IBM

Introducing Application Development ToolSet for OS/400 and the AS/400 Server Access Programs

Version 3

AS/400 Advanced Series

IBM

Introducing Application Development ToolSet for OS/400 and the AS/400 Server Access Programs

Version 3

> **Note!**
>
> Before using this information and the product it supports, be sure to read the general information under "Notices" on page v.

Contents

Notices

Any reference to an IBM licensed program in this publication is not intended to state or imply that only IBM's licensed program may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594, USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independent created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Canada Ltd., Department 071, 1150 Eglinton Avenue East, North York, Ontario M3C 1H7, Canada. Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States or other countries or both:

400	OS/2
Application System/400	OS/400
AS/400	PROFS
COBOL/400	RPG
IBM	RPG/400
IBMLink	SQL
ILE	System/36
Integrated Language Environment	System/38
Operating System/400	VRPG Client

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

About This Book

This book describes Application Development ToolSet for OS/400 (ADTS for OS/400), and how the components of ADTS for OS/400 work together to make you more effective and productive.

This guide also introduces the server access programs that come with ADTS for OS/400. The server access programs provide host access to Application Development ToolSet Client Server for OS/400 (ADTS CS for OS/400), which consists of CoOperative Development Environment/400 (CODE/400) and AS/400 VRPG Client. To provide you with information to help you decide if you want to migrate to a workstation environment, this guide also provides you with an overview of how you can enhance your work environment using the workstation features.

This book does not describe in detail each component of the ADTS for OS/400 base or the CODE/400 and VRPG Client features. For details on performing specific tasks, refer to the manuals and online help provided for each component and product.

For more information about using the CODE/400 and VRPG Client features together, see *Introduction to Application Development ToolSet Client Server for OS/400*, GC09-2189. For a complete list of products related to ADTS for OS/400 or ADTS CS for OS/400, see the "Bibliography" on page 27. For a complete list of all the books in the AS/400 library, see the *Publications Reference*, SC41-4003.

Who Should Use This Book

This book is for application programmers or analysts working in an Application System/400 (AS/400) host environment who want learn more about ADTS for OS/400 as an integrated set of utilities, rather than as separate components.

This book is also for RPG application programmers who want to develop new applications with graphical user interfaces (GUI) and create client-server applications. If you are one of these readers, you may want to go directly to the last section of the book.

To learn more about each component or product, or to answer a question that is not addressed in this book, refer to the specific manual for the component or product. A list of manuals is provided in the Bibliography.

vii

Chapter 1. What Can I Do with Application Development ToolSet for OS/400?

The IBM Application Development ToolSet for OS/400 product (ADTS for OS/400) is a set of integrated host-based application development tools, plus the client server access programs that provide the capability for communicating with the client application development environments: Application Development ToolSet Client Server for OS/400 (ADTS CS for OS/400, which is a combination of CODE/400 and VRPG Client features).

This chapter describes the:

- Benefits of using ADTS for OS/400 for:

  - Improving the management of your application development activities
  - Speeding up application development

- Components of ADTS for OS/400

- Client server access programs that provide the host access to the companion workstation products

- Client components of the workstation products

Improving the Management of Application Development

With ADTS for OS/400 you can improve the management of application development in the following ways. You can:

- Manage and control development

  You can define a flexible environment using a library and version-control tool so that you can manage and easily develop production and maintenance versions of an application.

  You can organize application libraries into projects and groups to represent stages in the application development process. You can work simultaneously with different versions of the code and still be assured that you will not write over one another's changes. An audit log records the changes in the application, including the commands used to change the application's hierarchy or components, the names of those who issued the commands, and the time the changes were made.

- Determine where fields are used in your application

  You can assess the impact of a potential change to both Original Program Model (OPM) and Integrated Language Environment (ILE) applications. After creating a data dictionary for libraries or projects, you can identify the objects that will be affected by a planned change. When you make the change, the data dictionary is updated and the affected objects are recompiled.

  This function is especially useful, for example, when you change the content of a field that is used throughout an application.

