Reminders

When an enterprise conducts an Oracle Scripting survey campaign using one or more targeted deployments, members of an Oracle Marketing list are sent an *invitation* to participate in a survey or Web script. Each invitation is an e-mail message in HTML format, sent from the fulfillment engine (a component of Oracle One-to-One Fulfillment).

If enterprise requirements dictate, and a survey deployment is set up accordingly, subsequent e-mail messages can be sent (either to specific list members, manually, or to all list members, on a scheduled basis), reminding them of their eligibility to participate. These are *reminders*. Invitations and reminders are fulfillment master documents.

When a list member receives the invitation or reminder e-mail message, and enters the URL into an Oracle Applications-compatible Web browser (or clicks the link from an appropriately configured e-mail client), the designated survey questionnaire script executes in the client's Web browser. List members who respond to the request and execute an Oracle Scripting transaction are known as *respondents.*

Script execution in this manner comprises use of the Scripting Engine Web interface. Based on the intended purpose of the script and the details of the survey deployment, this is referred to either as *executing a Web script* or *taking a survey.*

From an administration perspective, Oracle One-to-One Fulfillment master document, query, and template administration are required for targeted (list-based) survey campaign deployments. When setting these up, each required HTML master document is created, and then associated with a survey list query. The query contains specific structured query language (SQL) statements which obtain from the Oracle Marketing list, at minimum, the party ID and e-mail address of the list member, and generates a functional, unique survey URL. Each master document is then associated with an Oracle One-to-One Fulfillment template.

When a survey administrator designates that the deployment type is targeted, additional deployment parameters can be defined. Among the required deployment details for a targeted deployment, the administrator specifies the Oracle Marketing list, identifies the template for the invitation, and specifies the e-mail subject line for the invitation. Optionally, if reminders are required, then the reminder template, reminder e-mail subject heading, number of reminders, and reminder interval information is specified.

When a targeted deployment is activated, a SUBMIT GROUP FM REQUEST FROM IES concurrent request is generated which is scheduled to run at the deployment start date and time. When the start date and time are equal to or past SYSDATE, the Oracle One-to-One Fulfillment Mass Request API is invoked. This API causes information to be sent to the fulfillment server, resulting in the execution of the survey list query. The records returned from the Oracle Marketing list are parsed, and for each list record, the merge data is inserted into an HTML e-mail message which is sent to the appropriate mail server and e-mailed to the list member. As a result, each list member is sent an invitation or a reminder to participate in the survey, including the appropriate survey URL.

Administration of master documents, queries, and templates is typically a function of a fulfillment administrator. This section includes a discussion of the theory and suggested Oracle One-to-One Fulfillment administration steps required for executing targeted survey deployments.

The Survey Administration console includes an Invitations tab. This tab hosts some functionality from Oracle One-to-One Fulfillment, including the ability to administer master documents, templates, and queries, and to view status and request history. (You must have the fulfillment administrator role to access this tab; for more information, see Granting JTF Roles.) The full functionality of Oracle One-to-One Fulfillment can be accessed using the Fulfillment Administration console.

For more information on any of the following topics, consult with the fulfillment administrator for your environment, or refer to *Oracle One-to-One Fulfillment Implementation Guide.*

Note: The information contained in this section is the most current information available at the time of publication. Every effort is made to synchronize between the releases of Oracle Scripting and the various applications with which it integrates. Nonetheless, if the steps listed in this document do not match the user interface you see when you attempt to perform a task, please consult the product documentation appropriate to the release installed at your site.

This section includes the following topics:

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps

- Invitations
- Reminders
- Sample Invitation Master Document
- Sample Reminder Master Document
- Seeded Survey List Query
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

Concepts

See Also

- Survey Home
- Administering Survey Campaigns
- Administering Concurrent Programs for Survey Execution

8.3.1 Oracle One-to-One Fulfillment Terminology

A **template** is a high-level definition of one or more Oracle One-to-One Fulfillment master documents, associated queries, and any related collateral. Templates may also include campaign specifications. From an Oracle Scripting perspective, targeted survey deployments call upon a template that identifies an invitation master document (and its associated query). If reminders are also required for the survey deployment, and the text of the reminder document differs from the invitation, a second template defining the reminder master document and its associated query is required. No collateral is typically associated with a survey campaign.

A **master document** is a message sent from Oracle One-to-One Fulfillment. These typically include merge fields populated by an associated SQL query. From a survey perspective, master documents are invitation or reminder messages in HTML format that are sent by e-mail from the fulfillment engine to members of a defined Oracle Marketing list.

Merge fields are data elements in an Oracle One-to-One Fulfillment master document surrounded by merge field delimiters. When a fulfillment request is executed, a query is performed to retrieve the merge data from the table columns or views specified in the query. For each record in the list, these merge fields are replaced with the appropriate data obtained by the query, as a separate HTML document sent to the e-mail server for appropriate routing. *For each merge field used in a master document, there must be one corresponding request in the query* to retrieve the appropriate data. (The converse is *not* true; in other words, the query can request information that is not used as a merge field.)

Merge fields in *any* survey list query must include, at minimum, the survey URL. Providing that the data is returned by the associated query, other merge fields can be included in the relevant master document if desired. For example, master documents using the seeded survey list query may also include as merge fields the first and last name, e-mail address, and customer ID (referred to in TCA as the party ID) of the list member.

Queries are database requests for information constructed in structured query language (SQL). Each Oracle One-to-One Fulfillment master document must have an associated query which obtains information populated into the resulting merged document in placeholders called merge fields. A query must contain a request for each data element included as a merge field in the master document. The query may contain requests for additional data elements as well (providing the requests are valid), regardless of whether the data elements requested in the query are used. Thus, for example, you can construct a single query that requests all information used to populate an invitation master document and a reminder master document, even if they include different merge fields. A single query can be associated with any number of master documents. Return data from a successfully executed query is merged with the master document. In this process, the master document is replicated once for each row returned. For every replicated copy, each merge field placeholder is replaced with its corresponding data element from the query. The amount of merged master documents will match the number of list records in the Oracle Marketing list, one per list member. Each merged document is then sent using e-mail to the e-mail address of each list member by the fulfillment server.

The **Seeded Survey List Query** is a SQL query entitled *IES Template Query for Targeted Survey Deployments*. This query is seeded with an appropriate installation of Oracle Interaction Center Family Pack R or later release, or Oracle Scripting release 11.5.10 or later. This query will return, for each list member, the following data elements, all of which may be used as a merge field in an appropriate master document:

Data Element	Source Table or View	Source Column	Merge Field Name	Description
Survey URL	IES_SVY_HOSTING_ PROF_V	SURVEY_URL	SURVEY_URL	This provides the JSP template used. Data from this column is concatenated with text strings and values from other columns to construct a valid Survey URL.
Survey URL	IES_SVY_ DEPLYMENTS_V	SURVEY_ DEPLOYMENT_ID	SURVEY_URL	This provides the deployment ID of the active deployment. Data from this column is concatenated with text strings and values from other columns to construct a valid Survey URL.
Survey URL	IES_SVY_LIST_ ENTRIES_V	RESPONDENT_ID	SURVEY_URL	This provides the respondent ID of the list member, which tracks to a party ID. Data from this column is concatenated with text strings and values from other columns to construct a valid Survey URL.
Email address	AMS_LIST_ENTRIES	EMAIL_ADDRESS	EMAILADDRESS	E-mail address of list member. Usually not included in the master document, but passed to the fulfillment server for appropriate routing.
Party ID	AMS_LIST_ENTRIES	PARTY_ID	CUSTOMER_ID	The party_id of the list member, often referred to as a customer ID.
First name	AMS_LIST_ENTRIES	FIRST_NAME	FIRST_NAME	Not a required field. The list member's first name.
Last Name	AMS_LIST_ENTRIES	LAST_NAME	LAST_NAME	Not a required field. The list member's last name.

See Also

- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps
- Invitations
- Reminders
- Sample Invitation Master Document
- Sample Reminder Master Document
- Seeded Survey List Query
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

8.3.2 Fulfillment Activities for Targeted Survey Deployments

For ease of administration, survey administrators have access to an Invitations tab in the Survey Administration console. This tab gives access to the Oracle One-to-One Fulfillment functions required for performing survey administration for targeted deployments.

From the Invitations tab, survey administrators can administer fulfillment master documents, templates and queries, and verify fulfillment request status and history.

Some Functions Not Available to Survey Administrators

Administration of fulfillment servers or fulfillment groups is intended to be performed by a knowledgeable fulfillment administrator. As such, these functions (available in the Fulfillment Administration console using the Server and Group tabs) are excluded from the user interface in the Survey Administration console. Additionally, if fulfillment requests have successfully been sent to the fulfillment server but have failed to execute, a fulfillment administrator is required to determine the cause and to correct the error. A targeted deployment with a status of Active indicates only that the fulfillment request was successfully submitted to the server, and does not indicate successful transmission of the invitation or master document e-mail messages.

Since Oracle One-to-One Fulfillment implementation and functionality is a prerequisite of survey operations, a fulfillment administrator should already exist in the enterprise who has access to the Fulfillment Administration console to perform these administration steps.

Required Tasks

From an Oracle One-to-One Fulfillment perspective, you need to accomplish the following tasks to create and administer invitations and reminders:

- Create and format a master document in HTML
- Insert merge field references into the master document
- Upload the master document to MES
- Create a SQL query to populate merge fields into master document
- Associate the query to a specific master document
- Create a fulfillment template that identifies a master document and its associated query

See Also

- Oracle One-to-One Fulfillment Terminology
- Recommended Order of Steps
- Invitations
- Reminders
- Sample Invitation Master Document
- Sample Reminder Master Document
- Seeded Survey List Query
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

8.3.3 Recommended Order of Steps

Three Approaches

If you have already created an HTML document to serve as your invitation or reminder master document, you can upload this to MES from the Survey Administration console (Invitations tab > Master Document subtab > **Upload**). Then you can associate the master document with a query you have created or identified, and then create a template that includes that master document (the query is associated at the master document level).

Alternatively, Oracle One-to-One Fulfillment includes a master document editor. Using this feature from the Survey Administration console (Invitations tab > Master Document subtab > **Create and Upload**), you can quickly and easily perform most tasks associated with the creation of invitation or reminder master documents and their requisite queries. In sequence, you can create a SQL query, validate the query, create and format a customized master document, insert merge fields identified by the query, and automatically associate the query with the master document. Following this, you simply create a template, associating it with your master document.

The third approach is to combine the strengths of the previous two approaches described above by creating a master document using the master document editor (Invitations tab > Master Document subtab > **Create and Upload**), but associating it with an existing query (instead of creating a new SQL query as part of the master document creation process). For this approach, you must insert custom merge fields into your master document. Each corresponds to the alias or merge field data returned by the associated SQL query. If you want to take advantage of the seeded survey list query, but do not yet have a master document, this approach is appropriate.

Order of Steps

The recommended order of steps for administering invitations and reminders using Oracle One-to-One Fulfillment functionality is as follows:

1. Identify merge fields to be populated in the master document, and identify locations of tables or views in the Oracle Applications database schema from which the merge fields will be populated. At minimum, you must include the survey URL as a merge field.
2. Identify an existing SQL query, or create a SQL query, including a request for any data corresponding to a merge field in the master document. At minimum, the query must include the party ID and e-mail address of the list member, as well as SQL to obtain the appropriate survey URL.
   - If desired, you can use the seeded IES Template Query for Targeted Survey Deployments query, which also includes the first and last names of the list member.
   - Oracle Corporation strongly recommends following a defined format for the column aliases and content used for a survey list query. At minimum, syntax from the seeded query indicating column names and aliases for the three required columns (party_id, email_address and the survey URL) must

be included in your query to ensure the appropriate data is returned and processed correctly.

- An example of a query using this format is the seeded IES Template Query for Targeted Survey Deployments. This query includes the three required data elements (party ID, e-mail address, and survey URL), plus the list member's first and last name.

  To create a custom query, Oracle Corporation recommends that you copy all of the content of the SELECT clause of the seeded survey list query, and add additional columns to the SELECT clause if required. (If not included in your master document, you can remove the SQL to obtain the list member's first and last names.) Then, add additional constraints to the WHERE clause as needed.

- A single query can be used for any number of master documents, providing that the query retrieves the data for each required merge field in each master document associated with the query.

3. Create invitation and reminder master documents in HTML using a text editor or HTML tool of your choice, or the Oracle One-to-One Fulfillment master document editor.

   If using the master document editor with an existing query, specify each merge field as a custom merge field by entering the column or alias name in the Custom Merge Field field and clicking **Insert**.

4. Upload master documents to MES.

   - If you have a custom HTML master document, upload it from the Survey Administration console (Invitations tab > Master Document subtab > **Upload**) or from the Fulfillment Administration console (Template tab > Master Document subtab > **Upload**).
   - If you created your master document using the master document editor feature of Oracle One-to-One Fulfillment, this step is performed automatically.

5. Associate each required master document with its query.

   - On the Master Document page, locate the appropriate master document, select the existing query from the Query list, and click **Update**.
   - If defining the query using the master document editor feature of Oracle One-to-One Fulfillment, the association between master document and query is performed automatically.

6. Define templates.

   - One template is required for each master document. If a survey campaign includes invitation and reminder documents with the same text, only one template need be created. If a survey campaign includes invitation and reminder documents with *different* content, create one template for each master document.
   - To define a template, from the Invitations > Template subtab, click **Create**, enter a name and optional information, and click **Continue**.

7. Associate template with the required master document.

   From the View Template page, in the Master Documents area, search for and select the invitation or reminder master documents you want to associate with this template (as requirements dictate). When satisfied, click **Update**.

References

For more information, refer to *Oracle One-to-One Fulfillment Implementation Guide.*

See Also

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Invitations
- Reminders
- Sample Invitation Master Document
- Sample Reminder Master Document
- Seeded Survey List Query
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

8.3.4 Invitations

Typically, an invitation is an HTML message inviting list members to take a survey. The invitation is referred to in Oracle One-to-One Fulfillment as a master document. Using Oracle One-to-One Fulfillment APIs, invitations can be customized by populating data from RDBMS tables or views.

The HTML invitation contains placeholders called **merge fields**, in which information from a query is placed by the fulfillment server upon execution. Each invitation must include, at minimum, a unique survey URL. Based on the deployment ID (dID) and respondent ID (rID) numbers, a unique URL is generated for each list member. In this way, if a survey campaign has a business rule to allow list members to only take a survey one time, this rule can be enforced. Other merge fields often used include the list member's first and last name and identifying information such as customer ID number.

See Also

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps
- Reminders
- Sample Invitation Master Document
- Sample Reminder Master Document
- Seeded Survey List Query
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

8.3.5 Reminders

If required by a survey campaign, reminders can also be sent to list members. Like the invitation, a reminder is an e-mail message identified in Oracle One-to-One Fulfillment as a master document. It can be the same e-mail message initially sent to list members (in which case, only a single template is required), or it can be a different message (requiring a second template to identify the master document and its associated query).

Reminders typically emphasize the response end date of the specific deployment for that survey campaign. Like invitations, each reminder must include the survey URL as a merge field, and must have a corresponding request for that data field in its associated query.

Unlike invitations, reminders are an optional feature. In addition to being sent automatically (based on reminder requirements defined at the deployment level), reminders can be sent by a survey administrator manually upon request.

If using the same template and master document, you have the option of using a different e-mail message subject. If using a separate e-mail master document as the reminder, you may choose to word it differently to emphasize the deadline.

Since queries can be defined once and used many times, you can use for the reminder master document the same query used by the invitation. This is true even if the reminder requires merge fields not used in the invitation master document. For example, in a reminder you may want to emphasize the deadline by which list members must participate in the survey (this equates to the deployment end date). Even if this information was not included in the original invitation, a single query can be shared among the two HTML documents; the only information returned by the query that is *required* to be included in the invitation is the survey URL.

See Also

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps
- Invitations
- Sample Invitation Master Document
- Sample Reminder Master Document
- Seeded Survey List Query
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

8.3.6 Sample Invitation Master Document

The following is a sample invitation master document. Master documents for your enterprise may appear substantially different. For example, this sample includes four merge fields (signified below by open and close guillemet characters, otherwise known as angle brackets).

Upon delivery, for each list member, one e-mail message is generated, containing the first and last name of the list member, their customer number, and the unique survey URL for this deployment. All of these merge fields are included in the seeded query IES Template Query for Targeted Survey Deployments (described in the Seeded Survey List Query section). The only merge field *required* in the master document is the survey URL.

Sample Invitation Text

Following is the sample text of the e-mail message:

> Dear «FIRST_NAME» «LAST_NAME»:
>
> Thank you for your recent order on account «CUSTOMER_ID». We appreciate your business, and hope you enjoyed a positive customer experience.
>
> To help us evaluate our product offerings and ensure the highest levels of customer service, please take a moment to participate in this survey regarding your recent purchase. Click «SURVEY_URL» to begin (or type this URL into your Web browser).
>
> Sincerely,
>
> Vision Customer Service

Sample Invitation Code

Following is the HTML code of the above e-mail message master document. Note that the open and close guillemet characters which surround merge fields are designated in the HTML sample using ASCII references.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 TRANSITIONAL//EN">
<HTML>
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="TEXT/HTML">
<META NAME="Author" CONTENT="Vision Customer Service">
<TITLE>Customer Service Survey Invitation</TITLE>
</HEAD>

<BODY BGCOLOR="WHITE">

<P>Dear &laquo;FIRST_NAME&raquo; &laquo;LAST_NAME&raquo;:</P>

<P>Thank you for your recent order on account &laquo;CUSTOMER_
ID&raquo;. We appreciate your business, and hope you enjoyed a
positive customer experience.</P>

<P>To help us evaluate our product offerings and ensure the
highest levels of customer service, please take a moment to
participate in this survey regarding your recent purchase.
Click &laquo;SURVEY_URL&raquo; to begin (or type this URL into
your Web browser).<P>
```

```
<P>Sincerely,</P><BR>

<P>Vision Customer Service</P>

</BODY>
</HTML>
```

See Also

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps
- Invitations
- Reminders
- Sample Reminder Master Document
- Seeded Survey List Query
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

8.3.7 Sample Reminder Master Document

The following is a sample reminder master document. Master documents for your enterprise may appear substantially different. In this sample, the reminder uses three of the four merge fields used in the original invitation, and is associated with the same query. Note, however, that a reminder can include different merge fields, as long as the associated query provides that information.

Sample Reminder Text

Following is the sample text of the e-mail message:

> Dear «FIRST_NAME» «LAST_NAME»,
>
> Several weeks ago, you were sent an invitation to participate in our customer satisfaction survey. At the end of this month, we will complete the survey process, and collate the results to see how to best improve customer satisfaction.
>
> If you would like your opinions to be considered, please click «SURVEY_URL» to participate in this online survey using your Web browser.

As always, we appreciate your business and look forward to your opinions to enable us to improve our responsiveness.

Sincerely,

Vision Customer Service

See Also

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps
- Invitations
- Reminders
- Sample Invitation Master Document
- Seeded Survey List Query
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

8.3.8 Seeded Survey List Query

As of Oracle Scripting release 11.5.10 or later, or Interaction Center Family Pack R or later, Oracle Scripting provides a new seeded query (IES Template Query for Targeted Survey Deployments) that is included in the list of Oracle One-to-One Fulfillment queries.

Upon processing, this query provides data for the «FIRST_NAME», «LAST_NAME», «CUSTOMER_ID» and «SURVEY_URL» merge fields, as used in the above master document samples.

Each master document may have its own query. Conversely, the same query can be used any number of times for various master documents. To be successful, the query must contain a structured query language (SQL) request for each merge field in each associated master document. A single query may contain multiple SQL requests, including requests for data that are *not* used in your master document. The SQL in the query must be properly constructed to return the requested data from tables or views, regardless of whether the data is required for your master document.

Following is the content of the seeded query:

```
select
ale.first_name  FIRST_NAME,
ale.last_name  LAST_NAME,
'<a href='||ihp.SURVEY_URL||' dID='||asda.survey_deployment_id
||'&rID='||isle.respondent_id||'> here </a>' SURVEY_URL,
ale.party_id CUSTOMER_ID,
ale.email_address EMAILADDRESS
from
IES_SVY_LIST_ENTRIES_V isle,
ams_list_entries ale,
IES_SVY_DEPLYMENTS_V asda,
IES_SVY_HOSTING_PROF_V ihp
where
ale.list_entry_id = isle.list_entry_id
and isle.survey_deployment_id =
asda.survey_deployment_id
and ihp.survey_deployment_id = asda.survey_deployment_id
and asda.survey_deployment_id = :deployment_id
```

Oracle Corporation strongly recommends following the format used above for the column aliases and content in any survey list query.

To customize your own survey query, Oracle Corporation recommends the following:

1. Copy all of the content of the SELECT clause of the seeded survey query.
2. Optionally, if list member names are not required, remove the appropriate SQL code requesting this information.
3. Following the format above, in the SELECT clause, add relevant SQL code to return additional merge field data.
4. Following the format above, in the WHERE clause, add relevant SQL code to impose additional constraints on the return data, if required.

New Requirements for Enterprises Upgrading to Release 11.5.10 or Later

Existing Oracle Scripting implementations using targeted survey deployments must modify the query or queries in use when upgrading to this release or later. From this release forward, each survey list query must contain, at minimum, the select clause in the query below. Other fields must be added as necessary.

1. Master documents must be associated with an updated query to ensure continuous, uninterrupted function. If merge fields are restricted to the

SURVEY_URL and any or all of the following merge fields (FIRST_NAME, LAST_NAME, and CUSTOMER_ID), you can associate the invitation or reminder master document with the seeded IES Template Query for Targeted Survey Deployments query. No other changes are needed.

2. When creating custom queries, Oracle Corporation strongly recommends that you preserve the exact syntax from IES Template Query for Targeted Survey Deployments to ensure that the survey URL will be inserted into the invitation or reminder e-mail message.

3. Master documents that contain merge fields in addition to the three mentioned above will require changes as appropriate. Oracle Corporation recommends replacing your current query SELECT statement with the updated language in the seeded query, and then customizing it to select data for the other merge fields or to add new constraints, as required. Then associate this revised query to each master document as appropriate.

See Also

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps
- Invitations
- Reminders
- Sample Invitation Master Document
- Sample Reminder Master Document
- Using the Master Document Editor
- Verifying Status of a Fulfillment Request

8.3.9 Using the Master Document Editor

You can use the Master Document Editor feature of Oracle One-to-One Fulfillment to create a SQL query, create a master document (automatically stored in the MES knowledge repository) to use as an invitation or reminder, associate the query automatically with the master document, and add merge fields enabled by the query to the master document.

Alternatively, if you want to create a new master document but associate it with an existing query, you can create the master document, insert custom merge fields, and then (from the Master Document page) associate the appropriate existing query to a listed master document.

Use the following procedure to create a new master document and associate an existing query with it, or to create a SQL query, create and format a new master document, and include in the master document merge fields from the query.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator or One-to-One Fulfillment Administrator

Prerequisites

- A fulfillment server must already exist.
- User must have the JTF_FM_ADMIN role to perform this function.

Steps

1. If using the Fulfillment Administration console, click the Template tab.
2. If using the Survey Administration console, click the Invitations tab.
3. Click the Master Document subtab.

   The Master Document page appears.
4. From the Master Document page, click **Create and Upload**.

   The Master Document Editor page appears.
5. In the Master Document Name field, type a unique name for the master document.
6. If you want to define a new master document but associate it to an existing SQL query, skip to step 8.
7. If you want to define a new SQL query, associate the query to this master document, and use merge fields identified in the query in your master document, do the following:

   a. In the SQL Merge Field field, click **Create Query**.

      The SQL Editor page appears.

b. In the Query Name field, type a unique name for this query.

c. Optionally, in the Query Description field, type a description for this query.

d. In the SQL Query field, using well-formed structured query language, type or paste the contents for your query. It must include, at minimum, SQL code to obtain a valid survey URL.

   For example, to create a query to obtain a list member’s first name, last name, e-mail address, party ID, and valid survey URL, copy the contents of the sample query from Section 8.3.8, "Seeded Survey List Query" and paste into this field.

e. Click **Validate SQL and Save**.

   The SQL Editor page refreshes. For any merge field that requires a constraint to be defined, you will be directed to enter the constraint value or range.

f. If prompted, enter constraints for each merge field as appropriate. When finished, click **Validate SQL and Save**.

   - The Master Document Editor page refreshes.
   - The query criteria you entered is visible in the SQL Query area.
   - All merge fields available based on the query you entered are now accessible in the SQL Merge Field list.

8. If you want the master document to appear in plain text, click **Change to Plain text mode**.

   For example, if your master document contains no images or character-level formatting, or if the recipients of your master document will not be receiving the merged document in e-mail capable of interpreting HTML format, select plain text mode.

9. If you want the master document to contain formatting you can apply using the master document editor’s rich text format (RTF) formatting tools, keep the default option. To switch modes from plain text mode, select **Change to Rich text mode**.

10. In the Master Document Body field, type the text for your master document.

11. At each appropriate location in the body of your master document, add merge fields to your master document as appropriate.

Note: At minimum, your invitation or reminder master document must contain at least one merge field to generate a survey URL.

12. If entering merge fields based on a SQL query you created on this page, then do the following:

    a. From the SQL Merge Field list, select the appropriate merge field.

    b. Click **Insert**.

    c. Fix any spacing errors in the master document resulting from the inclusion of this merge field.

13. If entering custom merge fields, then do the following:

    a. In the Custom Merge Field field, type the precise merge field name for a value returned from a query.

       For example, type **SURVEY_URL**.

    b. Click **Insert**.

    c. Fix any spacing errors in the master document resulting from the inclusion of this merge field.

       Each merge field appears in the master document body. Merge fields are the fields available from a SQL query.

14. Verify the content and accuracy of your master document.

15. Optionally, if using rich text mode, apply formatting to your master document by selecting text and clicking the appropriate formatting tools.

16. When you have finished, click **Save**.

    The Master Document Editor page closes. The Master Document page appears. The master document you have just created and uploaded to MES is included in the list.

    If you defined a SQL query for this master document, you have successfully completed this task.

    If you simply defined a master document, you must now associate an existing SQL query with the new master document by completing the steps below.

17. For the appropriate master document, from the Query column, select the query you want to associate to this master document

For example, to associate the seeded query, select IES Template Query for Targeted Survey Deployments.

18. Click **Update**.

See Also

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps
- Invitations
- Reminders
- Sample Invitation Master Document
- Sample Reminder Master Document
- Seeded Survey List Query
- Verifying Status of a Fulfillment Request

8.3.10 Verifying Status of a Fulfillment Request

The Oracle One-to-One Fulfillment Request Status page (accessed from the Survey Administration console by clicking Invitations tab > Status subtab) indicates the status of fulfillment requests. A request status of **Submitted** indicates the fulfillment request has been successfully submitted to the fulfillment engine.

> **Note:** A successfully submitted request does not necessarily signify that the engine succeeded in delivering the master document in question. This status only indicates that the request was sent to the fulfillment engine as appropriate.

When checking the status of requests in the Request Status page, valid states include Submitted and In Process. If the request remains in process for an undue length of time, or if the status displayed is **Stalled**, consult with a fulfillment administrator.

To verify that the fulfillment engine successfully *completed* distribution of a fulfillment request, check for an outcome code of **Successful** in the Request

Summary page. Ascertain the status in the Outcome column from the Survey Administration console by clicking Invitations tab > History subtab.

If any status other than **Successful** displays in the Request History page, consult with a fulfillment administrator.

References

For specific information on implementing, using, or administering Oracle One-to-One Fulfillment, refer to *Oracle One-to-One Fulfillment Implementation Guide.*

See Also

- Oracle One-to-One Fulfillment Terminology
- Fulfillment Activities for Targeted Survey Deployments
- Recommended Order of Steps
- Invitations
- Reminders
- Sample Invitation Master Document
- Sample Reminder Master Document
- Seeded Survey List Query
- Using the Master Document Editor

8.4 Administering Concurrent Programs for Survey Execution

Concurrent programs are executed using Oracle Concurrent Manager, a Forms-based application built on Oracle Application Object Library (AOL) classes. Concurrent Manager is included in Oracle Applications 11*i* with an appropriate Rapid Install. Seeded concurrent programs can be scheduled to execute at certain times, or can be administered to execute in near-real time. To schedule or execute concurrent programs supporting Oracle Scripting's survey component, you must access Oracle Forms-based applications with a user assigned the iSurvey User responsibility.

Seeded Concurrent Programs for Survey Execution

Oracle Scripting uses three seeded concurrent programs to support survey component operations. The table below briefly describes these programs:

Concurrent Program	Survey function supported	Recommended Execution Method
SUBMIT GROUP FM REQUEST FROM IES Concurrent Program	Targeted survey deployments	■ Automatically (typical) ■ Manually (when Fulfillment Request ID fails to generate)
Summarize Survey Data Concurrent Program	Web interface reports available using Oracle Interaction Center Intelligence	■ Scheduled (typical - set up to execute on a periodic basis or on specific days) ■ Manually (on demand when fresh reporting tabulation is desired)
Update Deployment Status Concurrent Program	Survey Administration console table updates	Scheduled (set up to execute on a periodic basis or on specific days, generally once daily)

Tasks

- Scheduling or Executing Concurrent Programs
- Checking Concurrent Program Logs

Concepts

For more information, see the following concepts:

- SUBMIT GROUP FM REQUEST FROM IES Concurrent Program
- Summarize Survey Data Concurrent Program
- Update Deployment Status Concurrent Program

See Also

- Survey Home
- Administering Survey Campaigns
- Setting Up Invitations and Reminders

8.4.1 Scheduling or Executing Concurrent Programs

Use the following procedure to schedule or execute concurrent programs supporting survey operations for Oracle Scripting implementations.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

iSurvey User

Steps

1. From the Navigator, select **Survey Administratin > Requests > Run** and click Open.

   The Submit a New Request window appears.

2. From the Submit a New Request window, select **Single Request** and click **OK**.

   The Submit a New Request window closes, and the Submit Request window appears.

3. In the Run this Request... area, in the Name field, click the ellipsis points.

   The Reports window appears, listing the concurrent programs accessible to this responsibility.

4. To execute the Fulfillment concurrent program, from the Reports window, select SUBMIT GROUP FM REQUEST FROM IES and click **OK**.

   To execute the Summarize Survey Data concurrent program, from the Reports window, select **Summarize Survey Data** and click **OK**.

   To execute the Update Deployment Status concurrent program, from the Reports window, select **Update Deployment Status** and click **OK**.

   - The Reports window closes.
   - The Name field is populated with the selected report type in the Submit Request window.
   - The Parameters window appears and has focus.

5. If scheduling or executing the Update Deployment Status concurrent program, no parameters are required. Proceed to 7.

6. For other concurrent programs, enter the required parameters as prompted.

Note: You must enter parameter information using the appropriate data type. You cannot use the % wildcard character in this field.

See Seeded Concurrent Programs for Survey Execution for parameter information.

Optionally, to complete this field, you can search existing information in the database for each parameter. To do this:

a. Click the ellipsis points next to the parameter field.

   The Longlist window appears, indicating that the displayed list is lengthy.

b. In the Longlist window, in the Partial Value field, enter search criteria.

Note: You can use the % wildcard character in this field.

c. If you entered valid criteria, the Longlist window closes and the parameter field in the Parameters window is populated with your selection. The corresponding description field of your selection appears next to the field. To re-query, return to a.

d. If you entered partial search criteria, refine your search in the resulting window and click **OK**.

   The Parameters window refreshes, populated with your selection.

e. After entering all required parameters, click **OK**.

   The Parameters window closes. The Submit Request window refreshes, populated with your parameter selections.

7. If a default choice to execute the concurrent program is selected in the At these Times... area and meets your scheduling or execution requirements, proceed to 14.

8. If you want to schedule the concurrent program to execute, in the At these Times... area, click **Schedule**.

   The Schedule window appears.

9. If there is a saved schedule you want to use:

   a. Click **Apply a Saved Schedule.**

The Pre-defined Schedules window appears.

b. From the Pre-defined Schedules window, select the appropriate schedule.

c. Click **OK**.

The Schedule window refreshes, with the Advanced option selected in the Run the Job... area.

d. Optionally, to increment date parameters for each execution of the concurrent program, from the Schedule window, select **Increment date parameters each run**.

e. Optionally, to save the schedule, from the Schedule window, select **Save this schedule**.

f. To save and apply schedule details, from the Schedule window, click **OK**.

10. If you want to execute the concurrent program as soon as possible, from the Run the Job... area, select **As Soon as Possible** and click **OK**.

11. If you want to execute the concurrent program one time at a specified date:

a. In the Run the Job... area, select **Once**.

The Schedule window refreshes, displaying a Run At field.

b. Click the ellipsis points in the Run At field.

The calendar window appears.

c. From the calendar window, specify the date and time to execute the concurrent program, and click **OK**.

d. Optionally, to increment date parameters for each execution of the concurrent program, from the Schedule window, select **Increment date parameters each run**.

e. Optionally, to save the schedule, from the Schedule window, select **Save this schedule**.

f. To save and apply schedule details, from the Schedule window, click **OK**.

12. If you want to execute the concurrent program several times periodically:

a. In the Run the Job... area, select **Periodically**.

The Schedule window refreshes, displaying new fields.

b. In the Start At field, click the ellipsis points.

The calendar window appears.

c. From the calendar window, specify the date and time to start executing the concurrent program, and click **OK**.

d. If you want the concurrent process to execute periodically for an indefinite period, do not enter a value in the End At field.

e. To specify an end date, in the End At field, click the ellipsis points.

   The calendar window appears.

f. From the calendar window, specify the date and time to stop executing the concurrent program, and click **OK**.

g. In the Re-run fields, specify the value and interval for the period of time between executions.

h. In the Apply the Interval... area, determine the criteria for applying the interval.

i. Optionally, to increment date parameters for each execution of the concurrent program, from the Schedule window, select **Increment date parameters each run**.

j. Optionally, to save the schedule, from the Schedule window, select **Save this schedule**.

k. To save and apply schedule details, from the Schedule window, click **OK**.

13. If you want to execute the concurrent program on specific days:

   a. In the Run the Job... area, select **On Specific Days**.

      The Schedule window refreshes, displaying new fields.

   b. In the Start At field, click the ellipsis points.

      The calendar window appears.

   c. From the calendar window, specify the date and time to start executing the concurrent program, and click **OK**.

   d. To specify an end date, in the End At field, click the ellipsis points.

      The calendar window appears.

   e. From the calendar window, specify the date and time to stop executing the concurrent program, and click **OK**.

   f. In the Dates of Every Month area, click once on each day of the month in the calendar on which you want to execute the program. To deselect a calendar date, click it again.

g. In the Days of Every Week area, click once on each day of the week on which you want to execute the program. To deselect a week day, click it again.

h. Optionally, to increment date parameters for each execution of the concurrent program, from the Schedule window, select **Increment date parameters each run**.

i. Optionally, to save the schedule, from the Schedule window, select **Save this schedule**.

j. To save and apply schedule details, from the Schedule window, click **OK**.

14. In the Submit Request window, in the Upon Completion... area, check the **Save all Output Files** box.

15. Optionally, if you want a party to be notified each time the concurrent program is executed, then do the following:

    a. In the Upon Completion... area, click **Options**.

       The Upon Completion window appears.

    b. Verify that the **Save all Output Files** box is checked.

    c. In the Name column, for each party whom you want to notify upon completion, click the ellipsis points and select the individual from the database.

    d. Optionally, to print output files, select a networked printer from the Printer field, and specify the number of copies and default language.

    e. To save Upon Completion information, click **OK**.

       The Upon Completion window closes. The Submit Request window refreshes, displaying the selections you made.

16. To save and submit the concurrent request, in the Submit Request window, click **Submit**.

    The Decision window appears, providing the request ID of the concurrent request. Note this request number for subsequent tracking.

17. To submit another request, in the Decision window, click **Yes**, and then enter criteria for the next request in the Submit Request window.

    Otherwise, click No. The Decision window and the Submit Request window close. The Navigator appears.

18. Submit another request for concurrent processing by repeating this procedure, or log out of Oracle Applications by selecting **File > Exit Oracle Applications**.

References

- For specific details on executing or scheduling concurrent programs, please refer to Chapter 5 of *Oracle System Administrator's Guide*.
- For more information, see Seeded Concurrent Programs for Survey Execution.

See Also

- Checking Concurrent Program Logs

8.4.2 Checking Concurrent Program Logs

Concurrent program logs can provide clues for the failure of the concurrent program.

For example, when you activate a survey deployment, the SUBMIT GROUP FM REQUEST FROM IES concurrent request is generated (scheduled to run at the deployment start date and time). If the deploy start date and time are in the past, the concurrent program executes immediately. If this concurrent program fails, the concurrent request ID appears on the Update Deployment page, but no fulfillment request ID appears, and the Activate button reappears. In such an event, check the concurrent program logs to determine potential contributing factors for the failure of this request to execute.

Use this procedure to check concurrent program logs.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

iSurvey User

Steps

1. From the Navigator, select **Survey Administratin > Concurrent** and click Open.

The Find Requests window appears.

2. If you want to find only completed requests, select My Completed Requests.
3. If you want to find requests in process, select My Requests in Process.
4. If you want to find all requests generated using your Oracle applications user account, select All My Requests.
5. If you want to find a specific request, select Specific Requests, and enter additional search criteria to find the appropriate request.
   a. To search a concurrent request by ID, type the concurrent request ID in the Request ID field.

      For example, for a failed SUBMIT GROUP FM REQUEST FROM IES concurrent request, obtain the concurrent request ID from the Update Deployment page and type this value.
   b. To search a concurrent request by name, type the name of the concurrent request in the Name field.

      For example, type SUBMIT GROUP FM REQUEST FROM IES.
   c. To search a concurrent request by date submitted, in the Date Submitted field, click the ellipsis points, and specify the appropriate date using the calendar control.
   d. To search a concurrent request by date the request was completed, in the Date Completed field, click the ellipsis points, and specify the appropriate date using the calendar control.
   e. To search a concurrent request by the request status, from the Status list, select the appropriate status.

      For example, select Suspended.
   f. To search a concurrent request by the request phase, from the Phase list, select the appropriate phase.

      For example, select Inactive.
   g. To search a concurrent request based on the Oracle Applications account of the individual requesting the request, in the Requestor field, click the ellipsis points, and search for and select the appropriate user name.

      For example, select Inactive.
6. Select the Include Request Set States in Query option.

7. Optionally, to sort records returned, from the Order By list, select order criteria.

   For example, to sort by concurrent program name, select Name.

8. After entering your search criteria, click **Find**.

   The Requests window appears, populated with the results of your query.

9. Optionally, to ensure you are accessing the correct record, do the following:

   a. Click on the row of the appropriate record and then click View Details.

      The Request Detail window appears for the selected request.

   b. To close the Request Detail window, click OK.

10. To view the message log, click on the row of the appropriate record and then click View Log...

    A new Web browser window will open, displaying the message log for the selected concurrent program.

See Also

- Scheduling or Executing Concurrent Programs

9

Campaign and List Administration Tasks

Oracle Scripting interacts with various components in the Oracle Applications suite. Although Scripting can be executed alone, it is typically used in combination with various other applications. For example, Scripting is typically launched from an Oracle business application. If using Oracle Scripting from Oracle TeleSales or Oracle Collections, you must first administer campaigns in Oracle Marketing and Oracle Sales Online.

Additionally, in order to execute survey campaigns using the survey component of Oracle Scripting, Oracle Marketing lists are required.

The following Oracle Marketing and Oracle Sales Online administration tasks are therefore included in this document to provide a single location for campaign administration and list administration information required for Oracle Scripting.

The information contained in this section is the most current information available at the time of publication. Every effort is made to synchronize between the releases of Oracle Scripting and the various applications with which it integrates. Nonetheless, if the steps listed in this document do not match the user interface you see when you attempt to perform a task, please consult the product documentation appropriate to the release installed at your site.

Functionality of Oracle applications is constantly improving and, thus, changing. Oracle Marketing and Oracle Sales Online functionality, user interfaces, requirements, and dependencies may change at any time. Therefore, campaign and list administration procedures or information contained in the latest relevant Oracle Marketing and Oracle Sales Online product documentation, if different, should take precedence over the steps provided below.

This section includes the following topics:

- Administering Campaigns in Oracle Marketing
- Administering Campaigns in Oracle Sales Online

- Administering Lists in Oracle Marketing

9.1 Administering Campaigns in Oracle Marketing

In order to launch a specified script from Oracle TeleSales or Oracle Collections, campaign administration steps are required to be performed in Oracle Marketing.

> **Note:** The administrative tasks described herein for campaign management are performed in previously existing components of the Oracle Marketing user interface, accessible from the Campaign tab.
>
> Campaign schedules can now also be created using the Campaign Workbench tab, a new feature as of Oracle Marketing release 11.5.10 that streamlines some tasks. The Campaign Workbench provides an alternate method to create campaign schedules. From this UI you can also associate scripts with a campaign schedule. You cannot associate scripts to a campaign (this can only be done from the Campaign tab).
>
> For specifics on using the Campaign Workbench, refer to *Oracle Marketing User Guide*.

Tasks

You can perform the following tasks:

- Creating a Campaign
- Creating a Campaign Schedule
- Associating a Script to a Campaign Schedule
- Associating a Script to a Campaign

See Also

- Administering Campaigns in Oracle Sales Online
- Administering Lists in Oracle Marketing

9.1.1 Creating a Campaign

Use this procedure to create a campaign in Oracle Marketing.

Prerequisites

None

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Campaign tab.

   The Campaigns page appears, displaying a summary view table of any existing campaigns.

2. Click **Create**.

   The Create Campaign page appears.

3. In the Setup Type field, perform the following:

   a. Type your search criteria for a setup type and click the search icon.

      Note: You can use the % wildcard for database searches.

      For example, to create a campaign, type **Campaigns** and click the search icon.

      Setup type is determined by the business application with which the campaign is used. Select a setup type of **Campaigns** if using Oracle TeleSales, or **Collections** if using Oracle Collections.

      The Campaign Selector page appears, listing results matching your search criteria.

   b. Optionally, refine your search. You can use the % wildcard character anywhere on this page.

   c. In the Setup Name list, click the appropriate hyperlink.

      – If using Oracle TeleSales as your business application, click **Campaigns**.

   - If using Oracle Collections as your business application, click **Collections**.

   The Campaign Selector page closes, and the Create Campaign page refreshes, listing the item you selected.

4. In the Name field, type a name for this new campaign.

   For example, type **TestCampaign**.

5. Optionally, in the Program field, perform the following:

   a. Type your search criteria for a setup type and click the search icon.

      The Program Selector page appears, listing results matching your search criteria.

   b. Optionally, refine your search. You can use the % wildcard character anywhere on this page.

   c. In the table of programs, click the appropriate hyperlink.

      The Program Selector page closes, and the Create Campaign page refreshes, listing the item you selected.

6. If creating a campaign for use with Oracle Collections, in the Source Code field, type a unique source code.

   If using this campaign for other applications, this field is optional.

7. Optionally, from the Business Unit list, select a business unit.

8. In the version field, type a version number.

   When creating a campaign with the same name as an existing campaign, only one version is active at a time. This field allows you to designate the new version as active.

9. Optionally, from the Purpose list, select a purpose for the campaign.

10. Optionally, in the Start Period field, perform the following:

   a. Type your search criteria and click the search icon.

      The Period Selector page appears, listing results matching your search criteria.

   b. Optionally, refine your search. You can use the % wildcard character anywhere on this page.

   c. In the period list, click the appropriate hyperlink.

The Period Selector page closes, and the Create Campaign page refreshes, listing the item you selected.

11. Optionally, to enter an end period, in the End Period field, repeat step 10.
12. In the Start Date field, perform the following:
    a. Click the Date Picker widget.

       The Select a Date window appears.

    b. Using the controls, navigate to the appropriate month and year if required. Then, click on the appropriate date.

       The Select a Date window closes, and the Create Campaign page refreshes, listing the item you selected.
13. To enter an end date, in the End Date field, repeat step 12.
14. From the Currency list, ensure the appropriate currency is displayed. If not, select it from the list.
15. Optionally, from the Priority list, select a priority for the campaign.
16. In the Owner field, ensure the appropriate owner name is displayed. If not, type the appropriate search criteria and select the appropriate owner from the list.

    The current username populates this field by default.
17. Optionally, if a global campaign, select Global.
18. In the Country field, ensure the appropriate country is displayed. If not, type the appropriate search criteria and select the appropriate country from the list.
19. Optionally, if a confidential campaign, select Confidential.

    A confidential Campaign cannot be viewed by people outside of the team.
20. From the Language list, ensure the appropriate language is displayed. If not, type the appropriate search criteria and select the appropriate language from the list.
21. Optionally, if you intend to use this campaign as a template on which to build other campaigns, select Template.
22. Optionally, in the description text area, type a description of this campaign.
23. Review your campaign parameters and modify as required. When satisfied, click **Create**.

The Create Campaign page refreshes, becoming the Campaign Details page. A confirmation that a new item has been created appears on the top of the page.

24. If required, make any changes at this time and click **Update** to save your work.

See Also

- Creating a Campaign Schedule
- Associating a Script to a Campaign Schedule
- Associating a Script to a Campaign

9.1.2 Creating a Campaign Schedule

Use this procedure to create a campaign schedule in Oracle Marketing.

Prerequisites

A campaign must already exist to which you want to associate a schedule.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Campaign tab.

   The Campaigns page appears, displaying a summary view table of any existing campaigns.

2. Click the campaign for which you want to create a campaign schedule.

   For example, click **TestCampaign**.

   The Campaign Details page appears.

3. From the side panel, click the **Execution** hyperlink.

   The Campaign Details – Campaign Schedule page appears.

4. Click **Create**.

   The Create Campaign Schedule page appears.

5. From the Setup Type field, select the appropriate setup type.

   - For example, for campaign schedules to be used with Oracle Advanced Outbound Telephony, select **Direct Marketing – Telemarketing**.
   - If using a custom setup type to avoid the budget approval step, select the custom setup type.

   The Activity Type and Activity fields will auto-populate.

6. In the Name field, type a name for this new campaign schedule.

   For example, type **TestCampaignSchedule**.

7. From the Language list, ensure the appropriate language is displayed. If not, select the appropriate language from the list.

8. In the Coordinator field, ensure the appropriate campaign schedule coordinator name is displayed. If not, enter the appropriate search criteria and select the appropriate coordinator from the list.

   The current username populates this field by default.

9. Optionally, in the Source Code field, type a unique source code if not already populated. To use the source code of the parent campaign, select **Use Parent Source Code**.

10. In the Activity and Activity Type lists, ensure the values specified by the campaign schedule setup type are displayed.

11. Optionally, in the Marketing Medium field, enter the appropriate search criteria and select the appropriate marketing medium from the list.

12. From the Currency list, ensure the appropriate currency is displayed. If not, select it from the list.

13. From the Time Zone list, ensure the appropriate time zone is displayed. If not, select it from the list.

    This time zone setting should match the time zone for the geographic location where the campaign schedule activity takes place.

14. In the Start Date field, perform the following:

    a. Click the Date Picker widget.

       The Select a Date window appears.

    b. Using the controls, navigate to the appropriate month and year if required. Then, click on the appropriate date.

The Select a Date window closes, and the Create Campaign Schedule page refreshes, listing the item you selected.

15. Optionally, to enter an end date, in the End Date field, repeat step 14.
16. Optionally, in the Start Time fields, select the appropriate hour, minute, and AM/PM designation from the list.
17. Optionally, in the End Time fields, select the appropriate hour, minute, and AM/PM designation from the list.
18. Optionally, from the Objective list, select the appropriate objective for this campaign schedule.

    Objectives may not appear in the list of values, based on previous selections.
19. Optionally, from the Priority list, select the appropriate priority for this campaign schedule.
20. Optionally, in the description text area, type a description of this campaign schedule.
21. Ensure that the **Do Not Use Trigger** option is selected. When this option is selected, no values appear in the Trigger Condition list.
22. Optionally, to enter approval conditions, in the Execute After Approval by field, search for and select an appropriate approving authority.
23. Optionally, in the Send Notification to field, search for and select the users whom you want to be notified of the approval.
24. If using notifications, in the Subject field, enter an e-mail message subject for the approval notification.
25. If you want to use an e-mail template for the notification, in Template field, search for and select the appropriate template.
26. Review your campaign schedule parameters and modify as required. When satisfied, click **Create**.
27. The Create Campaign Schedule page refreshes, becoming the Campaign Schedule Details page.
28. If required, make any changes at this time and click **Update** to save your work.

References

- Section 9.2.1, "Assigning Agents to a Campaign Schedule"
- Section 9.2.2, "Assigning Campaign Schedules to Agents"

See Also

- Creating a Campaign
- Associating a Script to a Campaign Schedule
- Associating a Script to a Campaign

9.1.3 Associating a Script to a Campaign Schedule

Use this procedure to associate a particular deployed Script Author script to a campaign schedule for use in a business application. Thereafter, this association is applied to all users of the campaign schedule.

Prerequisites

- A valid campaign and campaign schedule must be defined.
- At least one appropriate script must be deployed from Script Author to the applications database.
- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.
- You must know whether the script is an inbound or outbound script.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Campaign tab.

   The Campaigns page appears, listing a summary view table containing all existing campaigns created by the current user.

2. From the Name field in the table, select the appropriate campaign.

> **Note:** This campaign must contain the campaign schedule under which your target group resides.

The Campaign Details page appears.

3. From the side panel, click **Execution**.

   The Campaign Details – Campaign Schedule page appears, displaying a summary view table of all campaign schedules associated with the selected campaign.

4. From the Name field in the table, click the campaign schedule to which you want to associate a script.

   The Campaign Schedule Details page appears.

5. From the side panel, click **Collaboration**.

   The Campaign Schedule Details – Contact Points page appears, displaying the contact points table, which lists any existing Oracle Scripting script associated to the selected campaign schedule.

6. From the Type list, select **Inbound Script** or **Outbound Script**, as appropriate.

7. In the Name field, search for and select the name of the specific script you want to associate.

   The Campaign Schedule Details – Contact Point page refreshes, populated with the selected values.

   c. To save the selected values, click **Update**.

      The Campaign Schedule Details – Contact Point page refreshes, indicating that the selected script is now associated with the campaign schedule.

8. If you wish to associate any other script with this campaign schedule, repeat steps 6 and 7 above.

See Also

- Creating a Campaign
- Creating a Campaign Schedule
- Associating a Script to a Campaign

9.1.4 Associating a Script to a Campaign

Use this procedure to associate a particular deployed Script Author script to a campaign for use in a business application. Thereafter, this association is applied to all users of the campaign.

Prerequisites

- A valid campaign must be defined.
- At least one appropriate script must be deployed from Script Author to the applications database.
- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.
- You must know whether the script is an inbound or outbound script.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Campaign tab.

   The Campaigns page appears, listing a summary view table containing all existing campaigns created by the current user.

2. From the Name field in the table, select the appropriate campaign.

   The Campaign Details page appears.

3. From the side panel, click **Collaboration**.

   The Campaign Details – Contact Point page appears, displaying a table listing any existing Oracle Scripting scripts associated to the selected campaign.

4. From the Type list, select **Inbound Script** or **Outbound Script**, as appropriate.
5. In the Name field, search for and select the name of the specific script you want to associate.

   The Campaign Schedule Details – Contact Point page refreshes, populated with the selected values.

6. To save the selected values, click **Update**.

   The Campaign Details – Contact Point page refreshes, indicating that the selected script is now associated with the designated campaign.

7. If you wish to associate any other script with this campaign, repeat steps 5 and 6 above.

See Also

- Creating a Campaign
- Creating a Campaign Schedule
- Associating a Script to a Campaign Schedule

9.2 Administering Campaigns in Oracle Sales Online

In order to have agents associated with campaign schedules or vice versa, campaign administration steps are required to be performed in Oracle Sales Online.

Tasks

You can perform the following tasks:

- Assigning Agents to a Campaign Schedule
- Assigning Campaign Schedules to Agents

See Also

- Administering Campaigns in Oracle Marketing
- Administering Lists in Oracle Marketing

9.2.1 Assigning Agents to a Campaign Schedule

Use this procedure to assign one or more agents to a campaign schedule in Oracle Sales Online.

Prerequisites

None

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Sales Online Super User

Steps

1. From the Oracle Sales Online administration console, click the Administration tab.

   > **Note:** In order to access this tab, your Oracle Applications account must be a member of a valid sales group. If the required group does not yet exist, it should be created prior to importing the administrator as a CRM resource. Groups and group membership are administered in CRM Resource Manager.

2. Click the Sales subtab.

3. From the side panel, under Call Center, click **Agent Assignments.**

   The Resource Assignment page appears.

4. In the Schedule Name area, click **Go**.

   The Valid Campaign Schedules page appears.

5. If the campaign schedule you want to assign does not appear, then in the Campaign Schedule Name field, search for and select the appropriate campaign schedule.

   For example, type **TestCampaignSchedule** and click **Search**.

   The Valid Campaign Schedules page refreshes, showing a list of valid campaign schedules that meet your search criteria.

6. In the Valid Campaign Schedules page, in the Select column, select one or more appropriate valid campaign schedules and click **Ok**.

   The Resource Assignment page refreshes, listing the valid campaign schedules you selected.

7. Under the Assign To heading, select **Resource**.

8. From the Current Resource Assignments area, click **Add Resource**.

   The Resource Assignment page refreshes. The Current Resource Assignments table includes one or more rows, with text fields under the Resources column.

9. In the Resource field, perform the following:

   a. Type search criteria for an agent (by agent’s last name) and click **Go**.

      The Resources Selector page appears, displaying results of your search.

b. Optionally, refine your search.

   You can use the % wildcard character anywhere on this page.

c. In the resources list, click the appropriate hyperlink.

   The Resources Selector page closes, and the Current Resource Assignments list on the Resource Assignment page refreshes, listing the specified agent name you selected.

10. Repeat the previous step as required for all agents. When satisfied, click **Update**.

11. If necessary, repeat steps 5 through 10 for each campaign schedule you want to assign.

See Also

- Assigning Campaign Schedules to Agents

9.2.2 Assigning Campaign Schedules to Agents

Use this procedure to assign one or more campaign schedules to agents in Oracle Sales Online.

Prerequisites

None

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Sales Online Super User

Steps

1. From the Oracle Sales Online administration console, click the Administration tab.

> **Note:** In order to access this tab, your Oracle Applications account must be a member of a valid sales group. If the required group does not yet exist, it should be created prior to importing the administrator as a CRM resource. Groups and group membership are administered in CRM Resource Manager.

2. Click the Sales subtab.
3. From the side navigation bar, under Call Center, click **Campaign Assignments.**

   The Campaign Assignment page appears.
4. In the Assign To area, select **Resource** and click **Go**.

   The Resources page appears.
5. If the resource you want to assign is not in the Resources list, then in the Resource Name field, perform the following:
   1. Type search criteria for an agent (by agent’s last name) and click **Search**.

      The Resources page refreshes, displaying results of your search.
   2. Optionally, refine your search.

      You can use the % wildcard character anywhere on this page.
   3. In the Select column, select one or more appropriate valid resources and click **Ok**.

      The Campaign Assignment page refreshes, listing the valid resources you selected.
6. Click **Apply**.

   The Campaign Assignment page refreshes.
7. Under the Current Campaign Assignments heading, click **Add Campaign**.