- Determine the relationships among programs, objects, and files used by your applications

  You can determine the programs that are calling a specific program, and the programs that are being called by a specific program. For ILE programs, you

can also determine the ILE programs and service programs that were created from binding a specific service program or module, and modules and service programs that are bound into a specific ILE program or service program.

After you understand the connections among objects, it is easier to find and recompile objects affected by a change to your application, and it is easier to debug your large applications.

- Work with lists of projects, libraries, groups, objects, members, and parts

  Working with projects, libraries, groups, objects, members, and parts in lists is easy and fast. You can perform multiple operations on items in the lists by selecting menu options instead of typing control language (CL) commands.

- Automate compilation of applications

  When you compile or build applications, all the changed components of the application and its dependent files are recompiled. This eliminates run-time level checks.

- Compare members or parts and merge members or parts

  You can track changes to members and parts and produce reports that you can use to estimate the size of the merge or make audit trails. You can merge updates to applications when the same files are being updated by different developers. You can merge your customized changes to third-party applications with the next release of the applications in either interactive or batch mode.

Speeding Up Application Development

With the tools provided in the ADTS for OS/400 product, you can develop applications faster while performing the following tasks:

- Edit program source

  Record-level prompting helps you code your programs with fewer syntax errors. Automatic syntax-checking verifies your code before you compile it.

- Debug programs

  You can debug programs and set breakpoints to find logic errors faster.

- Design screens, menus, and reports

  What-you-see-is-what-you-get (WYSIWYG) tools help you design displays and menus without having to code using data description specifications (DDS).

- Test displays and create prototype reports

  You can easily test the displays you create, and create prototype reports to save you time when writing reports.

- Create physical and logical files

  You can save time and simplify a repetitive chore by creating files faster and easier using a utility instead of typing CL commands.

- Create and maintain Advanced Printer Function forms

  You can create and maintain forms using the special print capabilities available on some IBM printers. The forms appear as though they are preprinted or created from a printer with a variety of special fonts.

- Enter data in data files

  You can quickly and easily enter data in data files without writing programs to insert the data. This utility speeds the testing of programs to be used to update data files.

- Work with double-byte characters

  You can create and maintain double-byte character set (DBCS) characters and their sort sequences.

- Compile programs

  By selecting options from menus and lists, you can save time instead of entering CL commands on the command line. You can also compile and build objects from the same list you used for selecting objects to create or update.

Introducing the Components of Application Development ToolSet for OS/400

ADTS for OS/400 provides a set of integrated utilities and features to help you develop host applications. Using these tools, you can create applications faster and easier while managing their development. Figure 1 shows the application development environment you work within after ADTS for OS/400 is installed with both the Application Development Manager/400 and Application Dictionary Services/400 (AppDict Services/400) features. You can see how the utilities work together to help you develop your AS/400 applications.
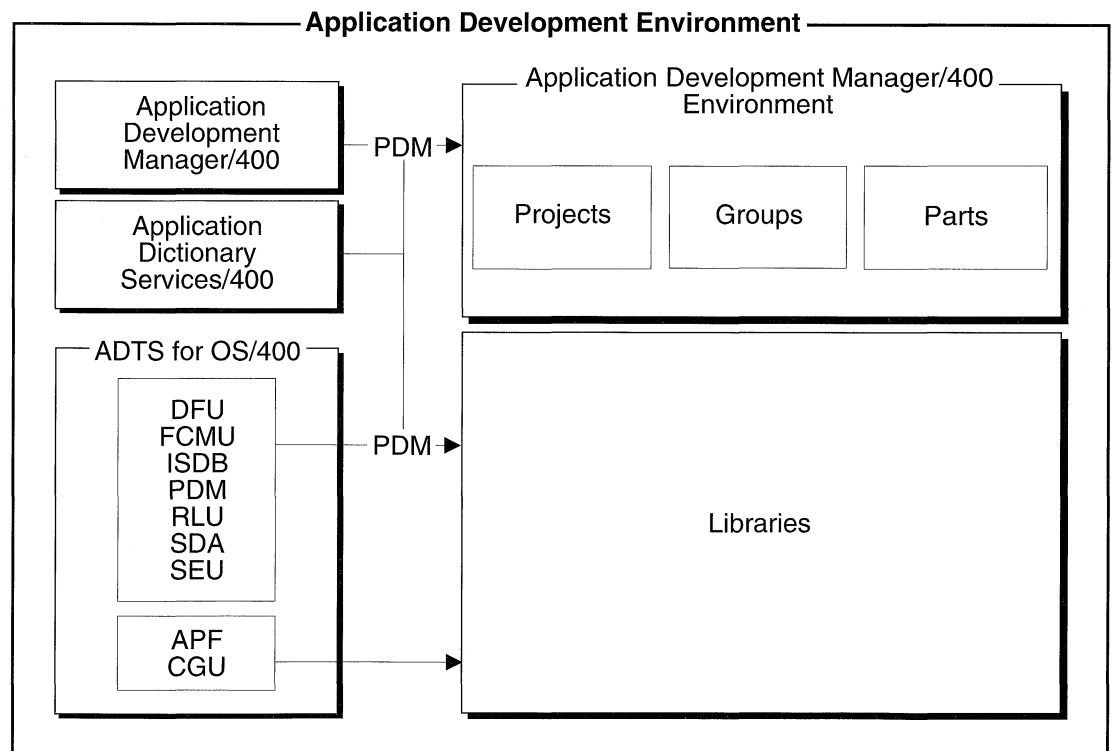