   The Campaign Assignment page refreshes. The Current Campaign Assignments table includes one or more rows, with text fields under the Name column.
8. In the **Name** field, perform the following:
   1. Type search criteria for a campaign schedule and click **Go**.

The Valid Campaign Schedules selector page appears, displaying results of your search.

b. Optionally, refine your search.

   You can use the % wildcard character anywhere on this page.

c. In the campaign schedules list, click the appropriate hyperlink.

   The Valid Campaign Schedules selector page closes, and the Current Campaign Assignments list on the Campaign Assignment page refreshes, listing the schedules you selected.

9. Repeat the previous step as required for all campaign schedules. When satisfied, click **Update**.

   Successful assignment of a campaign schedule to an agent is indicated by a confirmation message at the top of the page.

See Also

- Assigning Agents to a Campaign Schedule

9.3 Administering Lists in Oracle Marketing

Note: The administrative tasks described herein for list management are performed in previously existing components of the Oracle Marketing user interface, accessible from the Audience tab.

Oracle Marketing release 11.5.10 and later includes an Audience Workbench tab, from which list administration can be performed. For specifics on using the Audience Workbench, refer to *Oracle Marketing User Guide*.

Using the Survey component of Oracle Scripting, lists created with the list management features of Oracle Marketing are used as the contact sources for executing the survey campaign using e-mail as the media channel. The list must already exist at the time you define a deployment for a survey campaign. The list can be created from the Audience tab of the Survey Administration Console, or the Audience tab of the Marketing Administration Console. These lists are passed to Oracle One-to-One Fulfillment. Oracle One-to-One Fulfillment uses a master document (an electronic mail message in HTML format), and an associated query, to merge variables into the message at time of execution. The combination of e-mail master document and the query are referred to in Oracle One-to-One Fulfillment as a template. Upon execution of a concurrent request, the Fulfillment engine executes the query. For each list member, the data retrieved from the query is merged into a copy of the master document, personalizing the HTML invitation or reminder with unique data.

At minimum, an invitation or reminder master document template includes as merge field variables each list member’s e-mail address, and the survey URL. For list-based surveys, the URL contains a parameter to uniquely identify each respondent. This respondent identification code (rID) ties each response to its originating list member.

Typically, some other data from the Oracle Marketing list (for example, title of address, first name, last name, organization, organization address information, country, and so forth) are used in the master document as a merge field variable to personalize the e-mail message. From a survey perspective, the only merge field variables required are respondent e-mail address and survey URL. The minimum set of merge fields that are required in the Oracle Marketing list may be larger, and require certain fields, based on the method of populating the list with records and the specified list type.

The merged HTML documents are then sent to the electronic mail server designated by Oracle One-to-One Fulfillment. When the message is sent out by the mail server, all list members with valid e-mail addresses receive the message as an invitation (or reminder) to participate in a Web-based survey. Clicking on a hyperlink or otherwise accessing the unique survey URL contained in the invitation results in a Scripting Engine interaction in the Web interface as the respondent participates in the survey.

This model is only possible with one or more valid lists, which must be created and defined using Oracle Marketing functionality.

This section includes the following topics:

- List Creation Overview
- Creating a List
- Identifying Record Sources for a List
- Adding Records to a List
- Generating a List
- Marketing Campaigns as an Oracle Applications Dependency

See Also

- Administering Campaigns in Oracle Marketing
- Administering Campaigns in Oracle Sales Online

9.3.1 List Creation Overview

There are various uses for lists, and correspondingly various ways to identify records to load into an Oracle Marketing list. The list creation itself is a simple process resulting in a list object, at first an empty container awaiting records. Thereafter, you must identify a record source, add the records to the list, and generate the list.

This document assumes the Marketing Administration Console is used for these steps. However, the list management features of Oracle Marketing are also accessible from the Survey Administration Console under the Audience tab, which corresponds to the Audience tab in the Marketing Administration Console. Each appropriate subtab or side panel link required for Oracle Marketing administration can be accessed from either administrative console. If administering lists from the Survey Administration Console, the survey administrator must also be assigned the appropriate marketing responsibility.

See Also

- Creating a List
- Identifying Record Sources for a List
- Adding Records to a List
- Generating a List
- Marketing Campaigns as an Oracle Applications Dependency

9.3.2 Creating a List

Use this procedure to define a list in Oracle Marketing. Until record sources are subsequently defined for the list, it will remain in Draft status.

Prerequisites

You must know the list type and source type before creating a list. For more information, see Oracle Marketing product documentation.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the **Audience** tab.

   The Organization page appears.

2. Click the **List** subtab.

   The Lists page appears.

3. Click **Create**.

   The Create List page appears.

4. In the Name field, type a name for your list.
5. From Type list, select **Standard** or **Manual**.

   This field cannot be modified after the list is saved.

6. From the Source Type list, select the source type you want to use with your list.

   This field cannot be modified after the list is saved.

   Two typical source types for a list are **Persons** and **Organization Contacts**. These map to fields in the TCA architecture. Other source types can be added in Oracle Marketing if necessary (for example, from HR, the Employees source type). For more information, see Guidelines below.

7. If the appropriate person is not populated in the Owner field, do the following:

   a. In the Owner field, type your search criteria and click the search icon.

      You can use the % wildcard search character if it is the second or subsequent character in your search string. For exact title matches, enter search criteria using the syntax <Lastname>, <Title> <Firstname>. For example, to obtain the record Mr. George Washington, type **Washington, Mr. George** and click the search icon.

   b. In the Resource Selector page, refine your search criteria if necessary.

c. From the resulting list on the Resource Selector page, click the name of the appropriate owner.

   The Resource Selector page closes. The Create List page refreshes, containing the selected value.

8. Optionally, in the Description field, type a description or purpose for this list.
9. Click **Create**.

   The Create List page refreshes, becoming the List - Main page.

   If successful, at the bottom of the page, confirmation messages appear, indicating that the record has been created and updated.

Guidelines

- List types are not enforced by all applications. Manual lists can have each list entry manually designated from existing records. For all other lists, including lists that will be populated by Oracle Discoverer workbook, use Standard. Suppression lists contain list entries that other lists can use to filter out or suppress, effectively creating a stop-list mechanism that can be used for various other lists. Suppression lists are generally not used as lists of contacts to execute a campaign.
- A list source type is a concept used by Oracle Marketing to define how fields from the database map to the columns of a list entry, and which fields will be populated. A list entry, stored in the table AMS_LIST_ENTRIES, has many data fields. Two typical source types are Persons and Organization Contacts. These list source types define whether a set of list entries are contacts in an organization (in which case certain fields related to an organization are required, such as Organization, Organization Contact, and Country) or are persons (no organization information is necessary, but FIRSTNAME and LASTNAME are required). For the Organization Contacts source type, the PARTY_ID field contains the name of the organization, and the P_CONTACT_PARTY_ID field contains the name of the individual. However, for the Persons source type, the PARTY_ID field contains the name of the individual, and the P_CONTACT_PARTY_ID field is null.

See Also

- List Creation Overview
- Identifying Record Sources for a List
- Adding Records to a List

- Generating a List
- Marketing Campaigns as an Oracle Applications Dependency

9.3.3 Identifying Record Sources for a List

You must identify record sources to populate lists with list entries. Oracle Marketing lists can be populated in one of three ways:

- Importing a list from a flat file of comma-separated values.
- Manually building a list from existing records in other Oracle Marketing lists or in the TCA architecture.
- Using Oracle Discoverer Workbook from Oracle Marketing to identify list details and conditions to serve as a query to generate a list from the TCA architecture.

Tasks

You can perform the following tasks:

- Importing Record Sources
- Manually Selecting List Entries as Record Sources
- Using Oracle Discoverer Workbook to Define Record Sources

See Also

- List Creation Overview
- Creating a List
- Adding Records to a List
- Generating a List
- Marketing Campaigns as an Oracle Applications Dependency

9.3.3.1 Importing Record Sources

Use this procedure to import record sources into an Oracle Marketing list. Data types that can be imported include:

- Event registrations
- Leads
- Organizations, Contacts, Addresses

- Persons, Addresses

Prerequisites

- An Oracle Marketing list must already exist.
- You must have a flat file available on your computer's local file system or network, or available using the file transfer protocol.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the **Audience** tab.
2. Click the **Import** subtab.

   The Imports page appears.
3. Click **Create**.

   The Import: Introduction page appears.
4. From the Import Wizard list, select the type of record source you want to import, and click **Next**.

   See Guidelines below for information on record source types.

   The Import Step 1a: Definition page of the import wizard appears.
5. In the **Name** field of the import wizard, type a name for the record source you want to import.
6. Optionally, in the description field, type a description of this record source.
7. In the Source File area, designate the location of the record source you want to import, and provide details on how the record source is organized.

   a. Optionally, from the Character Set list, select a character set if different from the default selection.

b. To locate a file on your computer’s file system, select the **Client** option. To locate a file elsewhere on the network, click **Server**. To use the file transfer protocol to obtain the list, click **FTP**.

c. Click **Go**.

   The Import List window appears.

d. From the Import List window, click **Browse...**.

   The Choose File window appears.

e. Navigate to the appropriate comma-separated or ZIP file, select it, and click **Open**.

   The Choose File window closes. The Import List window refreshes, with the path and file name of your selection populating the appropriate text field.

f. From the Import List window, click **Submit**.

   The Import List window closes. The Definition page refreshes, with your selection populating the appropriate text field.

g. From the Column Delimiter list, ensure the appropriate delimiter type is selected.

   For example, if using comma-separated values, ensure **Comma** is selected.

h. From the Field Enclosed By list, ensure the appropriate field symbol used in the import file is selected.

   Fields in the import list can be separated by commas, double quotation marks, semicolons, or single quotation marks. Select the choice that reflects the symbol used to enclose each field within each record in the source list you want to import.

i. If the list record source contains file headers such as column definitions, check the File Header Exists box. Otherwise, clear this box.

   If you select this option, the first row of the import file will display as field names in the Source Fields area when mapping fields. Otherwise, the fields in the Source Fields area appear as Field1, Field2, Field3, etc.

j. Optionally, if merging lists or otherwise removing duplication, perform the following:

   – In the Remove Duplications Using Rule field, enter search criteria for rules to remove duplicate records in the list, and click the search icon.

The DeDuplication Rule LOV window appears.

- In the DeDuplication Rule LOV window, refine your search criteria if necessary.
- From the resulting list, click the name of the appropriate rule.

  The Definition page refreshes, containing the selected value.

k. Optionally, select the **Use Word Standardization** option.

8. Click **Next**.

   For record sources of the lead or event registration import types, the Import Mapping page of the import wizard appears. Proceed to step 11.

   For record sources of the Organizations, Contacts, Addresses import type or the Persons, Addresses import type, the Import Step 1b: Additional Details page of the import wizard appears.

9. Optionally, in the provided fields, enter additional information about the record source you want to import.

   If the description import type is a rented list, you must supply either an expiration date, a number of uses, or both.

10. Click **Next**.

    The Import Mapping page of the import wizard appears. From here you can import existing mappings, designate new mappings, and view records in the record source you want to import in the Preview field.

11. To map fields using an existing mapping, in the Load Existing Mapping field, search for and select the appropriate existing mapping. Verify that the required fields (fields preceded with an asterisk in the Target Fields column) are all included in the mapping, and make changes if required.

    Optionally, you can save or rename a mapping.

12. To create a new mapping, establish a match between the source fields from your record source and the target fields derived from the Oracle TCA architecture. Match each set of fields in the order used in your record source. This order (as well as the first few records in your record source) is displayed in the preview field below.

    For example, if the first entry in the Preview Count window is **George**, click Field1 or PERSON_FIRST_NAME in the Source Fields list and Person First Name Target Fields list.

13. Click the right arrow to confirm the first field mapping.

    In the Mapping: Source to Target column, the correlation you designated between the field in your record source and the field in the TCA architecture appears.

14. Repeat steps 12 and 13 above for each source field you want to include in your list, until complete.

    - Each field designated in the final target mapping must have a corresponding record in the Target Fields list.
    - The converse is not true. You may exclude from the final mapping some fields in the source fields list, as long as they are not required fields as designated by an asterisk in the Target Fields list. Data from excluded fields will not become part of the list. If you exclude some source fields in your final target mapping, your list may not pass validation used by applications such as Oracle Advanced Outbound Telephony.

15. Click **Next**.

    If successful, a confirmation appears on the Import Step 3: Review page of the import wizard. If required fields are missing, click **Cancel** and add them to the record source for each record for subsequent importation. If fields are mismapped, click **Restore** to clear the mapping page and remap until successful.

16. For two import record source types (both Organizations, Contacts, Addresses and Persons, Addresses), you can automatically generate the list during the import process. To do this, click the **Generate a List upon Import** box.

    Note: If you select this option, you do not need to perform the task described in Section 9.3.5, "Generating a List".

17. Click **Import**.

    The Confirmation page appears, displaying a confirmation note stating that the list has been submitted for processing.

18. If you want to import another record source, click **New Import**. Otherwise, click **Finish**.

Guidelines

- When you select an import type, you determine which fields, at minimum, are required to be included in the list. Note that fields in addition to those marked by an asterisk may be required, due to varying business rules related to the purpose for the list.

  Based on import type, the following fields are required:

Import Type	Intended Use	Required Fields
Event Registrations	Event Registrations	■ Event Source Code
Leads	Leads	■ Address1 ■ Country ■ Customer Name ■ First Name ■ Last Name ■ Promotion Code ■ Source System
Organizations, Contacts, Addresses	Organizations, Contacts, Addresses	■ Organization Name ■ Party ID
Persons, Addresses	Lists of persons only	■ Person First Name ■ Person Last Name
Persons, Addresses	Lists of persons and their addresses	■ Person First Name ■ Person Last Name ■ Country
Any import type to be used with the survey component of Oracle Scripting	Targeted survey campaigns	■ Survey URL

- If creating lists for use with the survey component of Oracle Scripting, you must include the survey URL as a field.
- When specifying additional details, if the description import type is a rented list, you must supply either an expiration date, a number of uses, or both.
- If the source set in the Additional Details page is Purchased List, the records from your import file will be added to the Customer Master list. This may also be true for the Others source type, based on your configuration.

See Also

- Manually Selecting List Entries as Record Sources
- Using Oracle Discoverer Workbook to Define Record Sources

9.3.3.2 Manually Selecting List Entries as Record Sources

Using the Add List Entries - Contact Search page, you can search for existing records in the TCA architecture or from an existing Oracle Marketing list, and add them as a list entry on a specified list. You can search by person information, address, organization information, or you can find records for a specified organization based on the individual's relationship to the organization.

Use this procedure to manually select list entries from existing records in the TCA architecture or from an existing Oracle Marketing list.

Prerequisites

- An Oracle Marketing list must already exist.
- The list entries you want to create must already exist.
- You must have information from which to query records to add list entries to the Oracle Marketing list.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.
2. Click the List subtab.

   The Lists page appears.
3. From the Name column of the lists list, click the appropriate hyperlink to select the existing Oracle Marketing list to which you want to add entries.

   For example, click **TestList.**

   The Lists - Main page appears.

4. From the side panel, click **Entries**.

   The List Entries page appears.

5. Click **Add to List**.

   The Add List Entries - Contact Search page appears.

6. Enter your search criteria and click **Search**.

   The Add List Entries - Search Results page appears, with a list of all records that match your search criteria.

7. Refine your search criteria if necessary.

8. In the Select column of the Add List Entries - Search Results page, mark each appropriate record that you want to include as a list entry in the selected list.

   Note: All list entries must have valid e-mail addresses in order to receive invitations or reminders to participate in a survey.

   Note: For testing purposes, it is recommended to add yourself as a list entry. Ensure a valid e-mail address is associated with your list entry record.

9. Click **Add to List**.

   The Confirmation List Entries page appears. The designated contacts are added to the specified list.

10. Make any changes if required, and click **Update.**

    Once the list is generated, it can be used in a survey campaign deployment.

11. Click **Return to List Entries** to verify that the selected entries were added to the appropriate list.

12. Make any changes if required, and click **Update.**

    For example, add an e-mail address to any missing list entries, and click **Update**.

See Also

- Importing Record Sources

9.3.3.3 Using Oracle Discoverer Workbook to Define Record Sources

Oracle Discoverer is interface software available from the Marketing Administration Console that allows users to view and select records from complex database structures without requiring knowledge of those structures or knowledge of Structured Query Language (SQL). Discoverer generates workbooks, which are lists of records similar in design to a spreadsheet. Workbooks may then be used to generate Oracle Marketing lists or to provide a SQL statement for a segment.

Use this procedure to create an Oracle Discoverer workbook from Oracle Marketing. Using the folders provided in the Oracle Discoverer layer of Oracle Marketing to define record sources for a list provides access to many data fields in the TCA. The workbook parameters are executed when the list is generated.

Prerequisites

- Oracle Discoverer must be appropriately implemented and configured for use in Oracle Marketing.
- You must have the appropriate responsibility and requisite group roles and membership to access the Discoverer component. For more information, see Oracle Marketing product documentation.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.
2. Click the Discoverer subtab.

   Oracle Discoverer application launches in a separate browser window. The Open Workbook window appears.
3. Click **Create a new workbook**.

   The Open Workbook window refreshes with added display information options.
4. Ensure the Table display option is selected.

5. Click **Next**.

   The Select Items window appears.

6. From the Available list, select the appropriate list management option.

   For example, select **Marketing Online Organization List Management**.

   The Select Items window refreshes, showing folders appropriate to the option you selected. You can drill down into each folder by clicking the plus sign.

7. From the folder area of the Available list, select the appropriate folder.

   For example, for a list based on organizations, select **Organization Contact List Details**, or for a person or address list, select **Marketing Online Person List Management**.

8. Click the plus sign preceding the folder to display the folder's contents.

   The Select Items window refreshes, showing fields in the database associated with your selection.

9. Select **Mandatory Identifier** and click the select arrow between the Available list and the Select list to move that field to the Select list.

10. Add to the select list all other mandatory fields.

    Include in the select list all fields preceded with the word "Include." Other fields may be mandatory, based on the purpose for the list.

    For example, for an organization list, you must also select the **Party Site Identifier, Email Address, Person First Name, Person Last Name**, and **Person Title** fields.

11. Review your field selections. When satisfied, click **Next**.

12. Click **Next**.

    The Table Layout window appears. No data is required for this window.

13. Click **Next**.

    The Format window appears. No data is required for this window.

14. Click **Next**.

    The Conditions window appears.

15. Optionally, enter any conditions by doing the following:

    a. From the Conditions window, click **New**.

The New Conditions window appears.

b. Optionally, in the name field, type a name for the new condition.

c. Optionally, in the description field, enter a description for the new condition.

d. In the Formula area, start defining the condition. From the Item list, select the appropriate item for which you want to create a condition.

   Each field in the Item list is preceded by its parent list management option and a period. For example, the Mandatory Identifier is named Organization Contact Details List.Mandatory Identifier.

e. In the Formula area, from the Condition list, select the condition operator.

   For example, select the equal sign.

f. In the Formula area, in the Values list, type a new value, or select **Create Calculation...** to create a calculation, **Select Item...** to select an item, or **New Parameter...** to define a new condition parameter.

   For example, type an organization ID.

g. If your condition is case-sensitive, ensure the Match case option is selected. If not, click to clear this option.

h. Check your condition parameters. When satisfied, click **OK**.

   The New Conditions window closes.

   To add additional conditions, repeat a through h of this step.

16. Click **Next**.

    The Sort window appears.

    Note: Do not include any sort conditions in your workbook. Generation of a list of list type Standard remains in Draft status if (a) a Sort condition is included in the workbook, or (b) if all the required fields (Party Id, Mandatory Identifier, One of the Include fields, Title, e-mail Address, First name, Last Name) are not included in the workbook.

17. Click **Next**.

    The Calculations window appears.

18. Optionally, add a calculation. When finished, click **Next**.

    The Percentages window appears.

19. Optionally, add a percentage. When finished, click **Next**.

    The Totals window appears.

20. Optionally, add a total. When finished, click **Next**.

    The Parameters window appears.

21. Optionally, add a parameter.

22. Click **Finish**.

    The Parameters window closes. After processing, your new workbook displays.

23. From the File menu, select **Save As...**, type a name for your workbook, and click **Save**.

24. To exit Oracle Discoverer, from the File menu, select **Exit**.

See Also

- Importing Record Sources
- Manually Selecting List Entries as Record Sources

9.3.4 Adding Records to a List

After creating a list and identifying record sources, records must be added to the list. Use this procedure to add records to a list in Oracle Marketing.

> **Note:** If you selected the **Generate a List upon Import** box when importing records of source type Organizations, Contacts, Addresses or Persons, Addresses, this procedure is not required.

Prerequisites

- An Oracle Marketing list must already exist.
- Record sources must already be identified.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.
2. Click the List subtab.

   The Lists page appears, displaying a summary view table of all existing lists.
3. From the table, click the name for the list into which you want to add records.

   The List – Main page appears.
4. From the side panel, click **Selections**.

   The List Selections page appears.
5. In the first row:

   a. Type an integer designating the order in which you want the first record source to be incorporated into the list. The lowest numbered record source will be used first. For more information, see Guidelines below.

   b. From the Operation list, select an operation type. Operations available are Include, Exclude and Intersect. For more information, see Guidelines below.

   c. From the Type list, select a record source type. Valid record source types include Import List, List, SQL, Segment, or WorkBook. For more information, see Guidelines below.

   d. In the Name field, search for and select the name of the record source you want to use to provide record entries to this list.

   e. If an Include operation, in the Dedupe Rank field, type a deduplication rank for the record source.

      A deduplication rank determines which record to retain when a user removes duplicate records after the list has been generated. Deduplication occurs only during an Include operation, in which cases it is a required parameter.

   f. Optionally, if an Include operation, in the Percent of Records field, type an integer designating the percentage of records in the record source you want to include in the list.

      For example, if the named record source contains ten thousand records and you only wish to import one thousand, type 10 in this field.

6. Repeat step 5 for each additional record source you want to include, exclude, or intersect to compose this list.

7. Review your list selection criteria. When satisfied, click **Update**.

   The List – Selection page refreshes, displaying a confirmation message.

Guidelines

- When designating an order for list selection, each record source in the list must contain a unique number. The lowest numbered record source will be used first.
- The first record source must be of type "include" so that records will be included in the list. Designate all record sources to include prior to designating record sources to exclude.
- The Source Types (Organization Contacts or Persons) should be the same for all the items (lists, workbooks, segments) used in the operation (Include, Exclude, or Intersect) and the main list that you've created.
- If one of the list sources is a suppression list or another list of type "exclude," designate this last in the list selection page, to ensure that records to be included will exist in the set from which exclusions are to occur.
- When designating the operation type, use the following criteria:

  Use **Include** to add all entries in the record source to the list. Include is a required operation for the first record source to be used.

  Use **Exclude** to remove from the final list any entries in the specified record source. Exclude is generally used to remove a set of entries from a larger record source that is included in the list.

  Use **Intersect** to designate a second or subsequent primary list criteria. The list sources to be included must be listed prior to the record source designated as an Intersect record source. For example, to intersect a record source of people making over $100,000 with a record source of people living in Montana, the resulting list will contain entries for people living in Montana who make more than $100,000. Thus, incorporation order is crucial, as other criteria must precede the Intersect designation.
- When designating record source types, use the following criteria:

  If adding records from an existing import list, select **Import List**.

  If adding records using an existing SQL statement, select **SQL**.

  If adding records using an existing marketing segment, select **Segment.**

If adding records from an existing Discoverer Workbook, select **WorkBook**. Select the Type.

See Also

- List Creation Overview
- Creating a List
- Identifying Record Sources for a List
- Generating a List
- Marketing Campaigns as an Oracle Applications Dependency

9.3.5 Generating a List

After creating a list, identifying, and adding record sources to the list, the list must be generated to populate the list with all required records. Use this procedure to generate a list prior to executing a survey using the list.

> **Note:** If you selected the **Generate a List upon Import** box when importing records of source type Organizations, Contacts, Addresses or Persons, Addresses, this procedure is not required.

Prerequisites

- An Oracle Marketing list must already exist.
- Record sources must already be identified.
- Record sources must already be added to the list.
- The AMS Marketing Approvals Process concurrent program needs to be running in order to generate Standard Lists (from a workbook, from SQL, or from import lists) and to change their status to Available. Otherwise, the list status will be trapped in Generating status. For Manual Lists, there is no requirement to run this process. For more information, refer to Oracle Marketing product documentation.

Login

Log into Oracle applications using the CRM Home Page login, or the Single Sign-On login if implemented.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.
2. Click the List subtab.

   The Lists page appears, displaying a summary view table of all existing lists.
3. From the table, click the name for the list which you want to generate.

   The List - Main page appears.
4. From the side panel, click Generation.

   The List – Generation page appears.
5. If you want to generate the list now, click **Generate immediately**.
6. If you want to schedule a time and date to generate the list in the future:
   a. Click **Generate at a future time**.

   b. Using the date control widget, specify the target date on which to generate the list

   c. Using the hour, minute, and AM/PM lists, specify the time to generate the list on the specified date.

   d. From the time zone list, specify the appropriate time zone.

   e. From the Generation Type, confirm that the selected type is appropriate. If not, select the appropriate generation type from the list.
7. Click **Update**.

   The List – Generation page refreshes to display a confirmation message.

See Also

- List Creation Overview
- Creating a List
- Identifying Record Sources for a List
- Adding Records to a List
- Marketing Campaigns as an Oracle Applications Dependency

9.3.6 Marketing Campaigns as an Oracle Applications Dependency

A campaign can be generally described as a focused effort to achieve a particular goal from a targeted population over a specific period of time for a particular business purpose.

From an Oracle Applications perspective, *marketing campaigns* are the vehicle for putting information about an organization's products, services, offers, and messages in front of customers and potential customers. Campaigns are created and administered in Oracle Marketing and used by various Oracle business applications. Although sometimes referred to as a marketing campaign, Oracle Marketing campaigns can also be used for sales, telesales, service and teleservice, collections... for any business purpose.

From an Oracle Scripting perspective, there is another type of campaign that meets the general definition: a survey campaign. This is the collection of business requirements required to execute a script as a survey in a Web browser using the Scripting Engine Web interface. Survey campaigns are created using the Survey Administration console. These will always be referred to specifically as survey campaigns.

Campaigns are a fundamental entity for Oracle Marketing, and campaign attributes are prerequisites or dependencies for various Oracle business applications. For example:

- In order to specify a specific script to launch when clicking the View Script button in Oracle TeleSales, a campaign must be defined, campaign schedules associated with the campaign must be defined (including a script associated with the schedule), and agents must be assigned to the schedule or vice versa.
- The same set of campaign administration steps is required:
    - To specify a specific script to launch automatically in Oracle TeleSales upon call delivery to the agent
    - To specify a specific script to launch automatically upon start of interaction in Oracle TeleSales or Oracle Collections
    - To launch a specific script from Oracle Collections by designating a code for promotion, campaign, campaign schedule, or event (assuming a code for the specific object is also defined in Oracle Marketing).
- Campaigns are also required in order to associate Oracle Marketing lists. These lists are sets of customer contact information derived from records in the Oracle Applications Trading Community Architecture (TCA).

- Oracle Advanced Outbound Telephony requires use of the campaign schedules created using Oracle Marketing. Each of these campaign schedules can be targeted for different execution channels. For example, a marketing campaign can have a campaign schedule associated with it that is targeted for e-mail execution, and another campaign schedule associated with the campaign that is targeted for telephony execution.

See Also

- List Creation Overview
- Creating a List
- Identifying Record Sources for a List
- Adding Records to a List
- Generating a List

10 Implementation Verification

Implementation verification tasks may differ, based on which components you implement and use at the enterprise.

At minimum, every implementation requires you to create or obtain a functional script and deploy it to your environment using Script Author. Oracle Corporation recommends creating or obtaining a simple test script without custom Java and a script that includes references to custom Java.

Only scripts to be executed in the Web interface (whether executed as surveys, or as scripts executed in a Web browser from an integrated self-service Web application) require you to perform survey resource and survey campaign administration steps.

All scripts must be executed in the Scripting Engine to verify your implementation. Because there are fewer environmental factors affecting success, Oracle Corporation recommends first verifying test scripts in the Scripting Engine agent interface, even if your implementation only requires you to execute scripts in the Scripting Engine Web interface. If your enterprise requires scripts to be executed in a Web browser, then after testing in the agent interface you should test your scripts in the Web interface.

Other verification tasks specific to your Oracle Applications operating system or environment may apply.

This section includes the following topics:

- Script Author Tasks
- Survey Administration Console Tasks
- Scripting Engine Tasks
- Testing Display Server for UNIX Environments
- Viewing Server Message Logs

10.1 Script Author Tasks

All implementations must be verified using one or more scripts. Oracle Corporation recommends first testing your implementation with the Scripting Engine agent interface, using a simple script with no associated custom Java. Step-by-step instructions to create and deploy a Hello World graphical test script using Script Author's standard visual tools is provided in this section for this purpose. If using the Script Wizard feature of Script Author, follow the instructions provided to create the Hello World wizard test script.

Testing a simple script with no associated custom Java by executing it in the Scripting Engine agent interface verifies that:

- A script can be launched in the designated environment, independent of the technical considerations related to custom Java.
- You eliminate certain environmental variables as the cause of problems if this test succeeds and the subsequent test (launching and executing a script that calls custom Java methods or other custom commands) fails.
- Script Author is compiling and deploying scripts successfully.
- Scripting User or Scripting Agent responsibility functions appropriately.

If your implementation requires custom Java, you must test a script that references custom Java. Oracle Corporation recommends using the regression test script posted on Oracle*MetaLink* and Oracle iSupport for that purpose. If you can successfully execute the regression test script, you quickly verify that:

- A script with custom Java methods associated with it can be launched in the designated environment.
- The Apache Web server can locate and interpret the custom Java classes.
- Your environment is correctly set up to deploy custom Java to the database, map custom Java to scripts, and make the appropriate methods available at runtime.
- Pending the appropriate configuration of environmental factors, a customized script will function properly in your environment.

Tasks

You can perform the following tasks:

- Creating a Hello World Graphical Test Script
- Creating a Hello World Wizard Test Script

- Using the Regression Test Script

See Also

- Survey Administration Console Tasks
- Scripting Engine Tasks
- Testing Display Server for UNIX Environments
- Viewing Server Message Logs

10.1.1 Creating a Hello World Graphical Test Script

Prerequisites

- To create or deploy scripts using the Script Author Java applet, you must use the Scripting Administration console. Thus, you must have access to an Oracle Applications account with the Scripting Administrator responsibility.
- All implementation tasks must be accomplished.

Steps

Follow the steps below.

Step	Perform From	Action	Result
1	Oracle Applications HTML Login	Log into Oracle applications using the CRM Homepage login (or the Single Sign-On login, if implemented) as a user with the Scripting Administrator responsibility.	The Scripting Administration console appears.
2	Scripting Administration console	From the Home tab, click **Launch Script Author**.	A new browser window opens, Oracle JInitiator launches, and the Script Author Java applet appears in a separate window.
3	File menu *or* Script Author Toolbar	Create a new script using one of the following methods: ■ From the Script Author **File** menu, select **New.** ■ From the Script Author toolbar, click the **New Script** icon.	The new script window appears, with options for starting a new graphical script or a new wizard script.

Step	Perform From	Action	Result
4	New Script window	From the New Script window, click Graphical script.	A new, untitled script appears within Script Author's visual layout region (the "canvas"), including a start node.
5	Script Author Toolbar	Optionally, adjust the view of the script from the Script Author toolbar. For example: ▪ To display a 100% view of the canvas, click the **Zoom to 100%** icon. ▪ To zoom in or out as required to display the entire script in the current window, click the **Zoom To Fit Graph In Window** icon. ▪ For a closer or further view of the canvas, click the **Zoom In** or **Zoom Out** icon, respectively.	The view of the script on the canvas adjusts per your request.
6	Script Author Toolbar	Click the **Panel Insertion Mode** icon (the blue sphere) from the toolbar to indicate you wish to insert a new panel.	Script Author Toolbar
7	Script Author canvas	Click *once* in the canvas below the start node. This action places a panel icon (shown as a blue sphere) into the canvas. **Note:** Until you switch to another toolbar object or switch to toggle selection mode (performed in the next step), the current toolbar object remains selected and will continue to place objects (in this case, panels) on the canvas each time the mouse is clicked in the work area. If you prefer to disable this feature, click to clear the sticky mode option under the **File** menu. The option to disable sticky mode is retained only for the duration of the current Script Author session. If you prefer this option, you must select it each time you launch Script Author.	Script Author canvas

Step	Perform From	Action	Result
8	Script Author Toolbar	Click the **Toggle Selection Mode** icon from the toolbar to turn off the panel insertion mode. **Note:** If you clear the sticky mode option as described in the step above, selecting Toggle Selection Mode will not be required. However, when using the same toolbar tool repeatedly (when dropping multiple objects, or drawing multiple branches of the same type), you will be required to select the desired object from the toolbar each time, which is not required when sticky mode is enabled.	Script Author Toolbar
9	Script Author canvas	**Right-click** the panel that you just added to the script. The Script Author displays a drop down menu with the following options: Edit, Edit Blob Name, and Edit Blob Properties. **Note:** If the menu displayed shows only two options, Edit and Edit Blob properties, then you have not correctly selected the Panel object. Left-click to close the menu and re-select the Panel object.	Script Author canvas
10	Panel Properties window	Select the **Edit Blob Properties** option from the menu created by right-clicking on the new panel. The Script Author displays the Panel**[n]** Properties window. **Note:** The Script Author automatically assigns numbering to each new panel created (represented by **[n]** above) to enforce unique panel names. When you type a value in the Name field of the panel properties (such as "Hello") and apply the change, the value you entered replaces the **[n]** as the new name of the panel you have just created. Thus, for example, Panel5 Properties becomes PanelHello Properties. For the sake of simplicity, the name of each new panel (the value of **[n]**) will be excluded in these instructions, and the window will be referred to generically as the Panel Properties window.	Panel Properties window

Step	Perform From	Action	Result
11	Panel Properties window	In the **Panel Properties** window, input the following values in the prescribed fields: Name: Overwrite the default "Panel[n]" entry and type **hello** From this point on, only the values you should enter will be provided. In each case, overwrite the default entry. Comments: *Leave blank* Label: **Greeting**	Panel Properties window
12	Panel Properties window	Click **Apply** in the Panel Properties window. **Note:** Clicking the **Apply** button saves the values entered or changed (just as if you had clicked the **Ok** button), but leaves the window open for additional work.	Panel Properties window
13	Panel Properties window	Click the **Answers** property displayed on the left pane of the Panel Properties window. The right pane of the window changes to display a text area which will list any questions - also known as answer definitions - that you define for the script.	Panel Properties window