Figure 1. The Application Development Environment with Application Development ToolSet for OS/400

Base Utilities

The ADTS for OS/400 set of integrated utilities contains:

- Programming Development Manager (PDM)
- File Compare and Merge Utility (FCMU)
- Interactive Source Debugger (ISDB)
- Source Entry Utility (SEU)
- Data File Utility (DFU)
- Screen Design Aid (SDA)
- Report Layout Utility (RLU)
- Advanced Printer Function (APF)
- Character Generator Utility (CGU)

Two separately orderable features of ADTS for OS/400 are:

- Application Development Manager/400
- Application Dictionary Services/400

A brief description of each of the ADTS for OS/400 utilities follows.

Refer to the other manuals in the ADTS for OS/400 library for a more complete description of the features and functions of each utility.

Programming Development Manager (PDM)

The Programming Development Manager (PDM) utility is a list-driven shell for working with system and Application Development Manager/400 objects. You can easily run AS/400 system and ADTS for OS/400 commands against these listed objects.

With PDM, you can work with lists of libraries, objects, members, and Application Development Manager/400 projects, groups, and parts. From the lists, you can select objects and execute many commands against these objects, such as, copy, delete, and compile. Your productivity is increased because commands can be executed on more than one item at a time. You can also search multiple members or parts for a string.

You can create your own user-defined options, and use them to execute commands on libraries, objects, members, and Application Development Manager/400 projects, groups, and parts. You can also tailor your own defaults for your sessions.

PDM is your host-based tool for working with system and Application Development Manager/400 objects. You can also work with system and Application Development Manager/400 objects from a workstation using WorkFrame/2 in CODE/400.

PDM can also be used to interface with Cooperative Development Environment/400 (CODE/400) workstation commands, allowing you to launch workstation commands from the host. For more information, see the online document *CODE/400 Tips and Techniques* in the ADTS CS for OS/400 Information folder in CODE/400. Look under the topic "Communications".

Figure 2 illustrates the relationships between PDM and the other utilities:
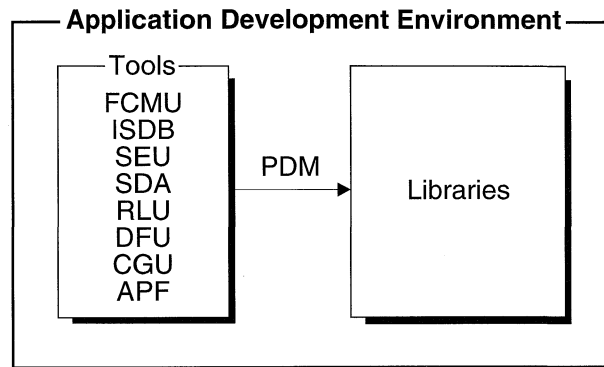
Figure 2. The Application Development ToolSet for OS/400 Utilities

File Compare and Merge Utility (FCMU)

The File Compare and Merge Utility (FCMU) automates the comparison of different versions of source code at the record, line, or word level. Using the compare facility, you can keep track of different iterations of source code.