14	Panel Properties window	Click the **Add** button, which is immediately below the Answers text area. The Answer Entry Dialog window appears.	Panel Properties window
15	Answer Entry Dialog window	Provide the following values for the fields in the Answer Entry Dialog window: Default for Distinct Branching: ***checkmark*** Selecting this check box is optional. Panels which do not use distinct branching are not required to enable this property. Name: **helloAnswer** UI Type: **Button** Label for Reporting: *Leave blank*	Answer Entry Dialog window
16	Panel Properties window	From the bottom of the Answer Entry Dialog window, click the **Edit Data Dictionary** button. The application displays the Edit Data Dictionary dialog.	Panel Properties window
17	Edit Data Dictionary dialog	Select the Lookups tab. **Note:** Lookup values are referred to in the Script Wizard as answer choices.	Edit Data Dictionary dialog

Step	Perform From	Action	Result
18	Edit Data Dictionary dialog	On the Lookups tab, click the **Specify lookups** radio button. The Lookups list and its corresponding Add button enable. This list is currently empty.	Edit Data Dictionary dialog
19	Lookup entry dialog	Click the **Add** button. The application displays the Lookup entry dialog.	Lookup entry dialog
20	Lookup entry dialog Edit Data Dictionary dialog	In the Lookup entry dialog, assign the following values to the fields: Display Value: **Continue** Value: **Continue** Click **Ok** to close the Lookup entry dialog and return to the Edit Data Dictionary dialog. Click **Ok** to close the Edit Data Dictionary dialog and return to the Answer Entry Dialog window.	Lookup entry dialog Edit Data Dictionary dialog
21	Panel Properties window	Click **Ok** to close the Answer Entry Dialog window. The answer that you have defined now appears in the Answers text area in the Panel Properties window. It should appear as an entry with the name **Answer:helloAnswer (default)**.	Panel Properties window
22	Panel Properties window	Click **Ok** to close the Panel Properties window. On the canvas below the start node, the panel (for which you have just configured name and answer properties) appears.	Panel Properties window
23	Tools menu *or* Script Author canvas *or* keyboard	Click *once* on the panel you have just created. When the markers appear around the panel to indicate that you have selected it, open the panel layout editor. **Note:** The panel layout editor can be opened by selecting **Panel Layout Editor** from the **Tools** menu, by clicking on the **Panel Layout** icon from the toolbar, or by entering the keyboard combination command Ctrl-Shift-E (Command-Shift-E for the Macintosh OS). It can also be accessed by selecting the **Panel Layout** property displayed on the left pane of the Panel Properties window, and then selecting **Panel Layout Editor** from the Panel Layout details area.	Tools menu *or* Script Author canvas *or* keyboard

Step	Perform From	Action	Result
24	Panel layout editor Toolbar	When the panel layout editor opens in Script Author 11.5.6 or later, representative controls for any answers you have defined appear on the page, with room above for text. Click once in the text entry area. From the toolbar, click the **Set Spoken Text Font** icon. Then enter the following text: **Hello, World!** **Note:** The text is formatted as "Spoken Text," allowing one-click formatting to provide an easy visual clue. For agents, this is typically text that would be read to a customer during an interaction. For scripts run in the Web interface as a survey, this is an easy way to emphasis text or use one of two preexisting formats.	Panel layout editor Toolbar
25	Panel layout editor *or* Unsaved Changes dialog	Save the panel text you have created and exit the panel layout editor. You can do this by selecting **Save** and then **Exit** (both from the **File** menu), or you can simply exit the panel layout editor and click the **Yes** button when the Unsaved Changes dialog asks if you would like to save the changes you have made.	Panel layout editor *or* Unsaved Changes dialog
26	Script Author Toolbar	Select the **Group Insertion Mode** icon from the toolbar.	Script Author Toolbar
27	Script Author canvas	Click **once** on the canvas, below the **hello** panel. The Script Author will place a Group object into the script at the insertion point.	Script Author canvas
28	Script Author Toolbar	Click the **Toggle Selection Mode** icon from the toolbar to turn off the Group Insertion mode.	Script Author Toolbar
29	Script Author canvas	**Double-click** the newly-added group. The Group Properties dialog is displayed.	Script Author canvas
30	Group Properties dialog	Fill out the fields in the Group Properties dialog in the following manner: Name: **WrapUpGroup** Comments: **Enables the Disconnect button** **Note:** The comment is not required.	Group Properties dialog
31	Group Properties dialog	Click **Apply** in the Group Properties dialog.	Group Properties dialog

Step	Perform From	Action	Result
32	Group Properties dialog	Click the **Shortcut** property from the items on the left pane of the Group Properties dialog. The right pane is re-written and a new field appears. You must fill it out as follows: Shortcut: **WrapUpShortcut** **Note:** This Shortcut name must appear ***exactly as typed here, with upper and lowercase text and no spaces.*** For scripts run in the agent application, the Scripting Engine associates the **Disconnect** button with any group containing the Shortcut property of **WrapUpShortcut**.	Group Properties dialog
33	Group Properties dialog	Click **OK** in the Group Properties dialog. The dialog closes and the new group appears on the canvas below the **hello** panel. A new group represents a second graph. The original graph on which the **hello** panel and the **WrapUpGroup** group reside is referred to as the Root Graph.	Group Properties dialog
34	Script Author canvas	From the Script Author canvas, **click once** on the **WrapUpGroup** icon to select it. The selection markers will appear.	Script Author canvas
35	Script Author Toolbar	Click the **Go Down Into Child Graph** icon from the toolbar. You will now see the child graph of the new group, which contains only a start node.	Script Author Toolbar
36	Script Author Toolbar	Click the **Panel Insertion Mode** icon from the toolbar and **click once** in the child graph below the start node.	Script Author Toolbar
37	Script Author Toolbar	Click the **Toggle Selection Mode** icon from the toolbar to turn off the Panel Insertion mode.	Script Author Toolbar

Step	Perform From	Action	Result
38	Script Author canvas	**Double-click** the new panel to display the Panel Properties window. Click on the Properties property from the items on the left pane of the Panel Properties window, and enter the following values in the prescribed fields: Name: **goodbye** Comments: *Leave blank* Label: **Last panel** **Note:** The Script Author "remembers" the last property panel or field location displayed in any similar dialog. Thus, you may need to select a different property or field in a given dialog in order to proceed with the steps indicated in these instructions.	Script Author canvas
39	Panel Properties window	Click **Apply** in the Panel Properties window.	Panel Properties window
40	Panel Properties window	Click the **Answers** property displayed on the left pane of the Panel Properties window. The right pane of the dialog changes to display a text area which will list any answers you define for the script.	Panel Properties window
41	Panel Properties window	Click the **Add** button to define your answer in the Answer Entry Dialog window.	Panel Properties window
42	Answer Entry Dialog window	Provide the following values for the fields in the Answer Entry Dialog window: Default for Distinct Branching: ***checkmark*** Name: **goodbyeAnswer** UI Type: **Button** Label for Reporting: *Leave blank*	Answer Entry Dialog window
43	Panel Properties window	From the bottom of the Answer Entry Dialog window, click the **Edit Data Dictionary** button. The application displays the Edit Data Dictionary dialog.	Panel Properties window
44	Edit Data Dictionary dialog	On the Lookups tab, click the **Specify lookups** radio button.	Edit Data Dictionary dialog
45	Lookup entry dialog	Click the **Add** button. The application displays the Lookup entry dialog.	Lookup entry dialog

Step	Perform From	Action	Result
46	Lookup entry dialog Edit Data Dictionary dialog	In the Lookup entry dialog, assign the following values to the fields: Display Value: **Finish** Value: **Finish** Click **Ok** to close the Lookup entry dialog and return to the Edit Data Dictionary dialog. Click **Ok** to close the Edit Data Dictionary dialog and return to the Answer Entry Dialog window. Click **Ok** to close the Answer Entry Dialog window.	Lookup entry dialog Edit Data Dictionary dialog
47	Panel Properties window	Click **Apply** in the Panel Properties window.	Panel Properties window
48	Panel Properties window	Click the **Panel Layout** property displayed on the left pane of the Panel Properties window. The right pane of the dialog changes to display the Panel Layout details, including a **Browse** button from which you can import HTML, and a **Panel Layout Editor** button.	Panel Properties window
49	Panel Properties window	Click **Panel Layout Editor.** The panel layout editor for the **goodbye** panel appears.	Panel Properties window
50	Panel layout editor Toolbar	In the panel layout editor dialog, click once in the text entry area. From the toolbar, click the **Set Spoken Text Font** icon. Then enter the following text, followed by: **Goodbye, World!** Press Enter to continue to the next line, and then click the **Set Instruction Font** icon. In the Text entry area, type the following text: **[Click the Finish button.]** **Note:** Instructional text is intended to provide instructions to the end user of the script.	Panel layout editor Toolbar
51	Panel layout editor *or* Unsaved Changes dialog	Save the panel text you have created and exit the panel layout editor. Click **Ok** to save and close the Panel Properties window. The **goodbye** panel is now visible below the start node of this child graph.	Panel layout editor *or* Unsaved Changes dialog

Step	Perform From	Action	Result
52	Script Author Toolbar	From the Script Author toolbar, click the **Termination Point Insertion Mode** icon. On the canvas, below the goodbye panel, **click once** to insert a termination node. **Note:** A termination node is required for each graph in a script, and exits the graph.	Script Author Toolbar
53	Script Author Toolbar	From the Script Author toolbar, select the **Default Branch Mode** icon. When over the canvas, the cursor turns into a cross-hairs symbol, indicating that you can begin drawing a line to define flow from one object to another within the canvas.	Script Author Toolbar
54	Script Author canvas	With the cursor displayed as a cross-hairs symbol, click on the **start** node and drag to **draw an arrow** between the start node and the **goodbye** panel. Release the mouse. The two objects are connected.	Script Author canvas
55	Script Author canvas Script Author Toolbar	With the cursor displayed as a cross-hairs symbol, again click on the canvas, this time connecting the **goodbye** panel and the termination node. If you have disabled sticky mode, you must again select the **Default Branch Mode** icon from the toolbar to accomplish this step. You have now completed a syntactically correct graph. Return to the root graph to complete this script.	Script Author canvas Script Author Toolbar
56	Script Author Toolbar	From the Script Author toolbar, select the **Go up to parent graph** icon. The canvas now displays the root graph.	Script Author Toolbar
57	Script Author Toolbar Script Author canvas	From the Script Author toolbar, click the **Termination Point Insertion Mode** icon. On the canvas, below the WrapUpGroup object, **click once** to insert a termination node. **Note:** For the root graph, the termination node indicates the completion of the script.	Script Author Toolbar Script Author canvas

Step	Perform From	Action	Result
58	Script Author Toolbar	From the Script Author toolbar, select the **Default Branch Mode** icon. Connect the start node to the hello panel, the hello panel to the WrapUpGroup object, and the WrapUpGroup object to the termination node with default branches.	Script Author Toolbar
59	Script Author Toolbar	Click the **Toggle Selection Mode** icon from the toolbar to turn off the **Default Branch** insertion mode.	Script Author Toolbar
60	Script Author Toolbar	Using click-and-drag techniques, select all the objects on the canvas (start node, hello panel, termination node, and Default branches). From the lower Tools menu, select the **Align Selected Objects Vertically** icon. All highlighted items will align vertically, presenting a neat graph on the canvas.	Script Author Toolbar
61	Script Author File Menu *or* Script Author Toolbar *or* keyboard	At the **File** menu, select **Save** to save your work. This command can also be executed by selecting the Save Script (disk) icon in the toolbar or by entering Ctrl-S from the keyboard (Command-S for the Macintosh OS).	Script Author File Menu *or* Script Author Toolbar *or* keyboard
62	Script Chooser dialog	In the resulting Script Chooser menu, you can save the script to your local file system or publish it to the applications database. Fill in the following: Location: Select the directory in which you wish to save the Script Author file. **Note:** For location, Oracle Corporation recommends **a child directory of the Script Author directory**, for example (your path may differ): `"[OraHome]\oracle\apps\ies\author\scripts"` File Type: Select the default **Oracle CRM script file (**.SCRIPT, .SCR) option. File name: **helloWorldTestScript.script** Click the **Save** button to save the file in the selected directory.	Script Chooser dialog

Step	Perform From	Action	Result
63	Script Author canvas	Even though you have saved and named your script in your operating system's file system, you still need to assign a name to the script which will be recognized from a database perspective. Right-click on any unpopulated section of the canvas to display two options, Edit and Edit Blob Properties, and select **Edit Blob Properties** to open the [Script Name] Properties dialog. You can also open this dialog by selecting **Script Properties** from the **File** menu. **Note:** If the dialog displayed shows *three* options: **Edit, Edit Blob Name**, and **Edit Blob Properties,** then you have not correctly selected the Script properties dialog, but properties for another object. The global script itself is the Binary Large Object (Blob) for which you want to modify properties. In this case, left-click to close the menu and re-select the **Script Properties** following one of the methods above.	Script Author canvas

Step	Perform From	Action	Result
64	Global [Script Name] Properties dialog	In the resulting global [Script Name] Properties dialog (which should currently be named something like **Untitled1 Properties**), fill in the following: Name: **helloWorldTestScript** Comments: **Graphical hello world test script, built using Script Author visual layout tools.** Script language: As appropriate for your environment.For example, **AMERICAN** for American English. Click the **Ok** button to save the global script properties and return to the canvas. **Note:** For the Name field, fill in a name under which this script will be deployed to the database. We recommend using the same name as that you assigned to the .SCRIPT file in the file system, if practical. Do not assign a file extension to this name. For the Comments field, you may elect to fill this in with information you will find helpful to identify this particular script or script version. This field is for information only and is not processed in any way. For the Script language field, enter the language in which this script is developed, based on the FND_ LANGUAGES table of the Oracle 8*i* database. For this example, ensure this field reads **AMERICAN**, for American English.	Global [Script Name] Properties dialog
65	Tools menu or Check Syntax Script Author Toolbar icon	Select **Syntax Check** from the **Tools** menu, or click the Check Syntax icon in the Script Author toolbar. Assuming all the syntax in your script is correct, a status line at the bottom of the Author will display the message "Syntax check successful, with no errors." If the syntax check fails, correct any errors and check the syntax of the script again until successful.	Tools menu or Check Syntax Script Author Toolbar icon
66	File Menu *or* Script Author Toolbar *or* keyboard	At the **File** menu, select **Save** again, to ensure you save the Global Script properties you assigned. **Note:** The method used to save the document is not important. You may elect to click the Save icon in the Script Author toolbar or enter Ctrl-S from the keyboard (Command-S for the Macintosh OS).	File Menu *or* Script Author Toolbar *or* keyboard

Step	Perform From	Action	Result
67	Script Author Tools menu *or* Script Author Toolbar	From the Tools menu, select **Deploy Script**, or from the Script Author toolbar, click the **Deploy Script to DataBase** icon. Upon successful deployment, a status message appears on the bottom of the Script Author frame, "Deployment to database successful."	Script deploys, as indicated by status message
68	Forms-based applications	To launch this script in the agent interface, you must perform the following actions: ■ Log into Oracle applications using the Personal Homepage login (or the Single Sign-On login, if implemented) with the appropriate responsibility to launch a script. For example, to launch the script in standalone mode, the user must have the Scripting User or Scripting Agent responsibility. ■ From the Navigator, select **Scripting Demo Form** and click **Open**. The Script Chooser window appears. This window is entitled Oracle Scripting. From the Script Chooser, select the appropriate script name and language. For example, select **helloWorldTestScript/AMERICAN**. ■ Launch the script by clicking **Start Scripting**.	Forms-based applications

Step	Perform From	Action	Result

69	Forms-based applications	To launch this script as a survey in a Web browser, you must perform the following actions:
		- Log into Oracle applications using the Personal Homepage login (or the Single Sign-On login, if implemented) with a user that has the Survey Administrator responsibility
		- Define header section, footer section (optional), error page and final page survey resources.
		For example, you can create references to seeded test survey resources in the Survey Resources tab. These are named IESSVYTESTHEADER.JSP, IESSVYTESTERROR.JSP and IESSVYTESTTHANKU.JSP.
		If using the footer section, provide the test header section resource name for this field.
		- Create a survey campaign, cycle, and standard (non-list-based) deployment. The survey campaign must identify a deployed 11.5.6 or later test script.
		- Activate the deployment. When the Survey Web URL field is populated with a hyperlink, note the deployment ID (dID) by placing the cursor over the hyperlink and viewing the full survey URL in the Web browser status field. The format of the survey URL should be as follows:
		`http://<server>.<domain>:`
		`<port>/OA_HTML/<hosted or`
		`standalone survey jsp>?`
		`dID=xx`
		- Enter the survey URL in a Web browser to launch the script in the Scripting Engine Web interface.
		- Execute the script by participating in the survey to completion.
			Forms-based applications

See Also

- Creating a Hello World Wizard Test Script
- Using the Regression Test Script

10.1.2 Creating a Hello World Wizard Test Script

Step	Perform From	Action	Result
1	Oracle Applications HTML Login	Log into Oracle applications using the CRM Homepage login (or the Single Sign-On login, if implemented) as a user with the Scripting Administrator responsibility.	The Scripting Administration console appears.
2	Scripting Administration console	From the Home tab, click **Launch Script Author**.	A new browser window opens, Oracle JInitiator launches, and the Script Author Java applet appears in a separate window.
3	Script Author File Menu *or* Script Author Tools Menu *or* keyboard *or* Script Author Toolbar	Start the Script Wizard using one of the following methods: From the File menu: 1 Select **New**. 2 In the resulting New Script window, click **Wizard Script**. From the Tools menu: 1 Select **Wizard**. 2 In the Choose Create or Edit window, select **Create Wizard Script** and click **Next**. From the keyboard: 1 Click **ALT + T** to select the Tools menu. 2 Type **W** to select the Wizard option. 3 Using the **Tab** key to navigate through wizard page options and the **space bar** to select the active option, in the Choose Create or Edit window, select **Create Wizard Script** and then select **Next**. From the Script Author toolbar: 1 Click **Script Wizard**. 2 In the Choose Create or Edit window, select **Create Wizard Script** and click **Next**.	The Script Wizard launches. The Define Script Properties window appears

Step	Perform From	Action	Result
3	Script Wizard - Define Script Properties window	Provide the following global script properties to this wizard script. Global script properties are properties that apply to the entire script, rather than a particular panel. Script Name: **helloWorldWizardScript** Description: **Hello world test script, built using Script Author Script Wizard.** Language: **AMERICAN** When complete, click **Next**.	The wizard script is assigned global script properties.
4	Script Wizard - Define Panel Information for Panel Default Panel 1	Provide panel information for the first panel. Panel Name: **hello** Panel Text: Overwrite the default "Enter text here" entry and type **Hello, World!** From this point on, only panel text values will be provided. In each case, overwrite the default text. Panel Text Style: **Spoken Text** Text Alignment: **Left** Question Alignment: **Left** Exit Panel Sequence: **Go to the next panel in sequence** When complete, click **Next**.	Panel, question, and answer choice properties generally appear at the top of each wizard page to assist you in navigating through the creation or editing process. For this new wizard script, until you click **Next**, none of these properties are defined. A temporary name (Default Panel 1) is automatically supplied by the wizard.

Step	Perform From	Action	Result
5	Script Wizard - Question Manager for Panel hello	After defining panel properties, if relevant, you must define question properties for each question in the panel. This is performed from the Question Manager Since the only function for panel hello is to display panel text and a Continue button, you do not need to define any questions for this panel. Unlike when using Script Author's visual layout tools, the wizard will automatically create the Continue button to route the runtime user to the destination specified using the Exit Panel Sequence property. Based on the panel properties established in the previous step, this script at runtime will display the next panel you define. To save this one-panel script and exit, click **Save**, and proceed to step 17. To continue learning how to create panels, define answer types and questions, and determine panel branching, from the set of navigation buttons to the right of the Question Manager window, select **Panel Manager** and continue this procedure.	Upon selecting Panel Manager, the Panel Manager window appears.
6	Script Wizard - Panel Manager for Script helloWorldWizard Script	Once you have one or more panels defined, from the Panel Manager window you can see a list of existing panels and their status. From this window you can also create new panels, edit the properties of existing panels, copy existing panels, and delete existing panels. If more than one panel is listed here, you can also move panels higher and lower in sequence. From the set of navigation buttons to the right of the Panel Manager window, select **Create**.	Define Panel Information window appears for new panel.
7	Script Wizard - Define Panel Information for Panel Panel2	Provide panel information for the second panel. Panel Name: **goodbye** Panel Text: **Goodbye, World!** Panel Text Style: **Spoken Text** Text Alignment: **Left** Question Alignment: **Left** Exit Panel Sequence: **End Script** When complete, click **Next**.	The Script Wizard progresses to the Question Manager for panel goodbye.

Step	Perform From	Action	Result
8	Script Wizard - Question Manager for Panel goodbye	After defining panel properties, if relevant, you must define question properties for each question in the panel. This is performed from the Question Manager. For this panel you will create a question. From the set of navigation buttons to the right of the Question Manager window, select **Create**.	The Script Wizard progresses to the define question main properties window.
9	Script Wizard - Define Question Main Properties	Questions have three main properties. These are similar to properties in a graphical script created using the Answer Entry dialog. The *question name* identifies this question. This is identical to the name property in a graphical script. The *question label* identifies this script to reporting features accessible through Oracle Discoverer. This is equivalent to the Label for Reporting property in a graphical script. The *type* identifies the answer control displayed at runtime. Wizard scripts can contain a text box (text), text area, radio button, drop-down list, checkbox group, multi-select list, or a password field. This is equivalent to the UI type property in a graphical script. Question Name: **measureSuccess** Question Label: **script successful?** Type: **Dropdown** When complete, click **Next**.	The Script Wizard progresses to the answer manager window.
10	Script Wizard - Answer Manager for measureSuccess Question	The answer manager provides wizard script developers the opportunity to provide answer choices to a specific question. (Answer choices are referred to as lookup values when using graphical scripts.) In this case, you will create three answer choices for the measureSuccess question you created. From the set of navigation buttons to the right of the Answer Manager window, select **Create**.	The Script Wizard progresses to the define answer choice window.

Step	Perform From	Action	Result
11	Script Wizard - Define Answer Choice for Untitled	The question types supported by wizard scripts include those questions that can accept only a single answer (text box, text area, password field, drop-down list or button), and those that can accept more than one answer choice (multi-select question types, including checkbox group and multi-select list question types). For text question types (text box, text area or password field answer choices), as well as checkbox group and multi-select list choices, a null answer (no input provided by end user at runtime) is also supported. Radio buttons, drop-down lists, multi-select lists, and checkbox groups must have answer choices provided by the script developer in order to display answer choices at runtime. Answer choices have two properties: the *answer value* (which is passed to the database) and the *answer label* (the answer choice viewed by the script end user). This is equivalent to lookup type value and display value fields in the data dictionary of the graphical Script Author. For this question, which will ask the script end user to determine the level of success for this test script, provide three answer choices. For the first set of answer choices, enter the following: Answer Value: Overwrite the default "Untitled" entry and type **Unsuccessful**. From this point on, only an answer value will be provided. In each case, overwrite the default text. Answer Label: Overwrite the default "Untitled" entry and type **Unsuccessful**. From this point on, only an answer value will be provided. In each case, overwrite the default text. You can also set a default answer choice selection. This will default in the appropriate answer control at runtime, although defaults can be changed at will. For the question "Should this be the default answer for this question," select **No**. This version of the script will use the panel level defaults. Select **Go to next default panel.** When complete, click **Next**.	The Script Wizard refreshes the answer manager, showing the single answer choice currently defined for this question.
12	Script Wizard - Answer Manager for measureSuccess Question	You must now configure another answer choice to display at runtime for this question. From the set of navigation buttons to the right of the Answer Manager window, select **Create**.	The Script Wizard progresses to the define answer choice window.

Step	Perform From	Action	Result
13	Script Wizard - Define Answer Choice for Untitled	For the second answer choice, enter the following: Answer Value: **Moderately successful** Answer Label: **Moderately successful** For the question "Should this be the default answer for this question," select **Yes**. This version of the script will use the panel level defaults. Select **Go to next default panel.** When complete, click **Next**.	The Script Wizard refreshes the answer manager, showing the two answer choices currently defined for this question.
14	Script Wizard - Answer Manager for measureSuccess Question	You must now configure the third and final answer choice to display at runtime for this question. From the set of navigation buttons to the right of the Answer Manager window, select **Create**.	The Script Wizard progresses to the define answer choice window.
15	Script Wizard - Define Answer Choice for Untitled	For the third answer choice, enter the following: Answer Value: **Very successful** Answer Label: **Very successful** For the question "Should this be the default answer for this question," select **No**. If you choose Yes, you will be reminded that this will overwrite the selection of Moderately successful as the default answer choice for this question. This version of the script will use the panel level defaults. Select **Go to next default panel.** When complete, click **Next**.	The Script Wizard refreshes the answer manager, showing all three answer choices defined for this question.
16	Script Wizard - Answer Manager for measureSuccess Question	You have now finished defining two panels, the second containing a single question with three answer choices. At this point, save your work. Click **Save**.	The Script Wizard progresses to the save script window.
17	Script Wizard - Save Script helloWorldWizard Script	From the save script window, for the prompt Do you want to save the script with a different name, select **No**. When complete, click **Next**.	The Script Wizard progresses to the continue with script save window.

Step	Perform From	Action	Result
18	Script Wizard - Continue With Script Save	You can choose one of the following save options: ■ To save the current wizard script and continue editing it, select **Save and continue editing**. ■ To save the wizard script in the database without deploying it, close the wizard tool, and view the Script Author canvas, select **Save and exit**. ■ To save the wizard script, deploy it to the database, close the wizard tool, and view the Script Author canvas, select **Save, deploy and exit**. Select the last option, **Save, deploy and exit**. When complete, click **Next**.	The Script Wizard progresses to the create graphical script copy window.
19	Script Wizard - Create Graphical Script Copy	From this window, by clicking Yes you can view the wizard script you have completed, using Script Author's visual tools. However, the script displayed is a *copy* of the wizard script. Any changes made in the resulting graphical script will *not* be reflected in the original wizard script, which will still be accessible to open, edit, view and deploy from the Script Wizard. Select **Yes** and click **Next**.	The Script Wizard progresses to the script is complete window.
20	Script Wizard - Script is Complete for helloWorldWizard Script	Click **Finish** to exit the wizard and return to the Script Author canvas. The wizard script then deploys to the database per your request, and the flow you created appears on the Script Author canvas. Note that this graphical copy is entitled "Copy of helloWorldWizardScript." At this point, you have the option of saving this graphical copy of the wizard script to the database or your local file system. Since you can always open the wizard script and graph a copy to use with Script Author's graphical tools, select **File > Close** to close the graphical Script Author without saving. In the resulting Select an Option window, click **No** to exit without saving.	Copy of helloWizardScript appears on the Script Author canvas. After closing the script, the graphical Script Author user interface appears, with no script open.

Step	Perform From	Action	Result
21	Script Author File Menu *or* Script Author Tools Menu *or* keyboard *or* Script Author Toolbar	Start the Script Wizard again, using one of the following methods: From the File menu: ■ Select **Open**. ■ In the resulting Open Script window, click **Wizard script**. From the Tools menu: ■ Select **Wizard**. ■ In the Choose Create or Edit window, select **Edit Wizard Script** and click **Next**. From the keyboard: ■ Click **ALT + T** to select the Tools menu. ■ Type **W** to select the Wizard option. ■ Using the **Tab** key to navigate through wizard page options and the **space bar** to select the active option, select **Edit Wizard Script** and click **Next**. From the Script Author toolbar: ■ Click **Script Wizard**. ■ In the Choose Create or Edit window, select **Edit Wizard Script** and click **Next**.	The Script Wizard launches.

Step	Perform From	Action	Result
22	Script Wizard - Script Manager	Scroll down through the list of existing wizard scripts if necessary to locate the appropriate script. Scripts are listed by global script name and include last revised date and script language. For any wizard script in the list, you can perform the following: ■ To modify any panel, question, or answer choice details, select the appropriate script from the list, and from the wizard function buttons to the right of the wizard scripts table, click **Edit**. ■ To retain the original wizard script but to make a copy of the script to modify, select the appropriate script from the list and click **Copy**. ■ To delete the wizard script from the database, select the appropriate script from the list and click **Delete**. ***Caution:*** This will permanently delete the selected script. Unless you have made a copy of this wizard script, *you will not be able to modify, edit, or execute this script after deletion.* ■ To create a graphical copy of this script for viewing or modification with the Script Author's visual editing tools, select the appropriate script from the list and click **Graph**. This will close the Script Wizard. ■ To deploy a saved wizard script to the database, select the appropriate script from the list and click **Deploy**.	The selected function will execute.

See Also

- Creating a Hello World Graphical Test Script
- Using the Regression Test Script

10.1.3 Using the Regression Test Script

Oracle Corporation provides a regression test script for the purpose of testing Oracle Scripting implementations and functionality. This script contains references to custom Java. Custom Java classes, compiled into JAR files, are included in the package and must be managed accordingly.

Prerequisites

- To create or deploy scripts using the Script Author Java applet, you must use the Scripting Administration console. Thus, you must have access to an Oracle Applications account with the Scripting Administrator responsibility.
- All implementation tasks must be accomplished.

Steps

To use the regression test script, perform the following:

1. Download and unzip the regression test script files.

   The regression test script is accessible from Oracle*MetaLink* and Oracle iSupport as note 124522.1. To locate, enter the document ID in the Search field. From the support document entitled Oracle Scripting Regression Testing Scripts, download the associated ZIP archive and unzip this package. Included are a Script Author file, a compiled Java class file, and a packaged Java archive.

2. Using the Script Author Java applet, open the Script Author file. Note the global script name (**File > Script Properties**) and change this if necessary to prevent overwriting any existing script with the same name in your environment.

3. Deploy the regression test script to the database from the Script Author Java applet (**Tools** > **Deploy Script**).

   You can view deployed scripts from the Scripting Administration console (Administration tab > Deployed Scripts subtab).

4. Upload the custom Java archive file to the Oracle Applications database from the Scripting Administration console (Administration tab > Jar Listings subtab).

5. Map the script to the custom Java archive from the Scripting Administration console (Administration tab > Jar Mapping subtab).

   The regression test script is now appropriately deployed, and can be tested using any Scripting Engine interface. To test using the Web interface, survey campaign requirements must be established using this script.

References

- Creating a Hello World Graphical Test Script

See Also

- Creating a Hello World Graphical Test Script
- Creating a Hello World Wizard Test Script

10.2 Survey Administration Console Tasks

To verify your implementation for scripts to be executed in the Scripting Engine Web interface, you must establish survey campaign requirements. Steps to perform for implementations that require scripts to be executed in an Oracle Applications 11*i*-compatible Web browser are described below.

Prerequisites

- To use the Survey Administration console, you must have access to an Oracle Applications account with the Survey Administrator responsibility.
- All implementation tasks must be accomplished. For enterprises using Oracle Applications on the UNIX platform, you must have a display X Server set up.
- To access the Invitations tab in the Survey Administration console, you must be a member of a fulfillment group and have the JTF_FM_ADMIN role.
- To establish invitations and reminders for targeted (list-based) survey operations, functioning invitation and reminder master documents and their associated queries must exist for Oracle One-to-One Fulfillment.
- Invitations and master documents must be associated with Oracle One-to-One Fulfillment templates, as appropriate.
- To establish list information in a deployment, you must have a corresponding, accurate list of 150 or fewer records created in Oracle Marketing.

Steps

1. Identify existing physical survey resources or create physical survey resources to use with your script in the Web interface.

   Physical resources are the actual HTML or JSP files that display at runtime. In the case of a URL for redirect, this file typically already exists, loaded on a Web site accessible to the Apache Web server at runtime.

   For OAF survey campaigns, survey resources are not required. If you do not associate *section* resources with a survey campaign, no header section or footer section appears at runtime. If you do not associate *page* resources, application defaults are used for the error page or final page, as appropriate, at runtime.

   Physical survey resources will be required if you choose to associate survey resources with an OAF survey campaign. No seeded HTML resources are available.

For JTT survey campaigns, you can use seeded JSP test resources to test your implementation, or you can customize your own resources based on these seeded resources. These seeded JavaServer page files (IESSVYTESTHEADER.JSP, IESSVYTESTTHANKU.JSP, IESSVYTESTERROR.JSP and IESSVYMENUBASEDERROR.JSP) are located in $OA_HTML in your environment. For error pages with a hosting type of menu-based, use the appropriate seeded resource (IESSVYMENUBASEDERROR.JSP.

2. Using the Survey Administration console, define survey resources (**Survey Resources** > **Create**).

   OAF survey campaigns do not *require* any resources to be associated in order to execute a script in a Web browser.

   - If using HTML survey resources, upload the physical HTML files into the database at the time of survey resource definition.
   - If using URLs for redirect, define each URL.

   For JTT survey campaigns, you must define (at minimum) a header section, error page, and final page resource.

   - For JTT survey campaigns, upload to the $OA_HTML directory on the APPL_TOP the physical JSP files which correspond to survey resource definitions.
   - For assistance or to obtain the correct privileges, consult with your systems administrator or your Web server administrator.

3. Using the Survey Administration console, create a survey campaign (**Survey Campaign** > **Create**). Identify the script you want to use as your survey questionnaire or Web script (for example, the Hello World or regression test script). For survey resources, reference the resources identified in steps 1 and 2 above.

   - The base technology stack you select for a survey campaign cannot be changed once the survey campaign is saved. For verification purposes, select the technology stack appropriate for script execution at runtime. This determines the way the script is processed in a Web browser at runtime.
   - You may choose to create a prototype survey campaign for verification. Prototype survey campaigns are identical to standard survey campaigns, except that the script used as the survey campaign questionnaire is not locked. Locking the script protects the integrity of the data collected during the course of a survey campaign. Designating the script as a prototype

provides you with flexibility; in the event that one test script is not functioning, you can change the script associated with the survey campaign without needing to recreate the survey campaign and all its dependent objects. This characteristic can only be selected or cleared while a survey campaign status is Open.

4. Within an existing survey campaign, create a cycle (as a child to the survey campaign).

   - To create a cycle from the Create Survey Campaign page, type a name in the Cycle Name field prior to clicking **Create**.
   - Alternatively, from the Update Survey Campaign page, in the Cycle and Deployments area, click **Add Cycle**, and in the Add Cycle page, type a name in the Cycle Name field and click **Submit**.

5. Create a deployment as a child to an existing survey campaign and cycle. From the Update Survey Campaign page, in the Deployments area, click **Create**.

   - For deploy date, use a date prior to or equal to SYSDATE. The time portion of the date and time string will be ignored for this parameter only.
   - For standard deployments, after providing click **Create**.
   - For targeted deployments, select Targeted from the Deployment Type list and click **Go**. In the expanded Create Survey Deployment page, include all required parameters, and click **Create**.

   Only enter list and reminder information if you have a corresponding, accurate list created in Oracle Marketing and you have functioning invitation and reminder master documents associated with a fulfillment template using Oracle One-to-One Fulfillment functionality.

6. Activate your deployment by clicking **Activate** in the Update Deployment page. For appropriate deployment types, note the survey URL.

7. If scripts will be executed using the Using an Oracle Applications 11*i*-compatible Web browser, access the survey URL (from an invitation or reminder, if a targeted survey campaign), and execute your script in the Scripting Agent Web interface. Test all paths in the script to ensure the implementation is successful.

8. Using an Oracle Applications 11*i*-compatible Web browser, access the survey URL. If a targeted survey campaign, access the URL from an invitation or reminder. Test your script in the Scripting Engine Web interface. Test all paths in the script to ensure the implementation is successful.

See Also

- Script Author Tasks
- Scripting Engine Tasks
- Testing Display Server for UNIX Environments
- Viewing Server Message Logs

10.3 Scripting Engine Tasks

To verify that the Scripting Engine will execute scripts, you can execute servlet tests designed to ensure scripts will execute in the Apache mid-tier architecture.

Then, test a simple script with no custom Java (such as Hello World), and a script containing custom Java (such as regression test), using the Scripting Engine.

> **Note:** Oracle Corporation recommends that all implementations test first using the agent interface. Implementations using the Web interface must then also perform the appropriate verification steps.

Each Oracle Scripting implementation includes at least one custom script. Oracle Corporation recommends, at minimum, that the first script used at an enterprise is developed by Oracle Corporation consultants or partners. In addition to testing the recommended Hello World and regression test scripts, it is recommended that you completely test your custom script prior to putting that script into production.

If custom Java is required, you must create and test your custom methods, compile and package your Java archive appropriately, and deploy the Java archive to your environment. To test these methods from Oracle Scripting, your script must contain valid references to the code, and must also be deployed to your environment.

This section includes the following topics:

- Apache Web Server Servlet Verification URLs
- Scripting Engine Agent Interface Verification
- Scripting Engine Web Interface Verification Tasks

See Also

- Script Author Tasks
- Survey Administration Console Tasks

- Testing Display Server for UNIX Environments

10.3.1 Apache Web Server Servlet Verification URLs

Script Servlet Verification

In order to execute Oracle Scripting in the agent application using the Apache Mid-Tier architecture, the Scripting Servlet must be operational.

The following URL can be used to verify that the Scripting Servlet is properly configured and operational:

```
http://<servername>.<domain>:<Apache Web server port>/<servlet_
zone>/oracle.apps.ies.client.application.ScriptingServlet
```

A message of **up and running...** indicates success. This message *must* be obtained to assure that scripts can be executed appropriately.

Apache JServ Servlet Verification

The following URL can be used to verify that the Apache JServ servlet engine is configured and operational:

```
http://<servername>.<domain>:<Apache Web server port>/<servlet_zone>/IsItWorking
```

A message of **Yes, It's working! Congratulations, Apache JServ/<version> is working!** indicates success. Note, however, that this servlet may have been removed by system administration and is not necessarily an indication of whether the Scripting Servlet will function.

AOL Test Servlet Verification

The following URL can be used to verify that the AOL test servlet is configured and operational:

```
http://<servername>.<domain>:<Apache Web server port>/<servlet_zone>/
oracle.apps.fnd.test.HelloWorldServlet
```

A message of **Hello World! From AOL oracle.apps.fnd.test.HelloWorldServlet** indicates success.

Note, however, that this servlet may have been removed by system administration and is not necessarily an indication of whether the Scripting Servlet will function.

See Also

- Scripting Engine Agent Interface Verification
- Scripting Engine Web Interface Verification Tasks

10.3.2 Scripting Engine Agent Interface Verification

Execute a script without custom Java using the Scripting Engine agent interface. For example, execute the Hello World test script to test your implementation. Then, execute a script that references custom Java using the Scripting Engine agent interface. For example, execute the regression test script to test your implementation.

> **Note:** Since testing for *all* customized scripts is recommended at the lowest HTML interpretation engine available, steps for implementing the Scripting Engine for the agent interface are recommended to be performed for all implementations.

Prerequisites

- To verify scripts using the Scripting Engine agent interface, you must have access to an Oracle Applications account with the Scripting User or Scripting Agent responsibility. This is recommended for all implementations.
- All implementation tasks must be accomplished.

Steps

1. Using the Scripting Engine agent interface, log into Oracle applications using the Personal Homepage login (or the Single Sign-On login, if implemented) as a user with the Scripting User or Scripting Agent responsibility.
2. From the Script Chooser, select and execute the regression test script in the Scripting Engine agent interface. Test all paths in the script to ensure the implementation is successful.

See Also

- Apache Web Server Servlet Verification URLs
- Scripting Engine Web Interface Verification Tasks

10.3.3 Scripting Engine Web Interface Verification Tasks

Once you can execute a script in the Web interface (based on an existing survey campaign deployment), you have verified your implementation.

Tasks

You can perform the following tasks:

- Executing a Script in the Web Interface

See Also

- Apache Web Server Servlet Verification URLs
- Scripting Engine Agent Interface Verification

10.3.3.1 Executing a Script in the Web Interface

Prerequisites

- To verify scripts in the Scripting Engine Web interface, survey campaign requirements must be established by a survey administrator.
- Survey resources referenced by the appropriate survey campaign must exist in $OA_HTML.
- A guest user account must already be established by an Oracle CRM applications system administrator.
- Advanced JTF properties must be set up by an Oracle CRM applications system administrator.
- You must have access to a survey URL for an active survey campaign deployment.
- For targeted survey deployments, you may require a survey URL with a unique respondent ID.

Steps

1. Using an Oracle Applications 11*i*-compatible Web browser, access the survey URL for the survey deployment you wish to test.
2. Execute your script in the Scripting Agent Web interface. Test all paths in the script to ensure the implementation is successful.

Performing any verification task other than testing the Guest User requires you to also perform all Survey implementation steps. These are in addition to Scripting Engine implementation tasks for all implementations and those required for the Web interface, since testing a script in the Web interface requires a survey campaign to be defined and deployed.

Note: Since testing for *all* customized scripts is recommended at the lowest HTML interpretation engine available, steps for implementing the Scripting Engine for the agent interface are *recommended to be performed for all implementations*.

10.4 Testing Display Server for UNIX Environments

Implementations of Oracle Scripting using UNIX servers require a display server to be set up to process graphics in order to view reports generated under the Reports tab of the Scripting Administration console.

Note: Interaction Center Family Pack P and later implementations no longer require a functional display server for a survey respondent to participate in a survey.

Note: Interaction Center Family Pack Q and later implementations no longer require a functional display survey for the Survey Administration console.

You can verify that the display server is appropriately configured by accessing a seeded JSP test page and viewing the resulting message.

The following URL can be used to verify that the display server is properly configured and operational:

```
http://<servername>.<domain>:<Apache Web server port>/OA_HTML/
iestestdisplayserver.jsp
```

Reference

For more information on the display server, see Setting Display Server for UNIX Environments.

See Also

- Script Author Tasks
- Survey Administration Console Tasks
- Scripting Engine Tasks
- Viewing Server Message Logs

10.5 Viewing Server Message Logs

Use this procedure to view message logs for Oracle Scripting.

Prerequisites

- As of Interaction Center Family Pack R or later or release 11.5.10, the FND: Diagnostics system profile must be set to **Yes** (typically at the user level) in order to access message logs.
- You must view message logs from an Oracle Applications framework application such as the Survey Administration console.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click **Diagnostics**.

   The Diagnostics page appears.

2. Select **Show Log** and click **Go**.

   The Search page appears. This is where you define criteria for message logs you want to view.

3. In the Parameters area, if you want to view message logs generated for the Oracle Applications user account under which you are currently logged in, then in the User ID field, leave the default value.

If you want to view message logs generated for a different user, you must first obtain the FND User ID of the appropriate user. For more information, see Finding the FND User ID Value of a Defined User.

4. In the Session ID field:

   - If you do not want message logs to be restricted to any session, then delete any value appearing in this field.
   - If you want to view message logs for your current session, then leave the default value.
   - If you know the session ID for a different session for which you explicitly want to view message logs, type this value.

5. Optionally, in the Message Text field, enter any message text you want to find.

6. Optionally, in the Begin Date field, select a date from which message logs were generated.

7. Optionally, in the End Date field, select the last date for which you want to retrieve message logs.

8. In the Categories area, clear any logging levels for which you do not want to view message logs.

   These are listed in sequence from least amount of information (and most critical) to most information.

   For example, clear all values except Unexpected and Error.

9. In the Module field, if you want to restrict message logs to a particular product or set of Java classes, enter your filter criteria.

   For example, to restrict message logs to all messages logged for Oracle Scripting, enter IES% or ies%. To restrict messages to the Scripting Engine component, enter IES.ENGINE% or ies.engine%.

   Include the percent (%) sign, which designates that you want to see all logs in the designated category.

10. Optionally, in the Sequence area, define a sequence.

11. Click **Go**.

    After processing, the Search field refreshes. Message logs meeting the specified criteria appear in the Results area.

12. If you want to export the message logs, click **Export**.

13. If you want to return to the Survey Administration console, click **Return to Application**.

References

For specifics on the profiles required for viewing FND server logs, see Oracle Scripting Profile Options > Information, Error and Debug Logging Profile Options.

See Also

- Script Author Tasks
- Survey Administration Console Tasks
- Scripting Engine Tasks
- Testing Display Server for UNIX Environments

11 Integrating Oracle Scripting

There are various ways to view integration of Oracle Scripting. Since Oracle Scripting consists of several components, you can consider integration from this perspective. Since Scripting involves technologies using applications built with Oracle Forms Developer (referred to as Oracle Forms-based applications) *and* applications using Java Server Pages (JSP) technology built on HTML, you can consider integration by technology stack. Finally, Scripting includes pre-built integration, while also allowing custom integration with Application Program Interfaces (APIs). Integration can also be described from this perspective. Each is described in turn below.

This section includes the following topics:

- Integration by Component
- Integration by Technology Stack
- Prebuilt Integration with Oracle Applications

11.1 Integration by Component

Oracle Scripting consists of Script Author, Scripting Engine, Scripting Administration console, and Survey Administration console. Each has different integration considerations.

This section includes the following topics:

- Script Author
- Scripting Engine
- Scripting Administration Console
- Survey Administration Console

See Also

- Integration by Technology Stack
- Prebuilt Integration with Oracle Applications

11.1.1 Script Author

The Script Author is the component of Oracle Scripting used to create, modify and deploy scripts. As of Oracle Applications release 11.5.8 (CRM Family Pack P), Script Author is a Java applet that is launched from an existing Oracle Applications session.

To launch Script Author, log into Oracle applications using the CRM Homepage login (or the Single Sign-On login, if implemented) as a user with the Scripting Administrator responsibility, and click **Launch Script Author** from the Home tab of the Scripting Administration Console. Oracle JInitiator launches in a separate browser window, and then Script Author opens in a separate applet window on the client. Since the connection is established and associated with a specific database and applications instance, using the Script Author Java applet allows script developers to open, save, and deploy scripts to and from that database instance. Database connection information is not required to be specified to save commands to the command library or to import library commands into a script.

The Script Author Java applet replaces a standalone Script Author Java application available in prior releases which required a Windows operating system on the client. Using the standalone application also required the client to be on the same side of the firewall as the applications database, and necessitated apps database user access to deploy scripts or otherwise access the database.

The Script Author applet can be used on the same computer as any other Oracle Application, but as a Java applet is not in itself integrated with other applications.

See Also

- Scripting Engine
- Scripting Administration Console
- Survey Administration Console

11.1.2 Scripting Engine

The Scripting Engine is the runtime component of Oracle Scripting. Scripts created using Script Author are executed in one of two interfaces: the Scripting Engine Agent Interface and the Scripting Engine Web Interface.

This section includes the following topics:

- Scripting Engine Agent Interface
- Scripting Engine Web Interface

See Also

- Script Author
- Scripting Administration Console
- Survey Administration Console

11.1.2.1 Scripting Engine Agent Interface

The first of the Scripting Engine's two interfaces is the agent interface, which is a set of Java components wrapped in an Oracle form. Users of the agent interface are typically interaction center agents. The Scripting Engine agent interface provides enterprises with a method of scripting interactions with customers or prospects and integrating desktop workflow between various applications.

Integration with the Scripting Engine agent interface occurs in three ways:

- Scripts can be launched from three integrated Forms-based business applications (Oracle TeleSales, Oracle Collections, and Oracle TeleService).
- Scripts can get, create, or update records in the applications database or in custom tables.
- Scripts can get or set information from any open Oracle form in Oracle TeleService, using the Script Author Forms commands *setValueInForm* and *getValueFromForm*.
- Using custom Forms commands, scripts may get and set information from some other Forms-based applications.

The agent interface is intended to be used in combination with integrated Oracle E-Business Suite applications to take full advantage of the contact handling, sales, service, or collections features of those applications. Interaction centers leveraging business applications and the Scripting Engine agent interface therefore have the full benefit of interaction center communications.

Agents can also execute scripts in "standalone" mode, without accessing other applications. In the standalone model, integration is typically limited to reading or writing to the Oracle applications database. Login to an Oracle Applications session is still required. Execution of scripts in standalone mode is not supported by Oracle Support Services.

See Also

- Scripting Engine Web Interface

11.1.2.2 Scripting Engine Web Interface

The Scripting Engine Web interface executes scripts in any Oracle Applications 11*i*-certified Web browser. Scripts executed in the Web interface rely on survey campaign information administered using the Survey component.

Self-Service Web Application Integration

The Scripting Engine Web interface is integrated with self-service Web applications such as Oracle iSupport. After using the application, customers can be surveyed regarding their self-service Web application experience by clicking on a hypertext link provided in the application interface. This links to a URL for an activated survey campaign deployment.

With customization to add the valid URL of any activated survey deploymenht, any Oracle self-service Web application is integrated with the Scripting Engine Web interface.

Targeted Deployment Integration with Oracle Marketing and TCA

For execution of targeted survey deployments, HTML invitations and reminders are sent from Oracle One-to-One Fulfillment to list members of Oracle Marketing lists, inviting those members to participate in a survey or Web script by accessing a valid survey URL. The survey URL includes a JSP template parameter, defining how the script executes in a Web browser at runtime. It also identifies the deployment (dID), and each individual potential respondent (using the parameter rID).

Additional information is also appended to the survey URL, which makes that information available to the Scripting blackboard during execution of the script. Parameters passed include the following:

Parameter	Blackboard Key	Description
apps user ID	IES_FND_USER_NAME	User name for the current apps session. ■ If deployment is standalone, this is the ID of the apps guest user. ■ If deployment is menu-based, this is the ID of the logged-in apps user.

Parameter	Blackboard Key	Description
parent party ID	P_PARTY_ID	■ If contact type is an organization, this is the party ID of the organization. ■ If contact type is a person, this is the party ID of the person.
contact party ID	P_CONTACT_PARTY_ID	■ If contact type is an organization, this is the party ID of the contact person within the organization. ■ If contact type is a person, this value is null.

Guidelines

- These parameters are only passed for targeted deployments.
- The apps user ID identifies the apps-level guest user if the hosting option is standalone. If it is menu-based, this parameter holds the Oracle Applications user name of the person currently logged in when the script is executed.
- The parent party ID and the contact party ID are data items from the trading community architecture (TCA).
- The parent party ID and the contact party ID are passed from columns stored in the AMS_LIST_ENTRIES table within the Oracle Applications schema.
  - The parent party ID column in AMS_LIST_ENTRIES, PARENT_PARTY_ID, is stored in the blackboard as blackboard key P_PARTY_ID.
  - The contact party ID column in AMS_LIST_ENTRIES , PARTY_ID, is stored in the blackboard as blackboard key P_CONTACT_PARTY_ID.
- Because TCA accounts for contacts that are either the primary party or a contact within an organization, you must understand how TCA handles data differently for each situation. For example, P_PARTY_ID could either be the name of a person or of an organization. Accordingly, CONTACT_PARTY_ID will be null when both fields are not required (when the contact type is a person).
  - If the party type of a selected contact is an organization, then the P_PARTY_ID parameter holds the name of the organization. In this case, the name of the specific contact person within the organization is stored in the parameter P_CONTACT_PARTY_ID.

- If the party type of a selected contact is person, then the P_PARTY_ID parameter holds the name of the person. In this case, the parameter P_CONTACT_PARTY_ID is null.

See Also

- Scripting Engine Agent Interface

11.1.3 Scripting Administration Console

The Scripting Administration console is an HTML administration console built on the deprecated JTT technology stack. This console is accessed by logging into Oracle HTML-based applications using an Oracle Applications user account with the Scripting Administrator responsibility. Using the Scripting Administration console, script administrators can launch Script Author as a Java applet, administer deployed scripts and custom Java archive files, and view panel footprint reports.

There is no integration between this component and other Oracle Applications. Like other HTML applications, users with multiple responsibilities can access other application functionality by changing the current responsibility for the authenticated Oracle Applications session.

See Also

- Script Author
- Scripting Engine
- Survey Administration Console

11.1.4 Survey Administration Console

The Survey Administration console is an HTML administration console built on the Oracle Applications framework. Using the Survey Administration console, survey administrators create, administer, and deploy survey campaigns and monitor progress in existing survey campaign deployments.

The Survey Administration console integrates directly with Oracle Marketing and Oracle One-to-One Fulfillment by hosting tabs from those HTML-based applications in the administration console. These tabs are the Audience tab and the Invitations tab, respectively.

Same Restrictions Apply

The same business rules, prerequisites, and requirements for accessing the user interface of integrated applications also apply to the survey administrator.

For example, to access the Invitations tab, the user requires not only the Survey Administrator responsibility, but also must be assigned the JTF_FM_ADMIN_ROLE role. For more information, see Granting JTF Roles.

The invitations tab provides access from the survey administration console to the same application pages that you can access using the Template, Query, and Status subtabs in the Fulfillment Administration console. Note that a fulfillment administrator is still required to set up fulfillment servers, groups, and to monitor fulfillment requests.

In addition to survey-specific functionality, the survey administration console provides access to an Audience tab. This tab displays the same content that you can see in the Oracle Marketing user interface under the Audience tab (for Oracle Marketing releases prior to 11.5.10) or the Audience Workbench tab (for Oracle Marketing release 11.5.10 and later). To access this tab from the Survey Administration console, the Oracle Applications user account must have a sales group ID and a sales role. These are administrative setup steps required to be performed by a user with the CRM Resource Manager responsibility. You can also configure the Survey Administrator responsibility to bypass this requirement. For this procedure, see Bypassing the Sales Resource Group Membership Requirement.

See Also

- Script Author
- Scripting Engine
- Scripting Administration Console

11.2 Integration by Technology Stack

This section includes the following topics:

- HTML-Based Integrated Applications
- Forms-Based Integrated Applications

See Also

- Integration by Component
- Prebuilt Integration with Oracle Applications

11.2.1 HTML-Based Integrated Applications

This section includes the following topics:

- Oracle Self-Service Web Applications
- Oracle One-to-One Fulfillment
- Oracle Marketing

See Also

- Forms-Based Integrated Applications

11.2.1.1 Oracle Self-Service Web Applications

Using self-service Web-based applications such as Oracle iSupport, customers can launch a script in the Scripting Engine Web interface, and execute it using a Web browser.

> **Note:** While this section describes application functionality within context of Oracle iSupport, scripts in this interface can be hosted in any Oracle self-service Web application.

When Oracle Scripting is integrated with self-service Web applications such as Oracle iSupport, users log into the application using an Oracle Applications 11*i*-certified Web browser and a user with the appropriate responsibility.

Users then access a page in the UI with one or more hypertext links. These links are created through application customization, in which an application administrator previously customized the UI to include, within the code for each hypertext link, a valid survey URL for an active standard survey campaign deployment.

From this page, the application user clicks the hypertext link. The page refreshes, displaying the survey questionnaire’s first panel as an HTML page in the same Web browser instance. If the survey campaign details specify a survey resource header section or footer section, these appear on each HTML page of the script.

Each page (corresponding to a single script panel) contains one or more question controls, including at minimum a submit button (typically labeled Continue). Clicking the Continue button progresses the script to the next panel, which appears as a separate HTML page.

If an error results, or when the script is complete, the survey resources designated as the error page or final page display, as appropriate.

Hosting Options and Technology Stacks

There are two hosting options for survey deployments: Menu Based and Standalone.

For deployments designated as menu-based, the navigational menus (tabs) of the self-service Web application are retained at the top of each page throughout. These are also referred to as hosted surveys or Web scripts. In all cases, hosted deployments use the Oracle Applications authentication information for an existing session. After a hosted survey is completed, the user can continue to use the application from which a script is launched in the web browser.

For deployments set as standalone, the Oracle Applications session (always required in order to execute a script) is initiated using a guest user. The only privilege granted to the guest user is to execute the Web script or survey specified by the survey campaign. After the script completes its execution and the final page is displayed, the Oracle Applications session is de-authenticated.

For standard (non-list-based) Web scripts or surveys using the Oracle Applications framework, the session will always be hosted, if the survey URL is selected from an authenticated Oracle Applications session. This is because all Scripting Engine Web interface transactions using the Oracle Applications framework use the same JavaServer page, OA.JSP, as the template from which to execute all scripts. Upon completion of the script in the Web browser, the Oracle Applications session is still valid, and you can return to other functions in that application.

In contrast, targeted (list-based) Web scripts or surveys do not support the menu-based hosting option if using the Oracle Applications framework. Regardless of the hosting option selected, the applications guest user will initiate the Oracle Scripting session. Even if the user is logged into an Oracle self-service Web Application when a link to an active targeted survey URL is selected, the menus from any open self-service Web application will *not* be hosted in the Web browser at script runtime. Upon completion of the script, any active Oracle Applications session will expire. Thus, no additional work can be completed in the open application without logging back into the application.

If the Base Tech Stack setting for the survey campaign is Deprecated - JTT, you can control whether a script executed in a Web browser is hosted in the self-service Web application's UI (and the session remains open upon script completion), or whether guest user authentication is used to run the script, and the applications session is terminated upon script completion. This hosting option (and the Oracle Scripting session authentication method) is determined by the JSP template parameter used in the survey URL. If the survey URL includes IESSVYMENUBASED.JSP, then the application uses the authentication of the existing Oracle Applications session. If the survey URL includes IESSVYMAIN.JSP., then the guest user Oracle Applications account is used to authenticate a new session, and is restricted to executing a script only. Upon completion of the script in the Web browser, the Oracle Applications session is terminated.

When defining targeted survey campaign deployments, there is a hosting options list selection. For JTT survey campaigns, when the survey administrator selects Standalone, the JSP template parameter IESSVYMAIN.JSP is used, and when the administrator selects Menu Based, the IESSVYMENUBASED.JSP template is added to the survey URL.

As indicated above, for an OAF deployment, even if Standalone is selected, if the survey URL is clicked from a valid Oracle Applications session, it will be hosted in the application's user interface.

To force a script to execute using guest user authentication using the Oracle Applications framework (in other words, to force it to execute with a standalone hosting option), copy the survey URL and paste it into a separate Web browser session with no current authentication and no valid cookie to a current authenticated session. For example, if you are using Oracle Applications with Netscape Communicator in an application such as Oracle iSupport, copy the survey URL, paste it into the Address field of a different supported Web browser (such as Microsoft Internet Explorer). Then the session will use the seeded guest user account for authentication. At the end of any session using the guest user, the Oracle Applications session will be terminated.

For more information, see Survey URL.

For hosted surveys using the OA framework, you can redirect the user to any URL defined as a survey resource, upon script completion in the Web browser.

For hosted surveys using the JTT framework, you can designate the page listing survey links as the Final page survey resource, if you define it as a survey resource. Thus, after the Web script or survey is completed, the user is returned to the page from which they launched the survey script. Other links on that page can be customized as desired by Oracle iSupport administrators.

For hosted surveys using the JTT framework, you can use the test resource IESSVYMENUBASEDTESTERROR.JSP (seeded with Oracle Applications) as the basis for your error page. Some customization is required.

Guidelines

- From the user interface of Oracle iSupport or another hosted self-service Web application, the user clicks a hypertext link for an active survey URL. The survey or Web script starts in the current Web browser window, preserving the menu structure.

- The survey is expected to be completed in order to deliver reliable information. Abandoned survey responses are tracked but not displayed in most reports.
- Upon completion of the script, you can continue working in the hosted application, as the Oracle Applications session remains valid.

See Also

- Oracle One-to-One Fulfillment
- Oracle Marketing

11.2.1.2 Oracle One-to-One Fulfillment

Oracle One-to-One Fulfillment provides enterprises with an automated method for sending information to its customers. Interaction center agents use Oracle One-to-One Fulfillment directly or through automated applications such as Oracle Marketing. Agents often receive customer requests for information, literature, and other correspondence. These requests include product and service inquiries, pricing questions, billing inquiries, and general customer care issues. Agents can immediately fulfill these requests using e-mail to automatically deliver predefined fulfillment items to identified customers.

Oracle One-to-One Fulfillment is set up and maintained by a fulfillment administrator, who must configure and run a fulfillment server to monitor fulfillment requests, and a fulfillment e-mail server (a logical channel for the output device that delivers electronic documents to customers).

From an Oracle Scripting perspective, Oracle One-to-One Fulfillment can be used to deliver "invitation" and "reminder" master documents (HTML documents) using electronic mail, inviting identified parties to participate in a survey campaign by visiting a defined URL.

The parameters required for integration with Oracle One-to-One Fulfillment for inviting potential respondents via e-mail to participate in a survey are included in the deployment details section of the Survey Administration console.

The Survey Administration console also includes an Invitations tab, which hosts Oracle One-to-One Fulfillment functionality. Survey administrators that need to access this tab must be assigned the JTF_FM_ADMIN_ROLE role. For more information, see Granting JTF Roles.

References

- Granting JTF Roles
- Survey Administration Console

See Also

- Oracle Self-Service Web Applications
- Oracle Marketing

11.2.1.3 Oracle Marketing

Oracle Marketing is a tool used to create and manage marketing and sales campaigns, assign scripts to particular campaign schedules, assign agents to specific campaigns, etc. You can create lists of customers in Oracle Marketing, and invite them using e-mail (using Oracle One-to-One Fulfillment "invitation" and "reminder" master documents) to participate in a survey campaign by visiting a defined URL.

The parameters required for integration with Oracle Marketing to use existing marketing lists as the source for sending e-mail invitations and reminders are included in the deployment details section of the Survey Administration console.

The Survey Administration console also includes an Audience tab, which hosts Oracle Marketing functionality. The Oracle Applications user account for survey administrators that need to access this tab must have a sales group ID and a sales role. For more information, see Bypassing the Sales Resource Group Membership Requirement.

References

- Bypassing the Sales Resource Group Membership Requirement
- Survey Administration Console

See Also

- Oracle Self-Service Web Applications
- Oracle One-to-One Fulfillment

11.2.2 Forms-Based Integrated Applications

Oracle Forms-based eBusiness applications that integrate with Oracle Scripting include Oracle TeleSales, Oracle Collections, and Oracle TeleService. From an interaction in any of these business applications, an agent can launch a script in the

Scripting Engine agent interface. Each business application supports at least two models: (1) allowing the agent to select any script from a list of deployed scripts for that instance, and (2) specifying a particular script to launch upon a particular condition or action. No additional configuration is required to enable the first model. To enable launch of a particular script, specific setup steps are required, which vary by application. Each of these is discussed in the section for integrating the particular business application. For more information, see *Oracle Scripting Implementation Guide Release 11i* and the implementation guide for each specific application in question.

To execute scripts in the agent interface, the agent's Oracle Applications account must have the correct responsibility associated with it to launch the business application in question. The table below indicates, by application, responsibilities which may be required, and from which form in the application the script can be launched.

Business Application	Required Responsibility	Form
Oracle TeleSales	TeleSales Agent	eBusiness Center
Oracle Collections	Collections Agent	Collections
Oracle TeleService	Customer Support	Contact Center

Other responsibilities for a particular business application may also have access to execution of scripts. The ones listed in the table above are established for agent-level privileges. For more information, consult the product documentation for the appropriate application.

Each of these Oracle Forms-based business applications are typically used by interaction center agents for sales, collections, or service, respectively. The functionality of these business applications can be greatly extended and customized by creating and using scripts launched from the appropriate business application.

Guidelines

- After logging into the appropriate business application, no specific Scripting responsibilities are required for an agent to launch a script.
- These scripts are launched in the Scripting Engine agent interface in a separate window.
- If choosing from the full list of deployed scripts, the agent must select the appropriate script name and language from the Script Chooser list.

- The script is expected to be completed prior to returning to the business application.
- If the script commits changes to the customer information in the database, these changes *will not* be reflected in the business application until a fresh request for customer information from the database is made from the business application.
- Individual implementation, setup, or configuration steps may be required in order to access the business applications listed above. For example, membership in a sales or telesales group (for Oracle TeleSales) or a collections group (for Oracle Collections) is required. Other conditions may apply for each application. These steps are outside the scope of this document. For more information, refer to the appropriate documentation for the relevant business application.

References

For relevant information regarding Oracle Forms-based applications, refer to *Oracle Applications Developer's Guide* Release 11i, Volume 1. Specifically, refer to the section Menus and Function Security.

See Also

- HTML-Based Integrated Applications

11.3 Prebuilt Integration with Oracle Applications

Interaction centers using the Scripting Engine agent interface can not only interact with customers but can also integrate desktop workflow between various applications. Oracle Scripting is intended to be used in combination with integrated Oracle E-Business Suite applications (Oracle TeleService, Oracle TeleSales, and Oracle Collections) to take full advantage of the contact handling and service, sales, or collections features of those applications, respectively. In terms of prebuilt integration, scripts can be launched from any of these customer-facing, interaction-based business applications. Additionally, Oracle TeleSales and Oracle Collections automatically send values for a selected customer to the Scripting blackboard for use in a script interaction.

Oracle Marketing is seeded with several Oracle Scripting building block scripts. These enable users of Oracle Marketing to authenticate customers, register for an event, create leads, and so on.

This section includes the following topics:

- Integrating with Oracle TeleSales

- Integrating with Oracle Collections
- Integrating with Oracle TeleService
- Integrating with Oracle Marketing

See Also

- Integration by Component
- Integration by Technology Stack

11.3.1 Integrating with Oracle TeleSales

This section includes the following topics:

- Launching Any Deployed Script from Oracle TeleSales
- Launching a Specific Script from Oracle TeleSales
- Parameters Passed to the Scripting Blackboard

See Also

- Integrating with Oracle Collections
- Integrating with Oracle TeleService
- Integrating with Oracle Marketing

11.3.1.1 Launching Any Deployed Script from Oracle TeleSales

From Oracle TeleService, you can launch in the Scripting Engine agent interface any valid script deployed to your Oracle Applications instance.

Prerequisites

- At least one appropriate script must be deployed from Script Author to the applications database.
- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.
- Oracle TeleSales sends a set of values for the selected customer to the Scripting blackboard when a script is launched from the eBusiness Center. In order to receive those values and use them in a script, the agent must be in an active Oracle TeleSales transaction with a customer selected in the eBusiness Center.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

TeleSales Agent

Steps

1. From the Navigate To menu, select **All Scripts**.

   The Oracle Scripting window (commonly referred to as the Script Chooser) appears.

2. From the Script Chooser, select a script name and language.
3. Click **Start Scripting**.

   The script will launch in a separate window.

See Also

- Launching a Specific Script from Oracle TeleSales
- Parameters Passed to the Scripting Blackboard

11.3.1.2 Launching a Specific Script from Oracle TeleSales

From Oracle TeleSales, you can launch in the Scripting Engine agent interface a designated script deployed to your Oracle Applications instance, leveraging Oracle Marketing campaign information.

There are three models for launching this designated script:

- Automatic launch of a specific script when a call is received.
- Automatic launch of a specific script when an interaction is started in the eBusiness Center.
- Launch of a specific script based on explicit agent request (agent clicks **View Script** button).

Oracle Marketing Campaign Administration Requirements

To designate the specific script to launch from Oracle TeleSales using any of these models requires the script to be designated at the campaign schedule level. Thus, as a prerequisite of launching a designated script, you must administer campaigns in Oracle Marketing. Oracle Marketing requirements include the following:

- A valid campaign.
- A valid campaign schedule associated with the campaign.
- An inbound or outbound script associated with the campaign schedule.
- Agents must be associated to the campaign schedule, or campaign schedules must be assigned to agents.

For more information on these steps, see Oracle Marketing product documentation. For your convenience, these steps are also documented in the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing in *Oracle Scripting Implementation Guide*.

Obviously, there is no explicit action to be taken by the agent for automatic launch of a script. Based on the type of automatic launch (on call delivery or on start of interaction), when the event or condition is met, the script launches in a separate window. Thus, the procedure below applies to launching a specified script from the eBusiness Center at a time chosen by the agent.

Specific profile settings are required to accomplish each method of launching a specified script. These are described in the guidelines below.

Prerequisites

- At least one appropriate script must be deployed from Script Author to the applications database.
- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.
- A campaign must exist in Oracle Marketing with a valid campaign schedule, and the agent using the appropriate business application must be associated with the campaign schedule associated with the campaign.
- The specific deployed script must be associated to either the campaign schedule from Oracle Marketing in order to launch without accessing the Script Chooser.
- System profiles must be set according to the guidelines indicated.
- Oracle TeleSales sends a set of values for the selected customer to the Scripting blackboard when a script is launched from the eBusiness Center. In order to receive those values and use them in a script, the agent must be in an active Oracle TeleSales transaction with a customer selected in the eBusiness Center.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

TeleSales Agent

Steps

1. Ensure a customer is selected in the eBusiness Center.
2. Ensure a campaign is selected in the Overview tab.

   The selected campaign, or a campaign schedule associated with the campaign, must have a script associated with it.
3. From the Navigate To menu, select **View Script**, or from the Overview tab, click **View Script**.

   The script will launch in a separate window.

Guidelines

- As indicated above, campaign administration of Oracle Marketing is required.
- For *any* model of launching scripts from Oracle TeleSales or Oracle Collections, set the OTS: Scripting Installation profile to **Yes**.
- For any model of *automatically* launching a *specified* script, also set the OTS: Script Launch on Interaction profile to **Yes**.
- For Oracle TeleSales to launch a script upon delivery of the call to the agent desktop only, also set the OTS: Script Launch on UWQ Delivery profile to **Yes**.

Following is a table describing the effects of the OTS profile settings for Oracle Scripting:

Profile Option	Setting	Result
OTS: Script Launch on UWQ Delivery	Yes	Automatic launch of a specific script in a separate window when a call is received in Oracle TeleSales.
OTS: Script Launch on UWQ Delivery	No or Null	No script launches on call delivery.

Profile Option	Setting	Result
OTS: Script Launch on Interaction	Yes	When an Oracle TeleSales interaction starts, the specified script automatically launches in a separate window.
OTS: Script Launch on Interaction	No or Null	No script launches on start of interaction.
OTS: Scripting Installation	Yes	This setting allows scripts to be launched from Oracle TeleSales.
OTS: Scripting Installation	No	Regardless of other profile settings, if set to no, scripts will not launch from Oracle TeleSales or Oracle Collections.

Launching a Specified Script on Call Delivery to Agent Desktop

Profile Option	Setting
OTS: Script Launch on UWQ Delivery	Yes
OTS: Script Launch on Interaction	Yes
OTS: Scripting Installation	Yes

Launching a Specified Script Automatically on Start of Interaction

Profile Option	Setting
OTS: Script Launch on UWQ Delivery	No or null
OTS: Script Launch on Interaction	Yes
OTS: Scripting Installation	Yes

Launching a Specified Script When Agent Clicks View Script

Profile Option	Setting
OTS: Script Launch on UWQ Delivery	No or null
OTS: Script Launch on Interaction	No
OTS: Scripting Installation	Yes

References

- For information on associating a script with a campaign schedule, see the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing > Associating a Script to a Campaign in *Oracle Scripting Implementation Guide*.
- For information on associating a script with a campaign schedule, see the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing > Associating a Script to a Campaign Schedule in *Oracle Scripting Implementation Guide*.

See Also

- Launching Any Deployed Script from Oracle TeleSales
- Parameters Passed to the Scripting Blackboard

11.3.1.3 Parameters Passed to the Scripting Blackboard

Oracle TeleSales and Oracle Collections integrate with Oracle Scripting by passing parameters from a current Oracle TeleSales or Oracle Collections interaction to the Scripting Blackboard each time a script is launched from the appropriate business application. The parameters passed are listed in the table below, and mostly relate to the Trading Community Architecture.

Trading Community Architecture

The Oracle Trading Community Architecture (TCA) is a data model that provides a virtual representation of the community in which that business is conducted. This model includes parties and related party entities.

A party is defined as a person, organization, relationship, or collection of parties that can enter into business relationships with other parties. TCA is used by Oracle Applications to store sets of information in the database about our customers. If an individual is a customer, the party type is Person. If an organization or enterprise is a customer, then the party type is Organization. Relationships or groups further define specific individuals or groups of individuals as contacts within an organization. The TCA model is valuable because, with strategies to store and access information in specified tables and columns and business rules to govern these, the same database schema is used despite whether the business relationship is business-to-business (B2B) or business-to-consumer (B2C).

Listed alphabetically below are the parameters currently being sent by Oracle TeleSales and Oracle Collections to the Scripting blackboard:

Parameter	Description	Comments
ADDRESS_ID	Location ID	For PARTY_TYPE of RELATIONSHIP, Address ID is not related to a given PARTY_ID
AST_RESOURCE_ID	Resource ID	This resource ID is derived from the global variable AST_ RESOURCE_ID from Oracle Marketing.
CAMPAIGN_ID	Source Code ID	If a campaign is associated with a campaign schedule, this is the ID associated with the campaign source code. Otherwise, this value is null.
CAMPAIGN_NAME	Campaign name	If a campaign is associated with a campaign schedule, this is the name of the campaign. Otherwise, this value is null.
CAMPAIGN_SOURCE_ CODE	Source Code	If a campaign is associated with a campaign schedule, this is the source code associated with the campaign. Otherwise, this value is null. If you do not provide a source code in Oracle Marketing, a unique code is generated by the system.
CONTACT_ID	Person Party ID	Contact ID if the selected contact is a party type of person. Otherwise, this value is null.
CUST_ACCOUNT_ID	Account ID	The account ID for this party.
INTERACTION_ID	Interaction ID	The interaction ID from the eBusiness Contact Center for the interaction record of the given party type.
P_PARTY_ID	Party ID of the account owner.	For an individual, the Person ID of the selected party. For a relationship customer, the organization party ID will be passed.
PARM.PARTY_ID	Party ID of the account owner.	For an individual, the Person ID of the selected party. For a relationship customer, the organization party ID will be passed.
PARTY_CONTACT_ID	Organization Party ID	Contact ID if the selected contact is a party type of organization. Otherwise, this value is null.
PARTY_ID	Party ID of the account owner.	For an individual, the Person ID of the selected party. For a relationship customer, the organization party ID will be passed.
PARTY_TYPE	Party type	Identifies the party relationship: person, organization, or relationship.
TM_ANI	ANI	This parameter (automatic number identification or ANI) indicates the billing number from which an inbound call originates. This is sometimes referred to as the caller ID.

Parameter	Description	Comments
TM_DNIS	DNIS information	This parameter indicates the number from which an inbound call originates. The Dialed Number Identification Service (DNIS) parameter indicates which number for an inbound call was dialed by the caller. DNIS information is often used for call routing or to separate campaigns and functions.

Obsolete Parameters

Following are parameters no longer supported using this model:

Parameter	Description	Comments
COLLATERAL_ID	Obsolete	Not applicable in this release.
EVENT_ID	Obsolete	Not applicable in this release.
LEAD_ID	Obsolete	Not applicable in this release.
OPP_ID	Obsolete	Not applicable in this release.
P_RESOURCE_ID	Obsolete	Not applicable in this release.
SOURCE_FORM_ID	Obsolete	Not applicable in this release.
SOURCE_FORM_FUNCTION	Obsolete	Not applicable in this release.

Guidelines

- Values are passed each time a script is launched from the eBusiness Center or from Oracle Collections.
- Any parameters not available from the appropriate business application will be null in the Scripting Blackboard.
- A customer must be selected in Oracle TeleSales or Oracle Collections in order to receive some or all values into the script.
- If a script relies on values passed from Oracle TeleSales or Oracle Collections for critical processing, the script should include a test for a null value before attempting the process. If a null value is found, the script should include panels to collect the required information at runtime, populating them with the appropriate blackboard answer keys to ensure the script can continue its processing.

- If a blackboard value retrieved into a script is subsequently changed in the business application before the script is completed, the updated value *will not display in the script*. If the script is completed and started again, the updated value *will* be retrieved into the script.

See Also

- Launching Any Deployed Script from Oracle Collections
- Launching a Specific Script from Oracle Collections

11.3.2 Integrating with Oracle Collections

Oracle Collections shares some interaction center functionality with Oracle TeleSales, and as such relies on many Oracle TeleSales application settings and behavior. For example, the blackboard values sent to a script when a script is launched from Oracle TeleSales or Oracle Collections (see Parameters Passed to the Scripting Blackboard) are identical.

As another example, as indicated in Guidelines below, specific Oracle TeleSales profile settings are required in order to launch a designated script from Oracle Collections.

Additionally, collections agents must be assigned specific role types and roles, and must have group membership with specific role types, roles, and usages. These are administered in CRM Resource Manager. Oracle Collections requirements include:

- Collections agent must be imported as a CRM resource. This is administered in Oracle CRM Resource Manager.

Note: To import the agent as a resource, that agent must already exist as an employee in the database. Employees are created in Oracle HRMS Manager (if installed) or in Oracle CRM Resource Manager.

- The CRM resource (the agent) must have the following attributes:

Role Type	Role
Collections	Collections Agent
Telesales	Telesales Agent

- The agent must have membership in a resource group. For Oracle Collections, the resource group must have the following attributes:

Role Type	Role	Usage
Collections	Collections Agent	Collections
Telesales	Telesales Agent	Sales and Telesales

> **Note:** In order to use Oracle Collections, *you must ensure the resource and the resource group have attributes set for both Oracle Collections and for Oracle TeleSales, ev*en if the agent is only expected to use Oracle Collections. For more information, *see Oracle Collections Implementation Guide release 11i.*

This section includes the following topics:

- Launching Any Deployed Script from Oracle Collections
- Launching a Specific Script from Oracle Collections
- Parameters Passed to the Scripting Blackboard

See Also

- Integrating with Oracle TeleSales
- Integrating with Oracle TeleService
- Integrating with Oracle Marketing

11.3.2.1 Launching Any Deployed Script from Oracle Collections

Using Oracle Collections, you can launch a script of your choice in the Scripting Engine agent interface from the full set of scripts deployed to your Oracle Applications instance.

Prerequisites

- At least one appropriate script must be deployed from Script Author to the applications database.
- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- Oracle TeleSales and Oracle Collections sends a set of values for the selected customer to the Scripting blackboard when a script is launched from the appropriate business application. In order to receive those values and use them in a script, the agent must be in an active Oracle TeleSales or Oracle Collections transaction with a customer selected in the appropriate business application.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Collections Agent

Steps

1. From the Collections window, navigate to the Profile tab.
2. In the Code field, enter a valid code associated with an active campaign schedule.

   When a valid code is entered, the Name and Type fields populate with appropriate information, and the View Script button is enabled.
3. From the Profile tab, click **View Script**.

   The Oracle Scripting window (commonly referred to as the Script Chooser) appears.
4. From the Script Chooser, select a script name and language.
5. Click **Start Scripting**.

   The script will launch in a separate window.

See Also

- Launching a Specific Script from Oracle Collections
- Parameters Passed to the Scripting Blackboard

11.3.2.2 Launching a Specific Script from Oracle Collections

Using Oracle Collections, you can launch a designated script in the Scripting Engine agent interface. There are two models for launching this designated script:

- Automatic launch of a specific script when an Oracle Collections interaction is started.
- Launch of a specific script upon agent entry of a specified code and the application's subsequent successful validation of that code.

Oracle Marketing Campaign Administration Requirements

To designate the specific script to launch from Oracle Collections using either of these models requires the script to be designated at the campaign schedule level. Thus, as a prerequisites of launching a designated script, you must administer campaigns in Oracle Marketing. Oracle Marketing requirements include:

- a valid campaign, *including the designation of a campaign source code*
- a valid campaign schedule associated with the campaign
- an inbound or outbound script associated with the campaign schedule
- agents associated to either the campaign or to the campaign schedule

For more information on these steps, see Oracle Marketing product documentation. For your convenience, these steps are also documented in the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing in *Oracle Scripting Implementation Guide*.

Starting a Collections Interaction

There are several ways to start an interaction in Oracle Collections. At any time, an interaction can be started by the agent by clicking the Start Interaction icon in the toolbar. Also, by default, Oracle Collections starts an interaction when a customer is selected in the Collections window. Like most Oracle applications, Oracle Collections can be modified to start an interaction after a customer is selected and any other event on the form is triggered (such as selecting a specific field).

Regardless of how an Oracle Collections interaction is started, there is no explicit action to be taken by the agent in order to cause the automatic launch of a script, if Collections is configured to do so upon start of an interaction. Thus, the procedure below applies to an agent causing the launch of a specified script from Oracle Collections by entering a code in the Code field within the Profile tab. The Code field associates with a campaign, campaign schedule, event, or promotion code, as established in Oracle Marketing.

Prerequisites

- At least one appropriate script must be deployed from Script Author to the applications database.
- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.
- As prerequisites of Oracle TeleSales and Oracle Collections, a campaign must exist in Oracle Marketing, and the agent using the appropriate business application must be associated with the campaign.
- The specific deployed script must already be associated to either the campaign or campaign schedule from Oracle Marketing in order to launch without accessing the Script Chooser.
- System profiles must be set according to the guidelines indicated below.
- Oracle Collections send a set of values for the selected customer to the Scripting blackboard when a script is launched from the Collections window. In order to receive those values and use them in a script, the agent must be in an active Oracle Collections transaction with a customer selected.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Collections Agent

Steps

1. From the Navigator, select Collections.

   The Collections window appears.

2. In the Profile tab, type the appropriate value in the Code field and press the tab key.

   Oracle Collections will automatically validate this code against campaign information established in Oracle Marketing.

3. If the code is valid, the Name and Type fields will populate with information from Oracle Marketing, and the specified script will launch in a separate window.

   If the code is not valid, no script will launch.

Note: The Name and Type fields cannot be used to launch a script. Only by typing the appropriate code in the Code field can agents launch the specified script automatically. If desired, an agent can elect to click the View Script button and manually select a valid script from the list of available scripts in the Script Chooser.

Guidelines

For Collections to launch a script upon delivery of the call to the agent desktop, the following Oracle TeleSales profiles must be set:

Profile Option	Setting
OTS: Script Launch on Interaction	Yes
OTS: Scripting Installation	Yes

References

- For information on associating a script with a campaign schedule, see the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing > Associating a Script to a Campaign Schedule in *Oracle Scripting Implementation Guide*.
- For information on associating a script with a campaign schedule, see the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing > Associating a Script to a Campaign in *Oracle Scripting Implementation Guide*.

See Also

- Launching Any Deployed Script from Oracle Collections
- Launching a Specific Script from Oracle Collections
- Parameters Passed to the Scripting Blackboard

11.3.3 Integrating with Oracle TeleService

This section includes the following topics:

- Launching Any Deployed Script from Oracle TeleService
- Launching a Specific Script from Oracle TeleService

- Obtaining Values in a Script from Oracle TeleService
- Creating Service Requests in Oracle TeleService from Oracle Scripting

See Also

- Integrating with Oracle TeleSales
- Integrating with Oracle Collections
- Integrating with Oracle Marketing

11.3.3.1 Launching Any Deployed Script from Oracle TeleService

Using Oracle TeleService, you can launch a script of your choice in the Scripting Engine agent interface from the full set of scripts deployed to your Oracle Applications instance.

Prerequisites

- At least one appropriate script must be deployed from Script Author to the applications database.
- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.
- System profiles must be set according to the guidelines indicated below.
- If the script being called uses any Forms commands from any TeleService form, that form must be open and appropriately populated with information in the Contact Center for those values to be made accessible to the script.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Customer Support

Steps

1. Ensure a customer is selected in the Contact Center.
2. From the Tools menu, select **Script**, or click on the Script icon in the toolbar.

The Oracle Scripting window (commonly referred to as the Script Chooser) appears.

3. From the Script Chooser, select a script name and language.
4. Click **Start Scripting**.

   The script will launch in a separate window.

Guidelines

- For Oracle TeleService to launch a script upon delivery of the call to the agent desktop, the following profile must be set:

Profile Option	**Setting**
Customer Care: Automatically Launch a Script for Identified Callers	<global script name>

> **Note:** In some cases, this profile option display value may be Customer Care: Default Contact Center Script.

See Also

- Launching a Specific Script from Oracle TeleService
- Obtaining Values in a Script from Oracle TeleService
- Creating Service Requests in Oracle TeleService from Oracle Scripting

11.3.3.2 Launching a Specific Script from Oracle TeleService

Using Oracle TeleService in a media-enabled environment, you can automatically launch a designated script in the Scripting Engine agent interface. As soon as a call is delivered to the Oracle TeleService Contact Center, if at least one of three parties is identified, the designated script launches in a separate window. In the event that a party is not identified (the TeleService screen pop is blank), the script is not launched. Only one of three possible parties (organization, person, or relationship) is required to launch the script, as identified by the HZ_PARTIES table in the Oracle Applications schema.

There is no explicit action to be taken by the agent for automatic launch of a script other than appropriate setup, as indicated in the guidelines below.

Prerequisites

- Oracle TeleService must be media-enabled.
- In order to launch a designated script automatically, the calling party (rel party) must be identified.
- At least one appropriate script must be deployed from Script Author to the applications database.
- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.
- System profiles must be set according to the guidelines indicated below.
- If the script being called uses any Forms commands from any TeleService form, that form must be open and appropriately populated with information in the Contact Center for those values to be made accessible to the script.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

Customer Support

Steps

1. From the Navigator, select Contact Center and click **Open**.

   The Contact Center window opens.

2. When a call is received, the designated script launches in a separate window.
3. Complete interaction within the script before returning to the Contact Center.

Guidelines

- For Oracle TeleService to launch a script upon delivery of the call to the agent desktop, the following profile must be set:

Profile Option	Setting
Customer Care: Default Contact Center Script	<global script name>

See Also

- Launching Any Deployed Script from Oracle TeleService
- Obtaining Values in a Script from Oracle TeleService
- Creating Service Requests in Oracle TeleService from Oracle Scripting

11.3.3.3 Obtaining Values in a Script from Oracle TeleService

In addition to being able to launch a script from Oracle TeleService, Oracle Scripting can retrieve values from the TeleService Contact Center using Forms commands in Script Author. Upon execution of the Forms command at script runtime, the Contact Center retrieves the appropriate contact information, and passes this information on to Scripting.

Some typical Forms commands used to pass values from Oracle TeleService to Oracle Scripting are listed in the table below:

Forms Command	Value Passed to Script
GetInteractionId	Interaction ID
GetCustomerId	Account ID/Party ID
GetCustAccountId	Customer Account ID
GetCustomerType	Customer Type
GetContactId	Contact ID
GetContactPointId	Contact Point ID
GetContactPointType	Contact Point Type
GetContactType	Contact Type
GetPrimaryFlag	Primary Flag

References

- For more information on Forms commands, see the Forms Commands section of the *Oracle Scripting Developer's Guide*.
- A script can also be associated with a Relationship Plan. For information on relationship plans, refer to *Oracle Customer Care Implementation Guide release 11i*.

See Also

- Launching Any Deployed Script from Oracle TeleService

- Launching a Specific Script from Oracle TeleService
- Creating Service Requests in Oracle TeleService from Oracle Scripting

11.3.3.4 Creating Service Requests in Oracle TeleService from Oracle Scripting

Oracle Scripting integrates with Oracle Forms-based applications. To create a service request by calling a PL/SQL package from Scripting, the following fields must be populated:

- SR Type
- SR Status
- SR Severity
- Summary
- Group Owner (see guidelines)
- Individual Owner (see guidelines)

Guidelines

- If set to "Yes", the Oracle TeleService profile option, Service: Group Owner Mandatory, requires the user to enter a group owner when creating a service request.
- If set to "Yes", the Oracle TeleService profile option, Service: Default Group Owner for Service Requests, requires either the group owner *or* an individual owner to be entered when creating a service request. This profile takes precedence over the Service: Group Owner Mandatory profile option, if set.
- The usage of a CRM resource must be defined as "Support" for the resource to be available in the Service Request tab in Oracle TeleService. This usage is set when the resource is defined using Oracle CRM Resource Manager.

See Also

- Launching Any Deployed Script from Oracle TeleService
- Launching a Specific Script from Oracle TeleService
- Obtaining Values in a Script from Oracle TeleService

11.3.4 Integrating with Oracle Marketing

As of Oracle Scripting release 11.5.9, Oracle Marketing is seeded with several Oracle Scripting building block scripts. These robust seeded scripts are supported and maintained by the Oracle Marketing team, and are documented in *Oracle Marketing Implementation Guide*, available on Oracle*MetaLink* (part number B10588-02).

These new building block scripts expand the offering previously available to include scripts for business-to-consumer (B2C) functions in addition to the business-to-business (B2B) functions provided by the Oracle Scripting building block scripts. The Oracle Marketing scripts also take advantage of changes to the APIs used to create leads and events that have been introduced in Oracle Marketing release 11.5.9. The API changes included in the new scripts will not be reflected in the original set of Oracle Scripting building block scripts.

Additionally, they replace completely the lead creation and event registration functionality previously accessible using the Oracle Scripting Building Block scripts. For more information, see Lead Creation Building Block and Seeded Command Obsolete.

Building Block Script	Function
AMSCAUTH.SCR	Customer Authentication
AMSERB2B.SCR	B2B Event registration (supersedes Oracle Scripting building block script IESEVENT.SCR)
AMSERB2C.SCR	B2C Event registration
AMSERWCA.SCR	Event registration with customer authentication
AMSLDB2C.SCR	B2C Lead creation
AMSLDB2B.SCR	B2B Lead creation (replaces Oracle Scripting building block script IESLEAD.SCR)
AMSLDWCA.SCR	Lead creation with customer authentication
AMSDATACQ.SCR	Data acquisition

These scripts are seeded with Oracle Marketing in current releases, and can be downloaded from the applications server from the product top directory ($AMS_TOP/scripts). Oracle Corporation suggests that users first transfer the files to a local directory, then launch Script Author, open each relevant building block script, modify as required, and deploy to the specific instance.

The Java code needed by the Marketing scripts is also delivered as part of the Marketing 11.5.9 release, so developers do not need to create any custom Java in order to use the Java methods.

See Also

- Integrating with Oracle TeleSales
- Integrating with Oracle Collections
- Integrating with Oracle TeleService

A

Upgrading from Releases Prior to 11.5.6

Oracle Scripting release 11.5.6 was a major functional release including significant improvements and changes. Some of the new features or aspects of Oracle Scripting may impact production environments using previous releases of Oracle Scripting 11*i* and custom scripts. This section is provided to guide you through the changes from previous releases to release 11.5.6 or later with minimum impact.

Upgrading Oracle Scripting as of release 11.5.6 or later can be accomplished using the latest IES rolled-up patch (RUP). Alerts indicating the latest Oracle Scripting RUP are available on Oracle*MetaLink* or Oracle iSupport. For latest release information, alerts, and to obtain patches, always consult Oracle*MetaLink* or Oracle iSupport first. If you continue to have questions or concerns, contact your Oracle Support representative.

The steps required to migrate any Oracle Scripting 11*i* environment from a pre-11.5.6 release to release 11.5.6 include:

- Section A.1, "Implementation Upgrade Tasks"
- Section A.2, "Installation of Script Author Post-11.5.6 Stand-Alone Application"
- Section A.3, "Converting, Re-Deploying, and Testing Scripts"

A.1 Implementation Upgrade Tasks

Implementation upgrade tasks include:

- Section A.1.1, "Upgrading to Oracle Applications 11.5.6 or Later"
- Section A.1.2, "Modifying System Profile Settings"
- Section A.1.3, "Referencing Custom Java"
- Section A.1.4, "Modifying Custom Java"

- Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File"

See Also

- Section A.2, "Installation of Script Author Post-11.5.6 Stand-Alone Application"
- Section A.3, "Converting, Re-Deploying, and Testing Scripts"

A.1.1 Upgrading to Oracle Applications 11.5.6 or Later

In order to use Oracle Scripting 11.5.6 or later, you must upgrade Oracle Applications to release 11.5.6 or later. This is accomplished either by executing a Rapid Install for 11.5.6 or later or by upgrading with the appropriate Interaction Center Family Pack for 11.5.6 or later.

> **Note:** Oracle Applications must generally be at the same patch level (for example, release 11.5.6) to function appropriately. Upgrading to release 11.5.6 can be achieved using the appropriate series of patches that meets your specific application requirements, or using the appropriate Oracle Platinum version to support a release 11.5.6 upgrade. Appropriate patching is prerequisite of a successful Oracle Scripting implementation and is not addressed herein. For more information, see Oracle*MetaLink* or Oracle iSupport, or contact your Oracle Support representative.

See Also

- Section A.1.2, "Modifying System Profile Settings"
- Section A.1.3, "Referencing Custom Java"
- Section A.1.4, "Modifying Custom Java"
- Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File"

A.1.2 Modifying System Profile Settings

To execute Oracle Scripting in the Scripting Engine agent interface in release 11.5.6 or later, you must change the IES : Architecture Type profile upon upgrading to release 11.5.6 or later. The profile options for this system profile setting have changed, as has the recommended architecture.

Guidelines

1. For upgrading to the Apache Mid-Tier architecture, select the value **Apache Mid Tier/Servlet Architecture**.
2. For using Oracle Scripting 11.5.6 or later in the Caching Architecture, select the value **Caching Architecture** (replacing **Two Tier Mode**).

   Note that using Oracle Scripting in the Caching Architecture will *only be supported* for customers previously using this architecture in a production environment, and only for a limited amount of time, as designated by Oracle Support Services.
3. The previous value, **Three Tier Mode**, has been obsoleted. New implementations will not include this lookup value. *Do not use this setting under any circumstances.*

See Also

- Section A.1.1, "Upgrading to Oracle Applications 11.5.6 or Later"
- Section A.1.3, "Referencing Custom Java"
- Section A.1.4, "Modifying Custom Java"
- Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File"

A.1.3 Referencing Custom Java

There are two types of custom Java used in association with Oracle Scripting:

- Custom Commands
- Custom Java Beans

See Also

- Section A.1.1, "Upgrading to Oracle Applications 11.5.6 or Later"
- Section A.1.2, "Modifying System Profile Settings"
- Section A.1.4, "Modifying Custom Java"
- Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File"

A.1.3.1 Custom Commands

The most common type is custom Java methods written using Oracle Scripting APIs to provide additional functionality to a script using a Command (for example, branch actions, embedded values in panel text, statements defining parameters for a conditional or indeterminate branch, and so forth).

These Java commands always run in the Java Virtual Machine (JVM) designated by architecture type. Thus:

- If using the Caching Architecture, the JVM is provided by JInitiator on the client workstation. In this case, your code must be compiled using JDK 1.1.8.

  If using the Caching Architecture, see Guidelines above for caveats.
- If using the Apache Mid-Tier architecture, the JVM is provided by the Apache JServ (which can be JDK 1.1.8, 1.2, or 1.3). The compiling restrictions are based on the JDK used by your version of Apache.

This section includes the following topics:

- Section A.1.3.1.1, "Referencing Custom Commands in the Caching Architecture"
- Section A.1.3.1.2, "Referencing Custom Commands in the Apache Mid-Tier Architecture"

See Also

Section A.1.3.2, "Custom Java Beans"

A.1.3.1.1 Referencing Custom Commands in the Caching Architecture Using the Caching Architecture of Oracle Scripting in releases 11.5.4 and 11.5.5, the APPSWEB.CFG file was required to be modified:

- **a.** To reference custom Java in support of custom commands,
- **b.** To obtain JDBC classes, and
- **c.** To obtain AOL/J classes.

If you are upgrading an Oracle Scripting implementation using the Caching Architecture, you must still modify the APPSWEB.CFG file to reference custom Java code (as well as to download JDBC classes to the client workstation). *Modification of the APPSWEB.CFG file to obtain AOL/J classes is no longer required. For details, see* Appendix D of the *Oracle Scripting Implementation Guide* release 11.5.4 or 11.5.5, as appropriate.

Note: If using the Caching Architecture, see Guidelines in this section for caveats.

See Also

Section A.1.3.1.2, "Referencing Custom Commands in the Apache Mid-Tier Architecture"

A.1.3.1.2 Referencing Custom Commands in the Apache Mid-Tier Architecture

Using the new Apache Mid-Tier Architecture, the class path of the JSERV.PROPERTIES file on the Apache Web server must be modified to reference custom Java. The requirement to upload custom Java classes to the applications server has not changed.

References

See Also

Section A.1.3.1.1, "Referencing Custom Commands in the Caching Architecture"

Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File"

A.1.3.2 Custom Java Beans

The second type of custom Java to support Scripting is custom java code compiled in a UI Java bean that is used to replace an entire panel in the runtime session. Custom Java beans are executed by Oracle JInitiator on the agent client. Thus, this code must be compiled using the version of JDK that is compatible with Oracle JInitiator on the agent client (the same version or lower). This may include JDK 1.3 or 1.1.8. For specific JDK or other software certification information, reference the Certify section of Oracle*MetaLink* or Oracle iSupport.

Note: While substitution of *an entire panel* with a Java bean is still supported, substituting a *question* in a panel with a Java bean is no longer supported as of release 11.5.6 and later.

See Also

Section A.1.3.1, "Custom Commands"

A.1.4 Modifying Custom Java

Moving from the Caching Architecture to the Apache mid-tier architecture changes the location of error messages. Since Oracle JInitiator is no longer providing the JVM on the client, some messages that were available in the Java console are now written to the JServ error logs.

In any custom Java code where you included print line statements (generally used for debugging or verifying that a certain step in a method was executed), you will need to change your code to continue viewing messages.

Change any instance of "System.out.println" to "System.err.println" to allow you to view the messages in JServ error logs.

Problem:

Prior to upgrading to the Apache mid-tier architecture, certain debug statements or messages appeared in the Java console that are no longer visible.

Cause:

The JVM executing code is located on the Apache web server whereas before the messages were "printed" to the client's Java console.

Solution:

To view debugging messages in the Apache Mid-Tier architecture, change any Java code using "System.out.println" to "System.err.println". This will result in the messages being written to the jserv error logs.

See Also

- Section A.1.1, "Upgrading to Oracle Applications 11.5.6 or Later"
- Section A.1.2, "Modifying System Profile Settings"
- Section A.1.3, "Referencing Custom Java"
- Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File"

A.1.5 Adding Custom Java Class Path to JSERV.PROPERTIES File

This step is only relevant for users of the Apache Mid-Tier architecture between releases 11.5.6 and 11.5.8 base release (prior to Interaction Center Family Pack P). This step is not required for users upgrading to Oracle Scripting release 11.5.8 patched to Interaction Center Family Pack P or later.

Any time a Script Author script references custom Java *(regardless of whether that script will be executed in the agent interface or in a Web browser as a survey)*, you must have an appropriately configured JAR file available to the class loader, containing compiled Java classes referenced in the script. In the Apache Mid-Tier architecture for Oracle Scripting, prior to Interaction Center Family Pack P, that custom Java needs to be stored on the applications server, and referenced in the Apache JServ classpath in order for the Scripting Engine to be able to call those classes at runtime. Following are steps to add a custom JAR file to the Apache Web server's JServ configuration file, known as the JSERV.PROPERTIES file.

> **Note:** There is more than one way to specify a class path for Apache. *If the method described below is not supported by your enterprise implementation,* consult with your Apache Web server administrator to determine the appropriate method for your environment. This may include identifying the classpath in a control file or another configuration.

Login and Responsibility

To access the JSERV.PROPERTIES file you must have physical or Telnet access to the Web server, and have applmgr or sysadmin privileges. Use the appropriate login for these privileges in your environment.

Prerequisites

- A custom Java method must be written and tested.
- All custom Java methods must be compiled into class files. Oracle Corporation recommends compiling custom Java code for use with Oracle Scripting with JDK 1.1.8. For more information, see Compiling Custom Code Using JDK 1.1.8.
- All class files must be packaged into a Java archive format appropriately.
- All JAR files must be uploaded to the applications server in a specified directory. Any directory accessible to the Web server is acceptable. For ease of maintainability, Oracle Corporation recommends creating a directory named ies_custom on the JAVA_TOP (Oracle file system on the applications server) specifically for storing custom JAR files for Oracle Scripting.

Steps

1. Connect using Telnet to the enterprise system.
2. Change to the directory where your Oracle HTTP Server is installed. Refer to the Appendix of your Oracle Applications Installation manual if you are not sure where to find this.
3. Change to the directory in which the JSERV.PROPERTIES file resides.

> **Note:** The actual name of this file may include an underscore and the Apache JServ port (for example, JSERV.PROPERTIES_9404). For the purposes of this document, the Apache Web server configuration file will be referred to generically as the JSERV.PROPERTIES file. If you have any difficulties locating the file, consult with your Apache Web server administrator.

The location of this file is environmentally dependent. It *may* be in one of the following paths:

<ORAHTTP_TOP>/Jserv/etc/jserv.properties

<ORAHTTP_TOP>/apache39/conf

The ORAHTTP_TOP variable may not be set for your environment, in which case the path will be fully qualified to identify the entire physical path in the file system. If your configuration is different, consult with your Apache Web administrator to locate this directory.

4. Load the JSERV.PROPERTIES file into a text editor.
5. Look in the JSERV.PROPERTIES file for several lines beginning with **wrapper.classpath=**.

   This is the section in the file that sets the CLASSPATH used by the web server.
6. If this configuration file contains a wrapper.classpath entry, you can modify it to add the classpath of any custom JAR file your script references in this line. For example:

```
wrapper.classpath=/d01/testora/iAS/Apache/xsu/xsu12.jar
```

> **Note:** If there is no listing, consult with your Apache Web server administrator to determine the appropriate location to specify a classpath for your Apache environment in support of Oracle Applications.

7. Save the JSERV.PROPERTIES file, and exit the text editor.

   Your web server is now configured to load custom Java classes for Oracle Scripting.

Guidelines

- Back up your JSERV.PROPERTIES file prior to customization. After configuring and testing, also back up your customized JSERV.PROPERTIES file in a separate location.

> **Note:** When you apply a patch, any customizations you have made to the JSERV.PROPERTIES file *may be lost*. In this case you will need to restore customizations to this file after patching.

- This process will have to be repeated any time you reference new classes in Oracle Scripting, regardless of whether the script is executed by agents in the Java UI or by survey respondents in a Web browser.

> **Note:** If you write new Java methods supporting scripts already in production, you may wish to combine the compiled code in an existing JAR file (and re-uploading) so you need not update the JSERV.PROPERTIES file. Oracle Corporation recommends backing up custom JAR files prior to changes to any system in production, and storing the backups in a separate location.

See Also

- Section A.1.1, "Upgrading to Oracle Applications 11.5.6 or Later"
- Section A.1.2, "Modifying System Profile Settings"
- Section A.1.3, "Referencing Custom Java"
- Section A.1.4, "Modifying Custom Java"

A.2 Installation of Script Author Post-11.5.6 Stand-Alone Application

Implementations of Oracle Scripting between release 11.5.6 and baseline release 11.5.8 *prior to Interaction Center Family Pack P* must install the Script Author standalone application to create, modify, or deploy scripts to the applications database.

As of release 11.5.8 (Interaction Center Family Pack P and later), Script Author is a Java applet launched from the Scripting Administration console. The standalone application is no longer available, and users are encouraged to use the Java applet to develop scripts. Such users will enjoy new capabilities available from the

Scripting Administration console, but only for scripts deployed from the applet. For example, custom Java archives in support of scripts can be loaded from the Scripting Administration console and can be mapped from this console to apply to any script or scripts deployed from the console only.

Script Compatibility

Any existing (pre-11.5.6) script *must be converted* to an 11.5.6 (or subsequent) version of Script Author prior to execution in the Scripting Engine provided by Oracle Applications 11.5.6 or later.

The method for obtaining and implementing the Script Author standalone application for Oracle Scripting releases 11.5.6 through baseline 11.5.8 prior to Interaction Center Family Pack P is described below.

This section includes the following topics:

- Section A.2.1, "Locating Script Author Installation Package"
- Section A.2.2, "Downloading Script Author Installation Package"
- Section A.2.3, "Preparing for Script Author Installation"
- Section A.2.4, "Installing Script Author 11.5.6 Through 11.5.8 Baseline Release"

See Also

- Section A.1, "Implementation Upgrade Tasks"
- Section A.3, "Converting, Re-Deploying, and Testing Scripts"

A.2.1 Locating Script Author Installation Package

Installation of Script Author as a standalone Java application is accomplished using the Oracle Universal Installer. As of release 11.5.6 and through Interaction Center Family Pack O, this is obtained as a ZIP file from the APPL_TOP (Oracle Applications file system), specifically in the product top for Oracle Scripting (product code IES).

This is in contrast to earlier releases of Oracle Applications 11*i*, in which the only way to access the Script Author installation package was on the 11*i* Call Center Applications Install CD, or as a ZIP file downloaded from Oracle*MetaLink*.

As of Interaction Center Family Pack P and later or release 11.5.9 or later, Script Author is only available as a Java applet. No installation package is provided for these releases.

See Also

- Section A.2.2, "Downloading Script Author Installation Package"
- Section A.2.3, "Preparing for Script Author Installation"
- Section A.2.4, "Installing Script Author 11.5.6 Through 11.5.8 Baseline Release"

A.2.2 Downloading Script Author Installation Package

Download the iesauthr.zip file from the Oracle Applications 11.5.6 or later APPL_TOP.

Prerequisites

- You must be using an Oracle Applications 11*i*-compliant Web browser.
- You must know the appropriate URL to access your Oracle Applications environment.
- You must have enough space allocated on the target computer to download and extract the Script Author files. The compressed file is approximately 80 megabytes in size. This size may vary on your local machine.

Login

None required

Responsibility

None required

Steps

1. From your Web browser, access the appropriate URL for your environment, using the following syntax, and press Enter.

   ```
   http://<server>.<domain>:<port>/OA_HTML/download/ies/iesauthr.zip.
   ```

   Fill in the server, domain, and port specific to your environment. For example:

   ```
   http://server1.yourcompany.com:7777/OA_HTML/download/ies/iesauthr.zip
   ```

2. Your Web browser will prompt you to perform an action with the specified file. Select the option that allows you to save the file to your local file system.

   a. If you are downloading the files using the target computer, navigate to the desired directory and save the file.

After installation, you do not need to retain the installation package, since it will be accessible from the same URL at any time. Therefore, you may save the installation package in any location, including your default TEMP directory.

b. If you are downloading the files for installation on a different computer, save the file to any location, and transfer the ZIP file to the target computer.

See Also

- Section A.2.1, "Locating Script Author Installation Package"
- Section A.2.3, "Preparing for Script Author Installation"
- Section A.2.4, "Installing Script Author 11.5.6 Through 11.5.8 Baseline Release"

A.2.3 Preparing for Script Author Installation

This task describes how to prepare the target computer for installation of Script Author 11.5.6 or later.

Login

None required

Responsibility

None required

Prerequisites

- You must have downloaded Script Author 11.5.6 or later from the APPL_TOP of your local environment.
- If required, you must have moved the Script Author installation package to the computer on which Script Author will be installed. Script Author can be installed on a computer running any Microsoft Windows 95, 98, 2000, or NT operating system.
- You must have enough space allocated on the target computer to accommodate the extracted Script Author files. Once extracted, Script Author requires approximately 100 megabytes. This size may vary on your local machine.

Steps

1. Locate the Script Author installation package, typically named iesauthr.zip.

2. In the target directory, extract the ZIP file to a directory with the same name.

   This will result in a directory named Iesauthr.

3. Review the readme.txt file located in the extracted path Iesauthr\Oracle Scripting Author Installation\readme.txt prior to installing.

See Also

- Section A.2.1, "Locating Script Author Installation Package"
- Section A.2.2, "Downloading Script Author Installation Package"
- Section A.2.4, "Installing Script Author 11.5.6 Through 11.5.8 Baseline Release"

A.2.4 Installing Script Author 11.5.6 Through 11.5.8 Baseline Release

This task describes how to prepare the target computer for installation of the Script Author standalone application for releases 11.5.6 through 11.5.8 MiniPack O. This information is not relevant for releases of Oracle Scripting after release 11.5.8 (Interaction Center Family Pack P and later).

Login

None required

Responsibility

None required

Prerequisites

- You must have downloaded Script Author 11.5.6 or later from the APPL_TOP of your local environment and extracted the Script Author installation package.
- You must install Script Author on a computer with a Microsoft Windows 95, 98, 2000, or NT operating system. Typically, this must be a networked computer that can see the Applications server on the same side of the firewall.
- You must have read the `readme.txt` file included in the installation package.
- You must have enough space allocated on the target computer to accommodate the installed Script Author files. Once extracted, Script Author requires only approximately 22 megabytes. This size may vary on your local machine.

Steps

1. Start the Oracle Universal Installer by running setup.exe from path:

   ```
   Iesauthr\Oracle Scripting Author Installation\install\win32\setup.exe
   ```

   Oracle Universal Installer launches and the Welcome screen appears.

2. Click **Next** to continue.

3. In the File Locations screen, specify the destination directory for Oracle Scripting Script Author.

   - The Source area displays the path to the components available for installation.

     Note: Do not modify the information in the Source area.

   - The Destination area displays the destination alias, and the default path to which Script Author will be installed.

     Note: The Script Author can be installed in any directory; installation into an Oracle Home directory is *not required*. If the target computer has an Oracle Home directory, this location is recommended and will typically default as the Destination. You may choose another destination if desired.

     In the Destination area, type the destination path or click **Browse** to choose a destination path. If the destination path does not have a name, type an alias for the destination path in the Name field. You can provide any alias for the purpose of installing Oracle Scripting Script Author.

     Click **Next** to continue.

4. In the Installation Types screen, select an installation type and then click **Next** to continue.

   Note: Each installation type installs the same components.

5. In the Summary screen, verify the installation settings.

   - To make changes, click **Previous** to return to the appropriate screen.

- To being the installation, click **Install**.

  The Install screen appears, and Oracle Scripting Script Author is installed in the oracle\apps\ies\author directory of the destination path. The Oracle Universal Installer also creates an Oracle Scripting folder in the Programs menu.

  When the installation is complete, the End of Installation screen appears.

6. Click **Exit** to close the Oracle Universal Installer.

This completes the installation of Scripting Author. After installing, you may delete the Script Author installation package if desired.

See Also

- Section A.2.1, "Locating Script Author Installation Package"
- Section A.2.2, "Downloading Script Author Installation Package"
- Section A.2.3, "Preparing for Script Author Installation"

A.3 Converting, Re-Deploying, and Testing Scripts

When accessing the Script Chooser Forms window to launch a script in the Scripting Engine agent interface, *all scripts deployed to the database* will be listed. Nonetheless, ***only scripts that have been recompiled in Script Author 11.5.6 or later will be executable in the 11.5.6 or later Scripting Engine***.

This is true regardless of whether the scripts are launched from an integrated application (Oracle TeleSales, Oracle TeleService, or Oracle Collections), or when launching in standalone mode.

Thus, any previously deployed scripts required to be executed in Scripting Engine release 11.5.6 or later must be opened in a matching 11.5.6 or later Script Author, converted (by saving), re-deployed, and tested.

Note: Ensure that all scripts to be converted are still supported. For example, there are two qualifications:

While substitution of *an entire panel* with a Java bean is still supported, substituting a *question* in a panel with a Java bean is no longer supported as of release 11.5.6 and later.

Scripts that contain multiple answers in a panel may not use the "button" UI type. The only supported use of a button UI type is for a panel with a single answer definition.

Steps

The following is a list of the recommended steps for conversion and redeployment:

1. Determine which scripts in the production environment must be converted (i.e., which scripts will be run in the 11.5.6 or later environment).
2. Prepare a test environment that has the 11.5.6 or later release of Scripting (including the 11.5.6 or later version of Script Author) installed.
3. For each script identified in Step 1, locate in your script archive the source .SCRIPT or .SCR file that was used to deploy the script to the production environment.

   Note: If you are unable to locate the exact .SCRIPT or .SCR file that was used to deploy the script currently in use in the production environment, you can obtain a copy of this script from the production database itself. To do so, from Script Author, select the Database tab of the **Open Script** dialog to open scripts directly from the referenced database.

4. Make a backup copy of the script.

Note: If you select Save from Script Author, the pre-11.5.6 script will be saved in the same location, appended with a file extension of .PREVIOUS. Thus, if the previous script were named SAMPLE.SCRIPT, the 11.5.6 or later version of the script will now be named SAMPLE.SCRIPT, and the earlier version will be retained as SAMPLE.SCRIPT.PREVIOUS.

Warning: If you use the Save As command, ensure you provide a new name for the converted script, since the <filename>.SCRIPT.PREVIOUS copy will not be created.

5. Launch the release 11.5.6 or later version of Script Author.
6. Open the script file (identified in Step 3 above) in Script Author.
7. The Script Author will warn you that continuing will modify the script. Once you do so, you cannot use this script with Script Author releases previous to 11.5.6.
8. Save the converted script.
9. Choose **Tools > Deploy Script** to re-deploy the script to the test environment.

   You are now ready to execute this script in the test environment.
10. Run the script in the Scripting Engine agent interface and test the re-deployed script.
11. After verifying that no new problems have been introduced, follow the same process to re-deploy the script to the production environment.

    If you discover any issues, correct the script, redeploy, and test until successful.
12. Repeat this process for each script identified in Step 1.

Guidelines

To ensure no confusion between source script files, consider creating separate folders or directories on the Script Author workstation, identifying scripts by Script Author version (at minimum, discriminating between pre-11.5.6 scripts and 11.5.6 or later scripts). Move any <filename>.previous copies of updated scripts into the appropriate pre-11.5.6 directory.

See Also

- Section A.1, "Implementation Upgrade Tasks"
- Section A.2, "Installation of Script Author Post-11.5.6 Stand-Alone Application"

B
Oracle Scripting Profile Options

This chapter describes profile option settings that are required for successful implementation.

This section includes the following topics:

- Setting Profile Options
- Finding Responsibility ID Values
- Oracle CRM Technology Foundation (JTT) Profile Options
- Current IES Profile Options
- Obsolete IES Profile Options
- Information, Error and Debug Logging Profile Options
- Additional Required Profile Options
- Reserved IES System Profile Options

B.1 Setting Profile Options

Use the following procedure to set any profile option.

Prerequisites

Before making Oracle Forms settings, ensure that all Oracle Applications server processes are up and running. In particular, if you stopped concurrent managers before applying Oracle Applications patchsets, restart them now by changing to `$COMMON_TOP/admin/scripts`, and executing adcmctl.sh <APPS user name/APPS password> start.

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

System Administrator

Steps

1. From the Navigator, navigate to **Profile > System**.

   The Find System Profile Values window opens.

2. Check the level(s) at which you want to set the profile option. The available levels are listed below:

   - **Site**: If you select this level, the profile will apply to the entire site. Individual profiles with a different value designated at any other level will take precedence.
   - **Application**: If you select this level, choose the application from the Application LOV for which you want to set the profile option.
   - **Responsibility**: If you select this level, choose the responsibility from the Responsibility LOV for which you want to set the profile option.
   - **User**: If you select this level, choose the user from the User LOV for whom you want to set the profile option.

3. In the Profile field, enter the profile name, such as IES: Architecture Type, or a wildcard search criterion such as IES%.

4. Click **Find.**

   The System Profile Values form opens with the results of your search.

5. Verify or set the profile option(s) at the levels that you selected.

See Also

- Finding Responsibility ID Values
- Oracle CRM Technology Foundation (JTT) Profile Options
- Current IES Profile Options
- Obsolete IES Profile Options
- Information, Error and Debug Logging Profile Options

- Additional Required Profile Options
- Reserved IES System Profile Options

B.2 Finding Responsibility ID Values

You must know the APPLICATION_ID value for Oracle Scripting and the RESPONSIBILITY_ID value for the Survey Administrator responsibility before you can set the required Oracle CRM Technology Foundation (JTT) profile options for users with these responsibilities.

> **Note:** The APPLICATION_ID value for the seeded Scripting User (or Scripting Agent) responsibility is 519. The RESPONSIBILITY_ID value for Survey Administrator is 21685. However, in some scenarios, the responsibility identification may differ, based on environmental factors.

Use the following procedure to find the APPLICATION_ID value and RESPONSIBILITY_ID value of a responsibility.

Prerequisites

None

Login

Log into Oracle applications using the Personal Home Page login, or the Single Sign-On login if implemented.

Responsibility

System Administrator

Steps

1. From the Navigator, navigate to **Security > Responsibility > Define**.

   The Responsibilities form opens.

2. Choose **View > Find.** Search for the responsibility, highlight it, and click **OK** in the search window.

   The Responsibilities form is populated with the record for the responsibility that you chose.

3. With your cursor in any field of the record, choose **Help > Diagnostics > Examine.**

   The Examine Field and Variable Values window opens.

4. In the Examine Field and Variable Values window, choose **APPLICATION_ID** from the Field LOV.

   The Value field in the Examine Field and Variable Values window is populated with the value of APPLICATION_ID.

5. In the Examine Field and Variable Values window, choose **RESPONSIBILITY_ID** from the Field LOV.

   The Value field in the Examine Field and Variable Values window is populated with the value of RESPONSIBILITY_ID.

See Also

- Setting Profile Options
- Oracle CRM Technology Foundation (JTT) Profile Options
- Current IES Profile Options
- Obsolete IES Profile Options
- Information, Error and Debug Logging Profile Options
- Additional Required Profile Options
- Reserved IES System Profile Options

B.3 Oracle CRM Technology Foundation (JTT) Profile Options

This section summarizes the Oracle CRM Technology Foundation (JTT) profile options that you need to set when implementing Oracle Scripting. Most of these are set at the site level.

Listed in the table below are the system profile name, a description of the function or purpose of the profile setting, recommended level settings, and a sample value.

Profile Option Name	Description	Recommended Level Setting	Value[1]
JTF_PROFILE_DEFAULT_APPLICATION	Default application ID (519=Oracle Scripting)	Site	519

Profile Option Name	Description	Recommended Level Setting	Value[1]
JTF_PROFILE_DEFAULT_BLANK_ROWS	Number of blank rows on HTML forms (can be set to any integer > 0)	Site	3
JTF_PROFILE_DEFAULT_CSS	Default Oracle CRM Technology Foundation Cascading Style Sheet	Site	jtfucss.css
JTF_PROFILE_DEFAULT_CURRENCY	Default currency. Enter the currency code in all uppercase letters.	Site	USD
JTF_PROFILE_DEFAULT_NUM_ROWS	Number of default rows (records) displayed on HTML summary tables (can be set to any integer > 0)	Site	10
JTF_PROFILE_DEFAULT_RESPONSIBILITY (application level only)	Default responsibility ID (21685=Survey Administrator)	Application	21685

[1] See Section B.2, "Finding Responsibility ID Values" if you need to find the APPLICATION_ID and RESPONSIBILITY_ID values for a responsibility.

See Also

- Setting Profile Options
- Finding Responsibility ID Values
- Current IES Profile Options
- Obsolete IES Profile Options
- Information, Error and Debug Logging Profile Options
- Additional Required Profile Options
- Reserved IES System Profile Options

B.4 Current IES Profile Options

This section summarizes the current Oracle Scripting (IES) profile options that you need to set when implementing the Scripting Engine and the Survey components of Oracle Scripting. Set the profile options in this section at the site level.

Listed in the table below are the system profile name, any seeded lookup values, a description of the function or purpose of the profile setting, and recommended settings.

Profile Option Name	Lookup Values	Description	Recommended Setting
IES : Architecture Type	Apache Mid Tier/Servlet Architecture Three Tier Mode Caching Architecture	Determines mode of Oracle Scripting operations. 1. Use Apache Mid Tier for all new implementations. 2. Never set to Three Tier Mode. 3. Use Caching Architecture only if upgrading existing caching architecture implementations.	Apache Mid Tier/Servlet Architecture
IES : Debug Mode	Debug off Debug on	Controls display at runtime of *client-side* informational and error messages for Scripting Engine operations. When enabled, these messages are displayed in a Java-based Debug console that appears (in a separate window) when a script is executed in the Scripting Engine agent interface. *Server-side* message logs are controlled by FND profiles. See Information, Error and Debug Logging Profile Options for more information.	Generally set to Debug off unless troubleshooting specific issues. When troubleshooting specific issues, enable at the Responsibility or User level for best results.
IES : Display Suspend Button on Script Frame	False True	When set to **True**, the Scripting Engine agent interface displays a Suspend button on the bottom of the script frame. This button is hidden if the profile is set to **False** (or remains null). When this button is clicked by an agent during script runtime, the current script interaction is suspended. (Requires Suspendable global script property option to be enabled.)	False unless using Oracle Scripting in standalone mode

Profile Option Name	Lookup Values	Description	Recommended Setting
IES : Dropdowns Display Default Null Choice in HTML UI	True False	By default, Oracle Scripting displays a text string "- select one - " at runtime for questions with a data type of drop-down list. Although "- select one -" appears in the list, this is a null value. Business rules for a drop-down list require the user to select one value for this question, resulting in a blackboard value other than null for each such question. Setting this profile to **False** suppresses the null value with the "- select one - " one string in the Scripting Engine web interface. For Web browsers only, the first answer choice in the list will appear as the selected value, unless another answer choice has been designated as a default answer.	Set to **False** to prevent " - select one -" string from displaying for scripts displaying in HTML UI. Leave null or set to **True** to ensure this string remains in both interfaces. When null or set to True, user action will be required at runtime to select an option from the list, as a null value is not acceptable for the drop-down list question UI type.
IES : Initial Script Frame Size	1280 X 1024 800 X 600 Maximized 1024 X 768	Controls the initial size of the Scripting Engine window frame, in pixels, in the Forms/Java agent user interface.	Prior to the introduction of this profile option, the initial size of the Oracle Scripting window in the agent interface was 800 X 600 pixels by default. Change as required to accommodate various monitor resolutions.
IES : Scripting Panel Display Mode	Display Multiple Panels at a time in Scripting window Display Single Panel at a time in Scripting window	Determines in Scripting Engine Java agent interface whether a single panel is visible upon script execution or whether multiple panels are displayed with active panel having focus.	At the discretion of the implementing enterprise.

Profile Option Name	Lookup Values	Description	Recommended Setting
IES: Survey Resources Caching	On Off	Determines whether survey resources for OAF surveys are cached at runtime. If no value is set, caching occurs by default (same behavior as setting to On).	Set to Off to avoid caching. Set to On or leave null to cache survey resources for OAF surveys.
IES: Proxy Server Name	None	Name of Proxy server (required if proxy server used for Scripting Engine agent interface only).	Syntax: http://<servername>.<domain>:<Apache Web server port>/<servlet_zone>
IES: Proxy Server Port	None	Port of Proxy server (required if proxy server used for Scripting Engine agent interface only).	N/A

See Also

- Setting Profile Options
- Finding Responsibility ID Values
- Oracle CRM Technology Foundation (JTT) Profile Options
- Obsolete IES Profile Options
- Information, Error and Debug Logging Profile Options
- Additional Required Profile Options
- Reserved IES System Profile Options

B.5 Obsolete IES Profile Options

This section summarizes obsolete Oracle Scripting (IES) profile options. *These options should not be used for any reason for Oracle Scripting release 11.5.8 or later operations.* Providing settings in some of these options could disable Scripting operations in the enterprise.

Profile Option Name	Lookup Values	Description	Recommended Setting
IES : JNDI name used for Scripting application	None	OBSOLETE	NONE
IES : Port for Oracle Scripting application	None	OBSOLETE	NONE
IES : SID of Oracle Scripting Database	None	OBSOLETE	NONE
IES : Scripting Bean Display Mode	Display Scripting bean in separate Frame Embed Scripting bean in form	OBSOLETE	NONE
IES : Server Host Name for Scripting application	None	OBSOLETE	NONE
IES: Agent password	None	OBSOLETE	NONE
IES: Agent user name	None	OBSOLETE	NONE
IES_SVY_OA_TEMP	None	OBSOLETE	NONE
IES : SVY ERROR PAGE	None	OBSOLETE	NONE
IES : SVY HEADER PAGE	None	OBSOLETE	NONE
IES: SVY FINAL PAGE	None	OBSOLETE	NONE
IES: TIME FORMAT	None	OBSOLETE	NONE

See Also

- Setting Profile Options
- Finding Responsibility ID Values
- Oracle CRM Technology Foundation (JTT) Profile Options
- Current IES Profile Options
- Information, Error and Debug Logging Profile Options
- Additional Required Profile Options
- Reserved IES System Profile Options

B.6 Information, Error and Debug Logging Profile Options

This section summarizes the system profile options that affect the logging of server-side informational, error, and debug messages in the Oracle Applications Logging Framework for OAF technology stack applications.

Client-side message logs are controlled by the IES : Debug Mode profile. See Current IES Profile Options for more information.

Listed in the table below are the system profile name, any seeded lookup values, the level available for setting the system profile, a description of the function or purpose of the profile setting, and recommended settings.

Profile Option Name	Lookup Values	Level	Description	Recommended Setting
FND: Diagnostics	Yes No	Site Application Responsibility User	Causes a Diagnostics link to appear in the HTML UI when set to **Yes**. This provides access to the log messages captured using the Oracle Applications Logging Framework.	Generally set to **No** unless specific user has need to access server logs. Generally recommended only for administrators or trusted users. For this reason, recommended to set at User level.
FND: Debug Log Enabled	Yes No	Site Application Responsibility User	Controls display at runtime of informational and error messages for Scripting Engine operations when set to **Yes**. These messages are logged to the Oracle Applications Logging Framework server logs.	Generally set to **No** unless troubleshooting specific issues. Recommended to set at User level to avoid heavy system load.
FND: Debug Log Filename	Null	Site Application Responsibility User	When defined, saves log messages (at the level determined by the FND: Debug Log Level profile setting) in a text file. Leave this value null to record log messages to database log files.	Generally, do not provide a filename unless directed to do so by Oracle Support Services for specific problem remediation. File size can quickly grow to an unmanageable size that may cause performance issues.

Profile Option Name	Lookup Values	Level	Description	Recommended Setting
FND: Debug Log Level	Statement Procedure Event Exception Error Unexpected	Site Application Responsibility User	Controls the level and degree of log messages captured, decreasing the amount of information by the order listed (Statement level logging captures the most information; Unexpected captures the fewest messages).	Generally, set this profile option to **Error** at the Application level for application **Scripting** to capture error-level message logs. For diagnosing specific errors, set this profile to **Statement** at the User level for the Oracle Applications user account experiencing problems. Use in combination with FND: Debug Log Module for specific problem diagnosis.
FND: Debug Log Module	None	Site Application Responsibility User	Determines the module for which messages are logged.	Generally, set to **ies%** at the Site level to log messages coming from all IES (Oracle Scripting) Java classes. To diagnose specific problems, set at the user level with more specific criteria. For example, to debug issues with the Scripting Engine, set to ies.engine%. This value is not case-sensitive. Ensure the % is included, to capture all messages included in the designated class.
Guest User Password or GUEST_USER_PWD	None	Site	Determines the Oracle Applications user account and password considered as the applications-level guest user.	Only required when logging for guest user must be enabled. Generally set to GUEST/<password>.

See Also

- Setting Profile Options
- Finding Responsibility ID Values
- Oracle CRM Technology Foundation (JTT) Profile Options

- Current IES Profile Options
- Obsolete IES Profile Options
- Additional Required Profile Options
- Reserved IES System Profile Options

B.7 Additional Required Profile Options

This section summarizes additional profile options that you need to set when implementing the Scripting Engine and the Survey components of Oracle Scripting. Set the profile options in this section at the site level.

Listed in the table below are the system profile name, any seeded lookup values, the level available for setting the system profile, a description of the function or purpose of the profile setting, and recommended settings.

Profile Option Name	Lookup Values	Level	Description	Recommended Setting
Apps Servlet Agent	None	Site Application	Identifies servlet zone of Apache Web server JServ used for Scripting Engine agent interface only. May be set at site level if one Apache server listener used. If different settings required for multiple applications, set at application level.	Syntax: http://<server name>.<domain>:<Apache Web server port>/<servlet_zone>
ICX: Language	None	Site	Establishes default language for the Internet Cartridge Exchange (ICX) settings for Web-based transactions	As appropriate.

See Also

- Setting Profile Options
- Finding Responsibility ID Values
- Oracle CRM Technology Foundation (JTT) Profile Options
- Current IES Profile Options
- Obsolete IES Profile Options

- Information, Error and Debug Logging Profile Options
- Reserved IES System Profile Options

B.8 Reserved IES System Profile Options

This section summarizes reserved Oracle Scripting (IES) system profile options. *These profile options should not be used at this time.* The profile options were either created for testing purposes only, or added by applications other than Oracle Scripting and are neither authorized for use nor maintained or supported. Profile options in this latter category generally do not provide any function whatsoever and may create problems if used.

Profile Option Name	Lookup Values	Description	Recommended Setting
IES : AUTHOR DEBUG MODE	Debug off Debug on	RESERVED.	NONE.
IES: - Assign Unserviced CPs	No Yes	RESERVED	NONE
IES: Access Code Update Type	Alphanumeric Numeric	RESERVED	NONE
IES: AutoAssign Contacts	No Yes	RESERVED	NONE
IES: Change Default Order Type	(Longlist)	RESERVED	NONE
IES: Default Template Type	Full Standard	RESERVED	NONE
IES: Manual Customer Product Definition Allowed	No Yes	RESERVED	NONE
IES: Survey Maintenance Page	None	RESERVED	NONE
IES: Termination Effect	Inactivate Remove	RESERVED	NONE
IES: Version Update Order Type	(Longlist)	RESERVED	NONE
IES:Organization Id	None	RESERVED	NONE

See Also

- Setting Profile Options
- Finding Responsibility ID Values
- Oracle CRM Technology Foundation (JTT) Profile Options
- Current IES Profile Options
- Obsolete IES Profile Options
- Information, Error and Debug Logging Profile Options
- Additional Required Profile Options

C

Oracle Discoverer Workbooks

Analysis and reporting for data received from survey campaigns has been enhanced. In previous releases of Oracle Scripting, three survey-related reports were accessible from the Survey Administration console. Beginning with Oracle Scripting release 11.5.9 (Interaction Center Family Pack Q or later), Oracle Corporation provides six pre-defined sets of reporting criteria to provide analysis information for the effectiveness of Oracle Scripting scripts executed as survey campaigns. No longer accessible from the Survey Administration console, these expanded survey analysis reports are now available as Oracle Discoverer workbooks, available as part of Oracle Interaction Center Intelligence reporting functionality.

Oracle Discoverer is a tool which enables report builders and analysts to create, modify, and execute ad hoc queries and reports. Casual users can view and navigate through pre-defined reports and graphs. Oracle Discoverer provides a business view (referred to as the end user layer or EUL) to hide the complexity of the underlying data structure. It enables users to focus on solving business problems instead of data access issues.

When scripts are executed as survey campaigns, return information for each deployment is stored in transaction tables in the IES schema of the Oracle Applications database. These transaction tables provide reporting data for three of the six survey analysis workbooks. Executing the Summarize Survey Data Concurrent Program moves relevant information into tables that summarize survey data from individual responses. These summary tables provide reporting data for the remaining three survey analysis workbooks. A matrix containing a list of the workbooks, a description of each, and the tables or views accessed by each workbook is provided below in Appendix C.2, "Survey Workbook Descriptions".

To view survey data using Oracle Discoverer workbooks, Oracle Applications users who are assigned a particular responsibility (based on the implementation) log into Oracle Discoverer User Edition and select the appropriate workbook from the database. The Oracle Applications user and responsibility must be granted administrative privileges to the end user layer as part of implementation. If you do not have access to the Oracle Scripting survey analysis workbooks, or do not have the appropriate privileges to access the workbooks, consult your Oracle Discoverer administrator, or perform the implementation steps described in this document.

Within the Survey Analysis business area are six pre-defined Oracle Scripting survey summary workbooks. Workbooks are lists of records, similar in design to a spreadsheet, that gather requirements for a Structured Query Language (SQL) database query. The columns in a workbook use ordinary business terms, shielding end users from requiring knowledge of the underlying relational database structure. When the query associated with each workbook is executed, users are prompted for parameters, after which the workbook builds and executes SQL statements which provide the requested information, which can range from simple to complex. The return data generated from the workbooks can be exported to a spreadsheet, allowing managers to create additional ad hoc reports on any data items available within a survey.

Using Oracle Discoverer, you therefore generate survey reports derived from survey data stored in one of two locations: in IES transaction tables within the Oracle Applications database schema, or summarized and compiled for analysis in survey summary tables following execution of the appropriate concurrent program.

Oracle Discoverer administrators can tailor existing workbooks to refine reporting information, or create new workbooks, using the Discoverer Plus or Discoverer Desktop components.

Oracle Discoverer employs an end user layer (EUL). An EUL is a metadata repository that allows users to view and select records from complex database structures without requiring knowledge of those structures or knowledge of Structured Query Language (SQL). The EUL consists of one or more business areas. Each business area contains folders (equivalent to tables and views) and each folder contains items (equivalent to columns in the table or view). Business areas are visible through Oracle Discoverer Administration Edition.

The EUL also contains workbooks. Workbooks are visible through Oracle Discoverer User Edition. Workbooks contain the specific data needed for various analysis tasks. Workbooks belong to business areas. Similar in appearance to a spreadsheet, a workbook references data in tables and views, and may have a query associated with it.

For the purpose of Oracle Scripting survey analysis, six workbooks are provided to generate specific reporting data on various survey-related activities. Each has a query associated with it which, when appropriately executed, provides an ad-hoc report on the appropriate parameters.

As part of post-installation steps for an Oracle Applications implementation, two EULs are created. Oracle Scripting reports are accessed through the Oracle Business Intelligence System (BIS) EUL.

This implementation guide contains the required steps to implement Oracle Discoverer workbooks to generate survey analysis reports for Oracle Scripting.

Use of Oracle Discoverer workbooks for survey reporting requires upgrade or installation of Oracle Applications 11.5.9 or later. Manual post-installation steps are required to import the survey reports into the EUL. This is accomplished using a shell script or by importing the workbooks into the Oracle Discoverer 4i Admin Edition. If you have not already imported these workbooks as part of the post-installation steps for Oracle Applications 11.5.9, you can do so by following the detailed steps provided in this appendix. To import the workbooks using the shell script or for more information on post-installation steps for Oracle Applications release 11.5.9, refer to Note.227071.1 on Oracle*MetaLink*.

> **Note:** For more information, see Oracle Discoverer product documentation or notes on Oracle*MetaLink*.

This section includes the following topics:

- Appendix C.1, "Implementing Survey Analysis Discoverer Workbooks"
- Appendix C.2, "Survey Workbook Descriptions"

C.1 Implementing Survey Analysis Discoverer Workbooks

To implement Oracle Discoverer workbooks for Oracle Scripting survey reporting, the following tasks are required:

Perform the following to implement Oracle Scripting survey reporting using Oracle Discoverer workbooks:

- Stage loader files (EEX files that provide the Oracle Scripting survey analysis business area, including workbooks) in preparation for import.
- Verify the accessibility of the BIS EUL to your Oracle Discoverer Administration Edition user. If BIS EUL is not accessible to your user, create an EUL.

- Import loader files (EEX files that provide business area and workbooks) into the EUL.
- Assigning User and Responsibility Privileges to the EUL.

EULs are created by Oracle Discoverer administrators. These users have access to Oracle Discoverer Administrative Edition with database administrator (DBA) privileges. These users must also have access to Oracle Discoverer User Edition (client/server version).

The EUL stores metadata for Oracle Discoverer. Oracle Scripting administrators that want to view the Oracle Discoverer reports must access the BIS EUL. If the BIS EUL has not been created, perform the procedure Creating an End User Layer. If this EUL exists and you have access to it from Oracle Discoverer, you may omit this procedure from your implementation steps.

Prerequisites

Prerequisites for implementing Oracle Discoverer reporting for survey analysis include the following:

- Oracle Scripting Release 11.5.9 or later
- Oracle Applications patched to release 11.5.9 or later, equivalent to Interaction Center Family Pack Q or later
- Windows NT server with Oracle Discoverer 4i Administration Edition

  Oracle Discoverer 4i Administration Edition and the corresponding adupdeul utility to import EEX files run only on Windows NT. If using Oracle Discoverer workbooks for reporting on Oracle Scripting survey activities, you must have at least one NT server with Discoverer Admin Edition installed. This may be a separate server, providing that it can communicate with the applications database.
- Appropriate patching of Oracle Discoverer, including Oracle Discoverer Patch 1834171

Tasks

This section includes the following tasks:

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"

- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

References

- Oracle Discoverer Plus User's Guide Oracle Discoverer 4.1
- Oracle Discoverer 4i Plus User's Guide Oracle Discoverer 4.1
- Oracle Discoverer Administration Edition Administration Guide Oracle Discoverer 4.1

See Also

Appendix C.2, "Survey Workbook Descriptions"

C.1.1 Staging Oracle Discoverer EEX Files for Import

Oracle Discoverer workbooks are created in Discoverer User Edition and can be saved as Discoverer export (EEX) files. The Discoverer files that provide Oracle Scripting survey workbook reports must be imported by an Oracle Discoverer Administrator before they are accessible to Discoverer users.

After installation of Oracle Applications release 11.5.9 or Interaction Center Family Pack Q, these files are installed on the appl_top in the following directory in your Oracle Applications server file structure:

```
$AU_TOP/discover
```

In order for these files to be imported during implementation, you must either map a network drive on the administration client to the appl_top where the EEX files are installed, or you copy the EEX files to the Discoverer administration server.

The appropriate EEX files to import include the following:

- bixiescmpgnrsflo.eex
- bixieslsthdrsflo.eex
- bixiesqstnsflo.eex
- bixiessrvysvflo.eex

- bixsvyqsfrwbo.eex
- bixiescrsstbrwbo.eex
- bixieslstsmmrflo.eex
- bixiesrmndrsvflo.eex
- bixiessvylstnflo.eex
- bixsvyqstxwbo.eex
- bixiescyclsvflo.eex
- bixiesmslstntflo.eex
- bixiesrspntrsflo.eex
- bixsvycmpswbo.eex
- bixiesdplcrssflo.eex
- bixiesnswrsflo.eex
- bixiessmmrystflo.eex
- bixsvydpldwbo.eex
- bixiesdplymntflo.eex
- bixiesqsdtvflo.eex
- bixiessrvynlybao.eex
- bixsvyqsdtwbo.eex

See Also

- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.2 Creating an End User Layer

Perform the following to create an EUL for Oracle Discoverer. This is not required if you have access to the BIS EUL.

Prerequisites

You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.

Responsibility

Oracle RDBMS DBA privileges

Steps

1. Log into Oracle Discoverer User Edition as an Oracle RDBMS user with database administrator (DBA) privileges.

   For example:

   > **Username:** sysadmin:System Administrator (case-sensitive)
   >
   > **Password:** sysadmin
   >
   > **Connect:** mydatabaseinstance

2. Click **Connect**.

   Upon connection, if no EUL exists accessible to your user account, the following message appears:

   > Oracle Discoverer Administration Edition:
   >
   > You do not have access to any End User Layers (EUL). You must create at least one EUL to which you have access to use Discoverer. Do you want to create an EUL now?

   Click **Yes**.

   The EUL Manager Wizard appears.

3. In the first window of the EUL Manager, click **Create an EUL...**

   The Create EUL Wizard appears.

4. Complete the EUL Wizard options as follows:

   In the Who will own the new EUL area:

- Select the **Select an existing user** option.

  Since you are using the same user, you are not prompted for a password.
- Check **Grant access to PUBLIC**.
- Check **New EUL is for use by Oracle Applications users ONLY**.

> **Note:** Selection of this last criteria is essential for application security and to ensure supportability.

In the EUL connection information area:

- In the User field, type the name of the user.
- In the Password field, type the user's password.
- In the Confirm Password field, retype the user's password.

Click **Next**.

The Create EUL Wizard Step 2 window appears.

5. Complete the EUL Wizard Step 2 options as follows:

   - From the Schema list, select the APPS schema.
   - In the Password field, type the apps level password.
   - Click **Next**.

     The Create EUL Wizard Step 3 window appears.

6. Complete the EUL Wizard Step 3 options as follows:

   - From the default tablespace list, select an appropriate tablespace into which EUL data will be inserted.
   - From the temporary tablespace list, select an appropriate tablespace into which data is temporarily stored.

   > **Caution:** Choosing the wrong tablespace can greatly impact performance. If you don’t know which tablespace is appropriate for the default or temporary tablespace, consult your database administrator.

   - Click **Finish**.

A series of progress indicators appears.

7. An Oracle Discoverer Administration Edition message appears, indicating that the EUL has been created successfully. Click **OK**.
8. Optionally, if you want to install tutorial information, click **Yes**. Otherwise, click **No**.
9. An Oracle Discoverer Administration Edition message appears, asking whether you want to connect as the owner of the new EUL that you have just created.

   Click **No** if you want to exit Oracle Discoverer Administration Edition.

   To proceed with the next procedure:

   - Click **Yes.**

     The Load Wizard appears.

   - Click **Cancel** to close the load wizard window, and proceed to step 2 of Section C.1.3, "Assigning End User Layer Implementation Privileges".

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.3 Assigning End User Layer Implementation Privileges

This procedure is only required if you have created a new EUL. Perform this procedure to assign administrative privileges for this EUL to a specific Oracle Applications user and responsibility (required in order to complete the implementation).

Prerequisites

- You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.
- You must be the owner of the EUL.
- You must have created or imported an Oracle Discoverer EUL.

Responsibility

Oracle RDBMS DBA privileges

Steps

1. Log into Oracle Discoverer Administrator Edition as an Oracle RDBMS user with database administrator (DBA) privileges. This must be the same user that created or imported this EUL.
2. From the Tools menu, select **Privileges**.

   The Privileges window appears with the Privileges tab selected.
3. From the user and responsibility drop-down list, select the appropriate user for whom you want to assign administrative access to this EUL.
   - Ensure the User box is checked.
   - Optionally, you can clear the Responsibility box to reduce the amount of records appearing in this list.

     For implementations with access to the seeded SYSADMIN Oracle Applications user account, selecting SYSADMIN is appropriate. Otherwise, ensure the user you select has the appropriate responsibility assigned.
4. From the Privileges list, select all options and click **Apply**.

   The selected user is granted administrative access to the EUL.
5. From the user and responsibility drop-down list, select the appropriate responsibility for which you want to assign administrative access to this EUL.
   - Ensure the Responsibility box is checked.
   - Optionally, you can clear the User box to reduce the amount of records appearing in this list.

     For implementations using the seeded SYSADMIN Oracle Applications user account in step 3 above, select the System Administrator responsibility.

For other implementations, any responsibility is appropriate (including System Administrator), as long as the user selected in step 3 already has this responsibility.

6. From the Privileges list, select all options and click **Apply**.

   The selected responsibility is granted administrative access to the EUL.

7. Click **OK** to save your work and exit the Privileges window.

8. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

   All remaining implementation or use steps in this document should be performed by the user and responsibility for which administrative access to this EUL is granted.

Guidelines

- For implementations with access to the seeded SYSADMIN Oracle Applications user account, using this account is appropriate.
- For other implementations, designate a user as appropriate. This user must be assigned administrative privileges to this EUL.

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.4 Importing Oracle Discoverer Objects into an End User Layer

This procedure is required for all enterprises that want to implement Oracle Scripting survey analysis capabilities.

Perform the following to import Oracle Discoverer objects (a business area and workbooks) into an end user layer. Use the BIS EUL if accessible to your user. Otherwise, use the EUL you created.

Prerequisites

- You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.
- This user must have administrative privileges to this EUL. For implementations with access to the BIS EUL and the seeded SYSADMIN Oracle Applications user account, using this account is appropriate.
- If you have access to the BIS EUL but do not have access to the seeded SYSADMIN Oracle Applications user account, perform the procedure described in Section C.1.3, "Assigning End User Layer Implementation Privileges" in order to designate an appropriate user and responsibility with administrative access to the appropriate EUL.

Responsibility

Use the responsibility for which EUL administrative privileges are assigned. See prerequisites above for more information.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with administrative privileges to this EUL.

   For example:

   Username: sysadmin:System Administrator (case-sensitive)

   Password: sysadmin

   Connect: mydatabaseinstance

2. Click **Connect**.

   The Load Wizard appears.

3. From the Load Wizard window, select Open an existing business area.

   The Load Wizard: Step 1 window refreshes, listing all business areas accessible to your user account.

4. In the business area selection list, if **Survey Analysis** appears, you do not need to import any EEX files.

   In the business area selection list, if the **Survey Analysis** business area is *not listed*, you must import the Oracle Discoverer export (EEX) files for use with Oracle Scripting survey reporting.

5. From the Load Wizard window, select **Cancel**.

   The Load Wizard window closes.

6. Optionally, close the Administration Tasklist window, which is not required.

7. From the **File** menu, select **Import**.

   The Import Wizard: Step 1 window appears.

8. From the Import Wizard window, click **Add...**

   The Open window appears.

9. In the Open window, navigate to the directory in which the Oracle Scripting EEX files have been staged in preparation for import.

   In this release, there are 22 EEX files required for Oracle Scripting survey reports.

10. Select all required EEX files and click **Open** to add them to the list.

    The Open window closes. The EEX files to import are now listed in the Import Wizard: Step 1 window.

11. Click **Next**.

    The Import Wizard: Step 2 window appears.

12. In the Import Wizard: Step 2 window, set the following options:

    - In the *If two objects match then what action should occur* area, select **Refresh the object** and check **Preserve display related properties**.
    - In the *How should two objects be matched up* area, select **Match objects by identifier**.
    - In the *Should the current user take ownership of imported workbooks* area, select **Always take ownership of imported workbooks**.

13. In the Import Wizard: Step 2 window, click **Next**.

    The Import Wizard: Step 3 page appears.

14. In the Import Wizard: Step 3 page, click **Start** to begin importing the selected EEX files.

15. When the importation completes, click **Finish**.

    The Import Wizard closes.

In the Data tab of the Discoverer Administration Edition work area, the business area you imported appears as an expandable navigation tree.

Each business area contains folders (equivalent to tables and views) and each folder contains items (equivalent to columns in the table or view).

If enabled, the Administration Tasklist also appears.

16. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.5 Assigning Responsibility and Business Area Privileges

The user and responsibility established in Section C.1.3, "Assigning End User Layer Implementation Privileges" has full administrative access to this EUL.

Perform this procedure to designate a user and responsibility for using Oracle Discoverer User Edition to generate and view workbooks, without administrative access.

For implementations with access to the seeded SYSADMIN Oracle Applications user account, performing this step allows you to have Oracle Discoverer survey analysis workbook users who do not have unrestricted access to Oracle Applications.

This procedure results in assigning permissions to access the workbooks to a specific responsibility. Following this procedure, assign the designated responsibility to workbook user accounts.

Prerequisites

- You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.
- This user must have administrative privileges to this EUL. For implementations with access to the BIS EUL and the seeded SYSADMIN Oracle Applications user account, using this account is appropriate.

Responsibility

Use the responsibility for which EUL administrative privileges are assigned.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with administrative privileges to this EUL.

   For example:

   Username: sysadmin:System Administrator (case-sensitive)

   Password: sysadmin

   Connect: mydatabaseinstance

2. Click **Connect**.

   The Load Wizard appears.

3. From the Load Wizard window, select **Cancel**.

   The Load Wizard window closes.

4. Optionally, close the Administration Tasklist window, which is not required.
5. From the Tools menu, select **Privileges**.

   The Privileges window appears with the Privileges tab selected.

6. From the user and responsibility drop-down list, select the appropriate responsibility for which you want to assign user access to the appropriate set of workbooks.

   - Ensure the Responsibility box is checked.
   - Optionally, you can clear the User box to reduce the amount of records appearing in this list.

For implementations using a responsibility specifically created for this purpose, ensure you select the new responsibility. For example, if you created a responsibility called **Survey Analysis**, select this responsibility.

7. From the Privileges list, select the User Edition box.

   The checkbox options below the User Edition category enable.

8. From the User Edition category, select all options and click **Apply**.

9. Click **OK** to save your work and exit the Privileges window.

10. From the Tools menu, select **Security**.

    The Security window appears with the business area to user tab selected.

11. In the business area to user tab of the Security window, in the Business area list, select Survey Analysis.

    Note: If the Survey Analysis business area is not listed, perform the procedure described in Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer". If you still do not see the Survey Analysis business area, consult your Oracle Discoverer administrator.

12. ensure the appropriate business area is selected.

13. From the Available users/responsibilities list, select the appropriate Oracle Applications responsibility for which you have granted user edition privileges in step 6 above, and click the right arrow.

    The selected responsibility appears in the Selected users/responsibilities list.

14. When satisfied with your selections, click **Apply** and then **OK**.

    The Security window closes.

    You have now granted permission for any user with a specific responsibility to open the Oracle Discoverer User Edition tool and to access the Survey Analysis business area. You must perform the next procedure to grant this responsibility permission to open the appropriate workbooks.

15. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.6 Assigning Workbook Privileges

Perform this procedure to grant to a specific responsibility the privileges to open the appropriate workbooks for survey analysis.

This step must be performed for all implementations for which the procedure in Section C.1.5, "Assigning Responsibility and Business Area Privileges" was performed.

Prerequisites

- You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.
- This user must have administrative privileges to this EUL. For implementations with access to the BIS EUL and the seeded SYSADMIN Oracle Applications user account, using this account is appropriate.
- This user must be the administrative user who imported the workbooks.

Responsibility

Use the responsibility for which EUL administrative privileges are assigned.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with administrative privileges to this EUL who is also the owner of the workbooks.

   For example:

   Username: sysadmin:System Administrator (case-sensitive)

Password: sysadmin

Connect: mydatabaseinstance

2. Click **Connect**.

   The Workbook Wizard appears.

3. From the Workbook Wizard window, select **Cancel**.

   The Workbook Wizard window closes.

4. From the File menu, select **Manage Workbooks** > **Sharing.**

   The Share Workbooks window appears with the Workbook to user tab selected.

5. From Workbook drop-down list, select the first workbook.

   The list of values in the Workbook list include all six workbooks in the Survey Analysis business area.

6. In the Users area, from the Available list, select the responsibility for which User Edition privileges were granted in the previous procedure and click **Add**.

   The selected responsibility appears in the Shared list in the Users area.

7. Click **OK**.

   The Share Workbooks window closes.

8. Repeat steps 4 through 7 for each workbook in Survey Analysis business area.

9. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

   Implementation of the survey analysis workbooks is now complete. Perform the next procedure to view survey analysis workbooks.

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.7 Verifying the End User Layer

Once all implementation steps described above have been performed, you can verify an Oracle Discoverer User Edition's access to the Oracle Scripting reports available in the Survey Analysis business area.

Perform this procedure to verify the implementation.

Note: This step is optional, to assist in troubleshooting.

Prerequisites

- Oracle Scripting-specific workbooks must already be imported.
- Your username should have the responsibility for which the User Edition privilege, business area privilege, and privilege to view survey analysis workbooks have been granted.
- You must have access to Oracle Discoverer User Edition (client/server edition).
- You must know the Oracle database username who owns the EUL.

Responsibility

The appropriate responsibility varies, based on implementation choices.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with the appropriate responsibility to view survey analysis workbooks.

   For example:

   Username: anyuser:Survey Analysis (case-sensitive)

   Password: yourpassword

   Connect: mydatabaseinstance

2. Click **Connect**.

   The Workbook Wizard appears.

3. From the Workbook Wizard window, select **Cancel**.

   The Workbook Wizard window closes.

4. From the Tools menu, select **Options...**

The Options window appears.

5. Click the **EUL** tab.
6. From the Select a default EUL list, ensure the Oracle database user that created the EUL appears.
7. Click **OK**.
8. From the File menu, select **Open**.

   The Open Workbook window appears.
9. Click Database and then click **Open**.

   The Open Workbook from Database window appears.

   All workbooks for which this responsibility has permissions appear in the list.
10. Select the appropriate workbook and click **Open**.
11. If the Workbook in Other Database Account window appears, select **Open the workbook in the current database account** and click **OK**.
12. An Oracle Discoverer message will appear asking for confirmation to run the query for the selected workbook.

    You have completed the verification. Since you do not need to execute the query, click **No**.
13. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

References

- Summarize Survey Data Concurrent Program
- Appendix C.2, "Survey Workbook Descriptions"

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"

- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.8 Viewing Oracle Scripting Survey Reports

Once implemented, Oracle Discoverer User Edition users with the appropriate responsibility can generate Oracle Scripting survey reports using the workbooks imported into the Survey Analysis business area.

Perform this procedure to view Oracle Discoverer workbooks in the Survey Analysis business area and generate fresh reports.

Prerequisites

- Oracle Scripting-specific workbooks must already be imported.
- Your username should have the responsibility for which the User Edition privilege, business area privilege, and privilege to view survey analysis workbooks have been granted.
- You must have access to Oracle Discoverer User Edition (client/server edition).
- Three workbooks require the Summarize Survey Data concurrent program to be executed prior to generating workbooks, to ensure current data. These include the BIX: Survey Campaign Summary Report, the BIX: Survey Deployment Detail Report, and the BIX: Survey Question Frequency Report.

Responsibility

The appropriate responsibility varies, based on implementation choices.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with the appropriate responsibility to view survey analysis workbooks.

   For example:

   Username: anyuser:Survey Analysis (case-sensitive)

   Password: yourpassword

   Connect: mydatabaseinstance

2. Click **Connect**.

   The Workbook Wizard appears.

3. From the Workbook Wizard, select **Open an existing Workbook**.

The Workbook Wizard window refreshes, showing workbook location options.

4. Select **Database**.

   The Workbook Wizard window closes. The Open Workbook from Database window appears.

   All workbooks for which this responsibility has permissions appear in the list.

5. From the Open Workbook from Database window, select the appropriate workbook and click **Open**.

6. If the Workbook in Other Database Account window appears, select **Open the workbook in the current database account** and click **OK**.

7. An Oracle Discoverer message appears asking for confirmation to run the query for the selected workbook. To execute the query, click **Yes**. To view the workbook without data, click **No**.

   Note: To run the query at a later time, select **Sheet > Refresh Sheet**.

   Click **Yes**.

   The Parameter Wizard appears.

   For specific guidance on workbook parameters, refer to the detailed workbook descriptions detailed in Appendix C.2, "Survey Workbook Descriptions".

8. Select the appropriate parameters and click **Finish** to execute the query.

   The query executes, populating the workbook with data.

   Note: Three workbooks (BIX: Survey Campaign Summary Report, BIX: Survey Deployment Detail Report, and BIX: Survey Question Frequency Report) require the Summarize Survey Data concurrent program to be executed prior to generating workbooks, to ensure current data. For any of these workbooks, if a query is executed *prior to the execution of the concurrent program*, and no data is available, the report will appear displaying column headings only. If the concurrent program had been previously executed, but not recently, stale data will appear in the workbook.

9. Optionally, to export the workbook to Microsoft Excel, click the Excel icon in the Discoverer User Edition toolbar.

References

- Summarize Survey Data Concurrent Program
- Appendix C.2, "Survey Workbook Descriptions"

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"

C.2 Survey Workbook Descriptions

The following table indicates the name of each workbook, its filename, and a description of each workbook.

Workbook Name	File Name	Description	Tables or Views Accessed by Workbook
Survey Question Frequency	bixsvyqsfrwbo.eex	This report provides answer frequency data for each survey deployment, showing only non-text fields. This report can be used to show how people have responded to survey questions at a summary level for each survey deployment. It provides a table of all non-text questions, plus the frequency of responses for all the questions.	▪ ies_svy_ques_data_v

Workbook Name	File Name	Description	Tables or Views Accessed by Workbook
Survey Question Detail	bixsvyqsdtwbo.eex	This report provides the detailed response for each survey that has been completed. It compiles all answers that each respondent has provided. It is sorted by response date.	■ ies_svy_surveys_all ■ ies_svy_cycles_all ■ ies_svy_deplyments_all ■ ies_svy_response_entries ■ ies_question_data ■ ies_questions ■ ies_question_types ■ ies_panels ■ ies_answers.ies_svy_list_entries_v ■ ies_ams_list_entries_v
Survey Question Text	bixsvyqstxwbo.eex	This report provides detail solely on the text data type fields (text field, text area, and password field) collected within a survey. This is useful for providing a collection of comments provided by survey respondents for specific questions.	■ ies_svy_surveys_all ■ ies_svy_cycles_all ■ ies_svy_deplyments_all ■ ies_svy_response_entries ■ ies_question_data ■ ies_questions ■ ies_question_types ■ ies_panels ■ ies_answers.ies_svy_list_entries_v ■ ies_ams_list_entries_v
Survey Campaign Summary	bixsvycmpswbo.eex	This report provides Summary information on each Survey Campaign, with totals provided at the survey campaign and cycle level. It allows you to get an overview of how many surveys have been sent out, whether there have been errors with the surveys, and the response ratio.	■ ies_svy_summary_stats_v ■ ies_svy_reminders_v

Workbook Name	File Name	Description	Tables or Views Accessed by Workbook
Survey Deployment Detail	bixsvydpldwbo.eex	This report provides detail information on each deployment, with totals provided at the cycle and deployment level.	▪ ies_svy_summary_stats_v ▪ ies_svy_reminders_v
Survey Cross Tabulation	bixiescrsstbrwbo.eex	This report allows the administrator to select a specific survey deployment, and then cross-tabulate the results from two different questions. You can select a question and a target value from one question and use that as the establishing criteria for reporting on a different response. For example, you can select all survey records in which the answer to the Overall Satisfaction question is "3 - somewhat satisfied," and then for that set of records, display the responses for a second question (such as "would you use this service again").	▪ ies_svy_surveys_v ▪ ies_svy_deplyments_v ▪ ies_svy_cycles_v ▪ ies_questions ▪ ies_answers ▪ ies_question_data ▪ ies_svy_response_entries

This section includes the following topics:

Appendix C.2.1, "Survey Question Frequency Workbook"

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.5, "Survey Deployment Detail Workbook"

Appendix C.2.6, "Survey Cross Tabulation Workbook"

See Also

Appendix C.1, "Implementing Survey Analysis Discoverer Workbooks"

C.2.1 Survey Question Frequency Workbook

This report provides answer frequency data for each survey deployment, showing only non-text fields. This report can be used to show how people have responded to survey questions at a summary level for each survey deployment. It provides a table of all non-text questions, plus the frequency of responses for all the questions.

Survey Question Frequency Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Campaign Name	Name of the survey campaign.
Cycle Name	Name of the cycle, in alphabetical order.
Deployment Name	Name of the deployment, in alphabetical order.
Panel Name	Name of the panel. These appear in the order in which the panels displayed in the script.
Question Name	Name of the question. These appear in the order in which the question was asked in the survey questionnaire.
Available Answers	Answers for each question. These appear in the order in which they are listed for the question.
Selected Answers	The answers that have been selected by survey respondents. Excludes abandoned survey responses. In other words, if a respondent did not answer even a single question in a survey questionnaire, then that survey transaction is excluded from these totals. If a respondent answered one or more questions, with null values for any other question, then the responses are included in these totals.
Total Answers per Question	The total number of responses for the question (sum of all the selected answers). Excludes aborted surveys.
Answer Frequency	Percentage of answers for this question. Calculated only for data types of radio buttons, check boxes, and drop-down lists. It is not valid for text fields or text areas.

Selection Criteria

- When executed, this workbook selects a specific survey campaign, and can use wild card characters on the survey campaign's cycles and deployments. For instance, users can see all cycles and all deployments for a particular survey campaign, or users can select for a specific deployment. If users select multiple cycles and deployments, the workbook sorts by cycle, then deployment, in alphabetical order.

- Excludes aborted survey responses (those with 0 answers). Only for abandoned (1 or more answers) and completed survey responses.

Sort Criteria

Survey campaign

Cycle name/ID

Survey deployment (breaks to a new page for each survey cycle, each deployment)

- Panel name
  - Questions listed in the order in which they appear in the panel

Calculations

- Calculates the total answers for each question for each deployment
- Calculates the answer frequency for each question for each deployment
- For each deployment, provides a sub-total which shows the total number of survey responses that were evaluated in this report

See Also

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.5, "Survey Deployment Detail Workbook"

Appendix C.2.6, "Survey Cross Tabulation Workbook"

C.2.2 Survey Question Detail Workbook

This report provides the detailed response for each survey that has been completed. It compiles all answers that each respondent has provided. It is sorted by response date.

Survey Question Detail Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Submit Date	Date the response was entered - RESPONSE_COLLECTED_DATE
Response ID	Survey Respondent ID
First Name	Contact's first name from the AMS List Entries table (for targeted survey deployments only). Blank if standard (non-list-based) deployment.
Last Name	Contact's first name from the AMS List Entries table (for targeted survey deployments only). Blank if standard (non-list-based) deployment.
Panel Name	Name of the panel. These appear in the order in which the panels displayed in the script.
Question Name	Name of the question. These appear in the order in which the question was asked in the survey questionnaire.
Answer	The survey respondent's selected answer to the question.

Selection Criteria

- Select a specific survey deployment
- Select on a Submit Date Range
- Excludes aborted surveys (those with 0 answers)

Sort Criteria

Survey campaign

Cycle name/ID

Survey deployment

- Panel name
  - Questions listed in the order in which they appear in the panel.

See Also

Appendix C.2.1, "Survey Question Frequency Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.5, "Survey Deployment Detail Workbook"

Appendix C.2.6, "Survey Cross Tabulation Workbook"

C.2.3 Survey Question Text Workbook

This report provides detail solely on the text data type fields (text field, text area, and password field) collected within a survey. This is useful for providing a collection of comments provided by survey respondents for specific questions.

Survey Question Text Responses Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Submit Date	Date the response was entered - RESPONSE_COLLECTED_DATE
Survey respondent ID	RESPONSE_ID
Panel Name	Name of the panel. These appear in the order in which the panels displayed in the script.
Question Name	Name of the question. These appear in the order in which the question was asked in the survey questionnaire.
Answer	The survey respondent's selected answer to the question.

Selection Criteria

- Must select a specific survey deployment.
- Can select a specific Response Collected Date range
- Excludes abandoned surveys (those with 0 answers)

Sort Criteria

Submit Date

- Panel name
  - Questions listed in the order in which they appear in the panel.

Calculations

Total responses evaluated.

See Also

Appendix C.2.1, "Survey Question Frequency Workbook"

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.5, "Survey Deployment Detail Workbook"

Appendix C.2.6, "Survey Cross Tabulation Workbook"

C.2.4 Survey Campaign Summary Workbook

This report provides Summary information on each Survey Campaign, with totals provided at the survey campaign and cycle level. It allows you to get an overview of how many surveys have been sent out, whether there have been errors with the surveys, and the response ratio.

Survey Question Detail Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Survey Campaign Name	Survey Campaign Name
Cycle Name	Name of Survey Cycle
Deployment Name	Name of Survey deployment
Deployment Status	Status of the deployment. Status types include Open, Pending, Error, Active, Cancelled or Incomplete. For a description of each status, see Deployment Status.
Number of Targeted Respondents	For targeted deployments, the total number of survey targets in a list. For standard deployments, the sum of abandoned and completed surveys (represents the number of opportunities to get survey feedback).
Number of Survey Errors	Number of surveys that could not be delivered to list member because of e-mail address or server errors.
Number Surveys Aborted	Number of surveys that have been opened, but no answers have been recorded.

Column Name	Description
Number Abandoned	Number of surveys that have been opened, but for which no answers were provided.
Number Surveys Completed	Number of surveys that have one or more answers.
Response Ratio	Completed responses divided by total number of invitations

Selection Criteria

- Can select survey campaign name to get a report on a specific survey campaign.
- Can also selectively report on survey campaigns by using wildcard search characters.

Sort Criteria

Survey campaign

- Cycle
  - Deployment

Calculations

- Number of survey errors
- Number of opportunities
- Number of abandoned surveys
- Number of completed surveys
- Response ratio
- Totals for Opportunities, Errors, Abandons, Completed (rolled up at cycle level)
- Totals for Opportunities, Errors, Abandons, Completed (rolled up at survey campaign level)

See Also

Appendix C.2.1, "Survey Question Frequency Workbook"

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.5, "Survey Deployment Detail Workbook"

Appendix C.2.6, "Survey Cross Tabulation Workbook"

C.2.5 Survey Deployment Detail Workbook

This report provides detail information on each deployment, with totals provided at the cycle and deployment level.

Survey Deployment Detail Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Campaign Name	Name of the survey campaign.
Cycle Name	Name of the cycle, in alphabetical order.
Deployment Name	Name of the deployment, in alphabetical order.
Deployment Status	Status of the deployment. Status types include Open, Pending, Error, Active, Cancelled or Incomplete. For a description of each status, see Deployment Status.
Start Date	The start date for a deployment. This will be blank at the survey campaign and cycle levels, which do not have start dates.
End Date	The deployment end date for a deployment. This will be blank at the survey campaign and cycle levels, which do not have end dates.
List Name	Name of list used to send invitation or reminder master documents.
Number of Invitations Sent	The number of survey invitations sent. This value will be blank for standard (non-list-based) survey deployments.
Number of Reminders Sent	The number of survey reminders sent. This value will be blank for standard (non-list-based) survey deployments.
Number Surveys Completed	Number of surveys that have one or more answers.
Number of Abandoned Surveys	Number of surveys that have been opened, but for which no answers were provided.
Number of Errors	Number of surveys that could not be delivered to list member because of e-mail address or server errors.
Response Ratio	Completed responses divided by total number of invitations

Selection Criteria

- Can select from survey campaign name, cycle name, and deployment name.
- Can use % wildcard search character.
- Can select by deployment status (Active, Pending, Open, Closed).

Sort Criteria

Survey campaign

- Cycle
  - Deployment
    - Deployment Status

Calculations

- Number of Invitations Sent
- Number of Reminders Sent
- Number of survey errors
- Number of opportunities
- Number abandoned surveys
- Number completed surveys
- Response ratio
- Totals for Opportunities, Errors, Abandons, Completed - At Deployment level
- Totals for Opportunities, Errors, Abandons, Completed - Rolled up at Cycle level
- Totals for Opportunities, Errors, Abandons, Completed - Rolled up at Survey Campaign level

See Also

Appendix C.2.1, "Survey Question Frequency Workbook"

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.6, "Survey Cross Tabulation Workbook"

C.2.6 Survey Cross Tabulation Workbook

This report allows the administrator to select a specific survey deployment, and then cross-tabulate the results from two different questions. You can select a question and a target value from one question and use that as the establishing criteria for reporting on a different response. For example, you can select all survey records in which the answer to the Overall Satisfaction question is "3 - somewhat satisfied," and then for that set of records, display the responses for a second question (such as "would you use this service again").

Survey Deployment Detail Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Selection Question	The question that is to be used as the primary selection
Cross Tab Question	The question that is cross tabulated against the selection question
Selection Question Choices	The available answers under the selection question
Cross Tab Answers	The available answers under the cross tab question
Cross Tab Answer Frequency	The frequency at which each answer was chosen for this particular selection question and cross tab question
Cross Tab Answer Ratio	The percentage of the individuals who had responded with a particular value under the selection question
Cross Tab Answer Percentage	The percentage of individuals who had this particular value for their selection question, out of all people surveyed.

Selection Criteria

- Must select a specific Survey Deployment
- Select a specific question and answer pair as the primary selection criteria
- Select the cross-tabulation question and answer definitions
- Excludes aborted surveys (those with 0 answers)

Sort Criteria

Sort by the answers in the Selections Question Choices column, then by the Cross Tab Answers column

Calculations

- Percentage of respondents who had each specific response for the cross-tabulation question.
- Total ratio of respondents out of the entire deployment who had the specific question/answer pair. For instance, if selecting for an age range of between 40-50, this would show the percentage of total respondents for the entire survey who were in this age range.

See Also

Appendix C.2.1, "Survey Question Frequency Workbook"

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.5, "Survey Deployment Detail Workbook"

Abbreviations and Acronyms

Abbreviations and Acronyms

Acronym	Definition
ANI	Automatic Number Identification
AOM	Application Object Model
API	Application Programming Interface
CRM	Customer Relationship Management
CTI	Computer-Telephony Integration
DBA	Database Administrator
DNIS	Dialed Number Identification Service
ERP	Enterprise Relationship Planning
EUL	End User Layer
GIF	Graphics Interchange Format
GUI	Graphic User Interface
HR	Human Resources
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol – Secure
IG	Implementation Guide

Acronym	Definition
JAR	Java Archive
JDBC	Java Database Connectivity
JDK	Java Development Kit
JRAD	Java Rapid Application Development
JSP	Java Server Pages
JTF	Java Technology Foundation
JVM	Java Virtual Machine
PL/SQL	Programming Language/Structured Query Language
SID	System Identifier
SQL	Structured Query Language
sysadmin	Systems Administrator
UI	User Interface
URL	Uniform Resource Locator

Product-Related Acronyms

Acronym	Definition
HRMS	Oracle Human Resource Management Systems
ICX	Internet Cartridge Exchange
JTA	Oracle Common Application Components
JTF	Oracle CRM Data Model
JTT	Oracle CRM Technology Foundation

Oracle Product Designations

Acronym	Definition
AMS	Oracle Marketing
AST	Oracle TeleSales
BIS	Oracle Business Intelligence System
BIX	Oracle Interaction Center Intelligence
IEO	Oracle Interaction Center Technology
IES	Oracle Scripting
IEX	Oracle Collections
ICX	Oracle Self-Service Web Applications
JTA	Oracle Common Application Components
JTF	Oracle CRM Data Model
JTT	Oracle CRM Technology Foundation
JTO	Oracle One-to-One Fulfillment

Glossary

action

A Script Author command that executes at runtime upon evaluation of the object containing the action. Actions can be associated with branches, or with blocks if designated as an API block. See also pre-action and post-action.

answer collection

The recording of end user responses ("answers") to all question controls ("questions") during a script transaction. Answer collection is performed by the Scripting Engine for a script executed in either the agent interface or the Web interface, when the Answer Collection option is selected as a global script property. Answers are collected in the IES_QUESTION_DATA table in the Oracle Applications schema. Script end user responses are collected for each "question" or answer definition designated as collectable.

answer definition

Obsolete term. See question.

answer UI type

Obsolete term. See question user interface type.

answer user interface type

Obsolete term. See question user interface type.

argument

Data provided to a function or method for use in its calculations and processing. Arguments are passed to the function or method by listing them in parentheses in the function or method name. In a Java method, these are also called parameters.

best practice Java method

Oracle Applications includes Java methods written specifically for Oracle Scripting to quickly enable frequently requested script runtime functionality. Class files containing these "best practice" Java methods are included in APPS.ZIP, each in the package oracle.apps.ies.bestpractice.<Class Name>.

best practice survey

Oracle Corporation provides some pre-built survey questionnaire scripts flows to serve as examples of how to customize scripts specifically intended for use as survey questionnaires. Aspects of these scripts can also be used in scripts to be executed in the agent interface. Best practice survey scripts are placed on the 11i APPL_TOP in the directory $IES_TOP/scripts during installation or patching of Oracle Scripting. These are provided on an as-is basis. Use of best practice surveys requires customization and is not supported.

blackboard

Virtual memory space in Oracle Scripting where information is stored in key/value pairs. This information is only retained for the duration of a single Scripting interaction.

building block

A portion of a Script Author script that contains functionality integrating with other Oracle Applications. Building blocks typically contain one or more API and typically perform a task-based operation such as creating a service request, registering for an event, and so forth. Building blocks are not intended to be used as standalone scripts, but are expected to be imported into a script and modified as appropriate. These are provided on an as-is basis. Use of building blocks requires customization and is not supported. Building block script components are created and supported by application development teams. As of this release, building block scripts are available from the Oracle Scripting and Oracle Marketing product teams. For information on building blocks, refer to *Oracle Scripting Developer's Guide* or product documentation for the specific Oracle application.

campaign

A focused effort to achieve a particular goal from a targeted population over a specific period of time for a particular business purpose. Also see marketing campaign.

class

A set of related objects that share common characteristics. In Java, a class contains a set of methods.

collectable

A boolean property of a Script Author question or answer definition within a panel. This option is selected by default in the general tab of the data dictionary of each specific question defined. Individual answers can be designated as uncollectable by clearing the collectable option. When answer collection is enabled at the global script level, all script end user responses provided in a script session will be saved for questions designated as collectable. Also see answer collection.

CRM Resource Manager

Oracle CRM Resource Manager is a responsibility included with a Rapid Install of Oracle CRM Applications so that HRMS person types (most often, employees) required by Oracle products can be created. It does not have the full functionality of HRMS. Note that if HRMS is installed, HRMS employees must be created using that tool and cannot be created with CRM Resource Manager.

cycle

A cycle is a repeatable survey event, utilizing the same script. Multiple cycles will also allow the same survey to be given through different media types, when supported. Multiple cycles are typically used to measure the same characteristics at a different moment in time.

deployed scripts

Deployed scripts are syntactically correct Script Author scripts that are executable using the Scripting Engine. The only method to deploy a script is to use Script Author (either using the **Tools > Deploy Script** command from a graphical script, or the **Save, Deploy, and Exit** command from the Script Wizard). When a script is deployed it is written to the IES_DEPLOYED_SCRIPTS table of the applications database. Deployed scripts that are selected as the survey questionnaire for a valid survey campaign can be executed in a Web browser using the Scripting Engine Web interface.

deployment

A deployment is the lowest and most granular set of requirements for a survey campaign. A deployment is associated with a specific survey and cycle, and identifies specific response start and end dates. List-based deployments identify Oracle Marketing lists and Oracle One-to-One Fulfillment templates. Deployments must be set to active status (they must be "deployed") before a survey campaign commences.

deployment ID

An identification number for each deployment in a particular instance. This is created upon setting a deployment to Active status. All respondents in a particular deployment will have the same deployment ID.

dID

See deployment ID.

error page

An error page is a survey resource that displays when the Scripting Engine encounters an error in processing a script in the Web interface (during execution of a survey or Web script in a Web browser). Survey resources defined for execution in the JTT technology stack are JSP files. Survey resources for execution in the OAF technology stack are either HTML files or (for error page or final page resources) URLs accessible to Oracle Applications at runtime, which redirect the end user to the designated location on the Internet or intranet. Error page resources are mandatory for JTT survey campaigns. If no specific error page is defined for an OAF survey campaign, a simple error page will be automatically generated by the application.

In the JTT technology stack, scripts executed from self-service Web applications can use the seeded test error page for menu-based scripts (IESSVYMENUBASEDTESTERROR.JSP), whereas surveys or web scripts launched using the guest user may user error pages based on the seeded error page for non-hosted scripts, IESSVYTESTERROR.JSP.

final page

A final page is a survey resource that displays after the Scripting Engine processes the last panel in the flow of a script executed in the Web interface (during execution of a survey or Web script in a Web browser). Survey resources defined for execution in the JTT technology stack are JSP files. Survey resources for execution in the OAF technology stack are either HTML files or (for final page or error page resources)

URLs accessible to Oracle Applications at runtime, which redirect the end user to the designated location on the Internet or intranet. When a file is used as the final page instead of a URL for redirect, it typically includes a thank-you message to the script end user, and indicates that the survey or Web script has ended successfully. Final pages often contain a hypertext link to the enterprise's home page or to the next logical destination for the end user (based on the purpose of the script). Final page resources are mandatory for JTT survey campaigns. If no specific final page is defined for an OAF survey campaign, a simple final page will be automatically generated by the application.

footer section

A footer section is a survey resource that, if associated with a survey campaign, displays at the bottom of each page (except the error page or final page) when a script is executed in a Web browser. Since each page represents a panel, the footer appears immediately below panel content (typically, below the Continue button). The footer section may reference corporate logos or other standardized graphics in GIF or JPEG format. If so, these graphics need to be stored on a directory accessible to the Web server and referenced appropriately in the footer section code. Survey resources defined for execution in the JTT technology stack are JSP files. Survey resources for execution in the OAF technology stack are HTML files. Footer section resources are the only optional survey resources for JTT survey campaigns. If no specific footer section is defined for any survey campaign (regardless of technology stack), no footer will appear on each page at runtime.

footprinting

Footprinting is the recording in the Oracle Applications database of the names of each panel in a script transaction that is visited during a script transaction, and the duration of time (in milliseconds) prior to the activation of the next panel. This feature can provide useful script tuning data. The default for this property is **true**. When enabled (or when the Answer Collection property is **true**), this data is saved in the IES_PANEL_DATA table in the Oracle Applications schema.

graphical script

A script contains the business rules, text and graphics to progress the script end user from panel to panel at runtime. Runtime scripts are executed using any interface of the Scripting Engine component of Oracle Scripting. A graphical script is created using the graphical tools of the Script Author, the authoring and development component of Oracle Scripting. Graphical scripts can also be created by graphing a wizard script. At runtime, scripts display panels containing questions, answers, and graphics. Each panel includes a Continue button. The graphical script end user progresses through each panel. The flow of the script is controlled by business rules built using the standard Script Author graphical tools, including associated custom code, and may be a rigid flow or may change dynamically, based on the end user's responses. Graphical scripts are created, edited and deployed using the Script Author Java applet, and can be listed, deleted, or associated with custom Java from the Scripting Administration console.

header section

A header section is a survey resource that, if associated with a survey campaign, displays at the top of each page (panel) when a script is executed in a Web browser (except the error page or final page). Immediately below the header section, the content of the current panel appears. The header section may reference corporate logos or other standardized graphics in GIF or JPEG format. If so, these graphics need to be stored on a directory accessible to the Web server and referenced appropriately in the footer section code. Survey resources defined for execution in the JTT technology stack are JSP files. Survey resources for execution in the OAF technology stack are HTML files. Header section resources are mandatory for JTT survey campaigns. If no specific footer section is defined for an OAF survey campaign, no information appears above panel content for each page at runtime.

invitation

An e-mail message master document, inviting potential survey respondents (identified in an Oracle Marketing list) to participate in a survey. Each e-mail message contains a customized URL, from which the invited list member can respond to the specified survey.

JRAD

Java Rapid Application Development, an Oracle proprietary technology stack also referred to as Oracle Applications Framework (OAF). This is the technology upon which the Survey Administration console is built.

JTA

Java Technology Applications, an Oracle proprietary technology stack referring to CRM Common Application Components.

JTF

Java Technology Framework, an Oracle proprietary technology stack referring to CRM Foundation products. Oracle applications using underlying CRM Foundation technology are further divided into the CRM Applications Foundation technology stack (known as JTA) and CRM Technology Foundation (JTT) technology stack.

JTT

Java Technology Tools, an Oracle proprietary technology stack referring to CRM Common Application Components. This is generally referred to as the Oracle CRM Technology Foundation technology stack.

list

A list is a set of records in Oracle Marketing that contains records for individuals or organizations, and contact information (at minimum, an e-mail address) for each. From an Oracle Scripting perspective, the relevance of a list is to serve as the source for potential survey respondents for a targeted survey campaign deployment. Each record represents one individual who will receive an invitation (an HTML document sent using Oracle One-to-One Fulfillment) to participate in the questionnaire. List members may also receive reminders (also HTML documents) to participate in a survey, which includes the survey end date. Lists are built using Oracle Marketing's list management feature. Use of targeted (list-based) surveys requires Oracle Marketing and Oracle One-to-One Fulfillment.

list member

Any individual included in a list built using Oracle Marketing's list management feature. All list members are potential survey respondents.

marketing campaign

A marketing campaign is a collection of requirements created in Oracle Marketing for the purpose of executing specific business rules of a marketing effort within an enterprise or interaction center. It is the primary organizational unit of a focused marketing effort. Campaigns define the marketing effort and, through association with other units, define schedules for execution and target populations for the effort. One marketing campaign must have at least one campaign schedule associated with it, and may have many. Campaign schedules can be targeted for different execution channels (such as e-mail, telephony, or traditional postal service mail). Each campaign schedule has a target group (list or lists created using Oracle Marketing to define the target population for the campaign). The marketing campaign is used by several Oracle Applications, including Oracle TeleSales, Oracle Collections, and Oracle Advanced Outbound Telephony.

master document

A master document is a message template used in Oracle One-to-One Fulfillment. Master documents typically include merge fields populated by an associated SQL query. Upon execution of the query, a copy of the master document is created for each list member, containing the data returned by the query. This creates a personalized message. From a survey perspective, master documents are invitation or reminder messages in HTML format that are sent by e-mail from the fulfillment engine to members of a defined Oracle Marketing list. Master documents are required for targeted survey deployment operations. Master documents and their associated queries are associated with specific Oracle One-to-One Fulfillment templates.

merge field

Merge fields are data elements in an Oracle One-to-One Fulfillment master document surrounded by open and close guillemet (for example, «merge_field»). When a fulfillment request is executed, specified table columns or views are queried. For each list member record, the master document is replicated, with the actual data returned from the query embedded or merged in the master document replicate in place of the merge field. List-based surveys require at minimum one merge field (the survey URL that the recipient must access to participate in the survey). Merge fields must correspond to data returned from a query.

method

A task that can be performed with specific data items or properties. A Java method can be a function (resulting in a return value) or a procedure.

Oracle HRMS

Oracle HRMS is a full-featured enterprise human resources management system. It is an ERP application using Oracle Forms technology to manage human resources used by a variety of ERP and CRM applications.

panel

A panel is the primary object of a script, which contains the information that is displayed by the Scripting Engine (in the panel presentation area or browser window) at runtime. Each panel must contain at least one question and may contain as many as needed. Each question in a panel displays a corresponding question user interface control (button, checkbox, text area, drop-down, and so forth) with which the script end user enters a response at runtime. Panels are created and edited using the Script Wizard or graphical tools.

Panels in wizard or graphical scripts may contain static panel text, and one or more questions. Questions of the appropriate type that are explicitly defined in a wizard script may include question-level validation commands. In addition, panels in graphical scripts may contain graphic images, dynamic text using embedded values, and validation, database or cursor lookup, or other commands (at the panel level or the question level).

If the purpose of a wizard script panel is simply to provide information to the end user, then it can automatically generate a question called Continue, which displays a Continue button at runtime. Panels created with graphical tools must contain one or more questions, for which the properties of each must be explicitly defined.

Whether built with the Script Wizard or in a graphical script, a panel is the only Script Author object to display at runtime in both interfaces (groups, blocks and branches simply contain information for processing a script).

PHP

Personal home page. Accessed when logging into Oracle Self Service applications, the personal home page lists the responsibilities available to the logged in Oracle Applications account in two categories: Self Service, to access self-service applications, and Applications, for access to ERP applications and Forms-based CRM applications.

post-action

A Script Author command that executes immediately following the evaluation of the object with which the post-action is associated, and prior to evaluation of the next object in the flow of the script. Post-actions can be associated with panels, blocks, groups, or the global script. See also action and pre-action.

pre-action

A Script Author command that executes prior to the evaluation of the object with which the pre-action is associated. Pre-actions can be associated with panels, blocks, groups, or the global script. See also action and post-action.

prototype

Prototype is a boolean characteristic of survey campaigns. Prototype survey campaigns are identical to standard survey campaigns, except that the script used as the survey campaign questionnaire is not locked. The purpose is to allow survey campaign administrators more freedom to refine requirements for survey campaigns, including modification to the script for the designated survey campaign. You can exclude prototype survey campaigns from survey campaign lists by selecting the **Exclude prototypes** option. This characteristic can only be selected or cleared while a survey campaign status is Open.

published scripts

Published scripts are stored in the IES_DEV_SCRIPTS table of the applications database, and are not executable. All wizard scripts are published scripts. To publish a graphical script you must be explicitly save it to the database from Script Author by selecting **File > Save As** and selecting the Database tab. Published wizard scripts are always syntactically valid; graphical scripts can be published as works in progress, regardless of their contents.

query

A specific statement of request for information to be returned in the form of records from the database. Queries are constructed in structured query language (SQL). For targeted survey deployments, each Oracle One-to-One Fulfillment master document must have an associated query which obtains information populated into the resulting document in placeholders called merge fields. A query must be valid to return any of the requested information.

question

In Script Author, a question is the property of a panel which results in the appearance of a question user interface (UI) control in the panel at runtime. All panels must contain at least one question and may contain any number. Questions may have commands associated with them in Script Author to validate responses at runtime, or to provide default user responses. When viewed from a graphical script, each question has a data dictionary associated with it, in which information (answer choices, commands) is stored.

question control

See question user interface type.

question UI type

See question user interface type.

question user interface type

The choice of user interface (UI) type selected by the script developer for a question in a panel determines the kind of control that displays at runtime with which the end user can provide input. The full range of question UI types supported by Oracle Scripting include text fields, text areas, radio buttons, checkboxes, buttons, drop-down lists, password fields, checkbox groups and multi-select list boxes. All nine are available in graphical scripts. Wizard scripts do not support the single check box or button question UI types. Each question supports only one question UI type. Once selected in a graphical script, the question user interface type cannot be changed (you must delete the entire question control and add a new one). Panels in graphical scripts that require the button UI type cannot contain any other questions. Some question UI types allow null or open-ended input (text fields, text areas, password fields, checkbox, checkbox group, multi-select list boxes). Others require end user input at runtime (radio buttons, buttons, drop-down lists) unless a default answer choice is associated with that question.

reminder

An e-mail message master document, reminding potential survey respondents (already invited to take a survey) of the appropriate URL to access to respond to the survey. Reminders often include information such as the last day the recipient may participate in the survey. Using the same master document, reminders can be sent in a batch process on a predefined schedule, or can be sent manually for any individual list member.

respondent

An individual responding to a request for information (in the form of participating in a survey questionnaire or Web script). For targeted survey deployments, respondents are list members who respond to an invitation sent by e-mail to participate in the survey.

respondent ID

A unique identification number for each list-based respondent in a particular instance. The URL for a list member includes as parameters both the deployment ID common to all participants in a given deployment and each list member’s unique respondent ID.

rID

See respondent ID.

responsibility

A responsibility is a level of authority that allows a user to access specific functionality and data in Oracle Applications. Responsibilities are assigned by a user with the System Administrator responsibility.

sample

The population from which information is solicited by survey.

script

A script is a miniature program built using the Script Author component of Oracle Scripting and executed at runtime using the Scripting Engine component of Oracle Scripting. The purpose of a script is to facilitate the flow of information between an enterprise and other parties. Each script enforces business rules programmed into it by script developers. Scripts consist primarily of panels with at least one question each; at runtime, script end users must click the Continue button in each panel to progress through the script. Graphical scripts also may contain groups and blocks. By design, scripts either enforce a rigid flow, or employ carefully constructed logic to route the end user through different potential paths in a script. to appropriate panels, based on responses to earlier script questions. Script Author provides the ability to create two kinds of scripts: wizard scripts and graphical scripts. Regardless of creation method, after deployment a script can be executed in any interface of the Scripting Engine. Executing a script in a Web browser requires the creation and activation of a survey campaign deployment.

A script contains the business rules, text and graphics to progress the script end user from panel to panel. Scripts are created using the Script Author component of Oracle Scripting. At runtime, scripts are executed using any interface of the Scripting Engine component of Oracle Scripting. The same script can be executed in the agent interface, or (after creating and deploying a survey campaign) executed in a Web browser as a survey questionnaire or web script accessed from a self-service Web application. Scripts are further qualified by type. See graphical script and wizard script.

script information area

The script information area is a programmable feature of the Script Author in which script developers can place information that appears as a header in any script executed in the Scripting Engine agent interface at runtime. The script information area is a global script property and appears in every session of a script executed in the agent interface when this information is defined. It is referred to in the Script Author user interface as the Static Panel. You can access this area by selecting **File > Script Properties > Static Panel** or by right-clicking on an empty area of the canvas and selecting **Edit Blob Properties > Static Panel**.

static panel

See script information area.

survey campaign

A survey campaign is the collection of requirements needed to execute a script in a Web browser using the Scripting Engine Web interface. This is the super class that references the global script. A survey campaign must have information for at least one cycle and at least one deployment. Survey campaigns are created using the Survey Administration console. As prerequisites to creating a survey campaign, you must create and deploy from Script Author the script containing questions and answer choices, and you must define survey resources.

survey questionnaire

Set of questions built using the Script Author component of Oracle Scripting for execution in the Scripting Engine as a survey questionnaire, typically to gather feedback, obtain market data, tabulate opinion, measure satisfaction, or otherwise collect data directly from specific (customers, employees, prospects) or non-specific target populations. A survey questionnaire script is physically identical to a script executed in the agent interface, although it may be customized to take advantage of sophisticated HTML interpretation available to Web browsers.

survey resource

Survey resources provide functionality to scripts executing in the Scripting Engine Web interface. Survey resource definition is accomplished from the Survey Administration console **Survey Resources** tab > **Create**, and must be defined before they can be used in a survey campaign. The four survey resource types include header section, footer section, error page, and final page resources. Header and footer sections appear in each page of the Web browser, just above or below panel contents, respectively. Error page resources are displayed when an error condition occurs during script execution in a Web browser. Final page resources are displayed after the Scripting Engine processes the last panel in the flow of a script. For survey campaigns defined in the JTT technology stack, survey resources are JSP files, and all resources except the footer page are mandatory. For survey campaigns defined in the OAF technology stack, survey resources are HTML files. For OAF survey campaigns, error page or final page resources can also be URLs accessible to the Web server at script runtime, which redirect the script end user to the specified URL during an error or upon completing the script, respectively.

Physical JSP survey resources corresponding to survey resource definitions must be uploaded manually into the $OA_HTML directory on the applications server and are served by the Web server as appropriate. For OAF survey resources, the HTML files corresponding to the survey resource definition are uploaded to the database at the time of survey resource definition.

For JTT survey campaigns, you can use seeded JSP test resources to test your implementation, or you can customize your own resources based on these seeded resources. These seeded JavaServer page files (IESSVYTESTHEADER.JSP, IESSVYTESTTHANKU.JSP, IESSVYTESTERROR.JSP and IESSVYMENUBASEDERROR.JSP) are located in $OA_HTML in your environment. For error pages with a hosting type of menu-based, use the appropriate seeded resource (IESSVYMENUBASEDERROR.JSP.

survey respondent

See respondent.

SYSDATE

This refers to the current Systems Date of the database. When creating records in a database the SYSDATE is typically time-stamped as creation date, and SYSDATE is also used as a default setting for creating responsibilities from an Oracle Applications perspective.

template

Oracle One-to-One Fulfillment's term for a group of master documents and any collateral. Templates may also include campaign specifications. Templates are identified with a survey campaign at the deployment level to identify invitation and reminder master documents and their associated queries. Only required for targeted (list-based) survey deployments. No collateral is typically associated with a survey campaign.

trading community architecture (TCA)

The trading community architecture is a customer model designed to support complex trading communities. The entire Oracle E-Business Suite uses the TCA customer model, providing a common model for customer data in all Oracle business applications. TCA strives to model *all* relationships within a trading community, supporting business-to-business (B2B) *and* business-to-consumer (B2C) models equally.

Uniform Resource Locator (URL)

A specific string of characters that identifies resources located on the Web. Also known as Uniform Resource Identifiers (URIs), the provide navigation to documents, images, downloadable files, services, electronic mailboxes, and other resources to Web browser users with one click. Full URLs include access methods or protocols such as HTTP or FTP, generally proceeded by a colon and two slashes, and then the full location. Nested directory levels are represented by additional slashes.

Web script

A Script Author script executed in a Web browser from an Oracle self-service Web application such as Oracle iSupport or Oracle iStore is referred to as a Web script. Like all scripts executed in the Scripting Engine Web interface, this requires as a prerequisite an active survey deployment. The URL for this deployment is then embedded in the self-service application as a hyperlink. When the end user selects the link, a new Web browser window opens, the user's current Oracle applications validation information is used to authenticate the script session, and the script executes in the new browser window. Web scripts are a valuable resource to provide scripted information to Web application users or to solicit feedback to the enterprise regarding the application user's experience.

wizard script

A wizard script is a script created using the Script Wizard feature of the Script Author, the authoring and development component of Oracle Scripting. Like graphical scripts, wizard scripts are executed in any Scripting Engine interface. At runtime, they display panels containing questions, answers, and graphics. Each panel includes a Continue button required to be clicked by the script end user at runtime to progress through each panel. The flow of the script is controlled by business rules built using the Script Wizard. Based on the choices of the script developer, the end user's flow through the script may be rigid (the same panels are seen regardless of how the user answers) or may change dynamically (demonstrating different flows based on the end user's responses). Wizard scripts can be listed, created, copied, edited, deployed, or converted to a graphical script only from the Script Wizard feature of the Script Author.