You can compare:

- Differences only
- Differences within the entire file
- Differences within the context of the surrounding lines
- Final statistics of the comparison only

The compare facility also has many powerful options that increase its capabilities, such as:

- Language-sensitivity, which allows you to ignore things like RPG comments during a compare

- Location-specificity, which allows you to compare specific columns in one file with specific columns in another file

You can also merge different versions of the same source member. Use the Source Entry Utility (SEU) interface in interactive mode to do the merge, or submit the merge job to batch.

With the merge facility, you can:

- Merge selected maintenance updates in the target file member
- Merge all maintenance updates into the target file member
- Print a report without merging the updates

In addition to native source member support, FCMU is also integrated with Application Development Manager/400, offering you complete multi-version source control.

Interactive Source Debugger (ISDB)

This tool is a full-functioned host-based source debugger with an easy-to-use interface. You can use it to set conditional breakpoints to solve logic errors, and to capture your debugging commands in a log. Using ISDB to control your debugging session, you will be able to find logic errors more efficiently.

You can use ISDB to:

- Debug up to 10 programs simultaneously

- View the source code as you are debugging it

- Mark locations in your code with labels so that you can return to them later

- Set up to 50 breakpoints, including conditional breakpoints

- Display, monitor, and change the values of variables

- Execute a program up to a breakpoint or specified line, or step through a program line by line

- Search for text strings

- Debug programs written in original program model (OPM) languages, including RPG/400, COBOL/400, and CL/400

- Debug programs in batch mode

- Display the status of RPG indicators

ISDB is your host-based tool for debugging. You can also debug your OPM programs with a workstation-based interface using Debug Tool in a Cooperative Development Environment/400 (CODE/400) environment.

Source Entry Utility (SEU)

The Source Entry Utility (SEU) is comprised of a a full-screen editor that checks the syntax of your source code as you enter it, and a member list facility from which you can select the source members that you want to work with.

The main functions of SEU are:

- Syntax-checking function for language statements after they are entered through calls to language compilers

- A comprehensive set of line and global editing commands, including copy, delete, move, insert, exclude, hide, set, and find

- Editor profiles that can be maintained for each user

- Scan and substitute functions that can be used to locate text, even within other members

- A simultaneous scanning function that can be used to scan compiler listings while you are editing new source members

- Language-specific prompts and format lines to make entering source statements easier

- User-created prompts

- A simultaneous viewing function that can be used to view source members or spooled listings while you are editing or browsing another member

SEU is your host-based tool for editing source. You can also edit your source with a workstation-based interface using Live Parsing Extensible Editor (LPEX) in a Cooperative Development Environment/400 (CODE/400) environment.

Data File Utility (DFU)

The Data File Utility (DFU) is a program generator you can use to define, create, and maintain database applications that are suited to data entry or file maintenance. DFU can also be useful when you want to create test data for an application under development.

Using DFU, you can work with different file definitions, such as:

- RPG File (F) and Input (I) specifications
- Interactive Data Definition Utility (IDDU) definitions
- A file definition stored with a database file (Data Description Specifications)

DFU also supports the following tasks:

- Sequential, indexed, and direct file access
- Field validity checking
- Audit reporting
- Creation of customized application display files
- Use of the runtime support provided by the Operating System/400
- Updating of files without the need to create additional programs
- System38- and System/36-compatible DFU functions

Screen Design Aid (SDA)

The Screen Design Aid (SDA) utility is a tool for interactively designing, creating, and maintaining application displays, menus, and online help information. SDA automatically generates the Data Description Specifications (DDS) for display files.

SDA can help you prototype displays. You can use it to test newly created or updated display files immediately, so your customers can participate more easily in the design of the interface.

Using SDA you can:

- Define fields and constants for displays

- Select fields from existing database files

- Change the attributes and colors of fields and constants

- Move, copy, or delete fields or constants

- Display or change field conditions

- Display or hide a ruler

- Define extensive or context-sensitive help areas

- Define double-byte character constants and fields

- Print the screens used in your applications

SDA is your host-based tool for screen design. You can also design your screens with a workstation-based interface using Data Description Specifications Design Utility (DSU) in a Cooperative Development Environment/400 (CODE/400) environment.

Report Layout Utility (RLU)

The Report Layout Utility (RLU) is a full-screen editor you use to create and edit report images described using Data Description Specifications (DDS). You can use RLU to edit any member that contains printer file DDS.

You do not need to know a lot about DDS coding forms or syntax to use RLU. The report image looks like an actual listing produced by a high-level language application. RLU increases your productivity by letting you focus on the layout and contents of a report rather than on the coding of the DDS.

RLU can help you design reports in the following ways:

- After you create the layout (or image) of the report, RLU interprets the design and defines the DDS for you

- After you specify the record format and fields, RLU builds the report image

After you have created the report design and saved it as an RLU source member, you can use the RLU source member as a prototype, or you can compile it as a printer file and use it with your applications to print reports.

RLU is your host-based tool for developing reports. You can also create your host reports with a workstation-based interface using Data Description Specifications Design Utility (DSU) in a Cooperative Development Environment/400 (CODE/400) environment.

Advanced Printer Function (APF)

The Advanced Printer Function (APF) utility supports advanced printing functions on the IBM 5224 and 5225 printers. It generates forms, has alternate-character capability, and provides Optical Character Recognition (OCR) printing capability. You can also create logos, graphs, and bar codes. You can migrate your System/36 and System/38 APF form definitions and symbol sets, or use the comparable functions that are provided.

With APF, you can:

- Define alternative character sets and symbols (sample symbol sets are provided)

- Generate Universal Product Code (Version A and E), Code 39, EAN (European Article Number) 8 and 13, or PLESSEY bar codes

- Build and print logos and emblems

- Generate bar graphs

- Produce a customized form that can be merged with spooled application report data

Character Generator Utility (CGU)

The Character Generator Utility (CGU) allows you to define and maintain a set of double-byte characters on the AS/400 system. CGU also maintains sort information for each character defined.

CGU works with all four double-byte character set (DBCS) languages:

- Japanese
- Korean

- Simplified Chinese
- Traditional Chinese

You can install the sort and font tables for all four languages on your system, but you can only create, update, or delete characters in one language at a time—the language configured on your system. You can, however, copy and print characters from the other languages. When you are ready to create, update, or delete characters in another language, just reconfigure the system for that language.

Application Development Manager/400 Feature

The Application Development Manager/400 feature is an integrated part of ADTS for OS/400. It helps application developers to organize their work by providing version control and an intelligent build function for compiling updated parts. It eliminates the potential for problems such as missing programs that need to be compiled, or finding out that two or more programmers are making modifications to the same source members.

You can access Application Development Manager/400 from the AS/400 Programming Development Manager main menu options, or you can issue CL commands on the command line.

Application Development Manager/400 control enables you to organize libraries within projects. Within one project you can create groups that represent specific stages in the application development process. For example, you can create a test group and multiple development groups. Using this approach, you and other programmers can work simultaneously with different versions of the code without the risk of overwriting someone else's changes. An audit log registers changes in the application, the commands used to change the hierarchy or components in the hierarchy, the identification of the issuer of the commands, and the date when the activity occurs.

While you are working in Application Dictionary Services/400, you can also start a number of the components that are a part of Application Development ToolSet/400: the Source Entry Utility, the Screen Design Aid, the Report Layout Utility, the File Compare and Merge Utility, the Interactive Source Debugger, and the Data File Utility. This ability to access such components increases the efficiency of your development environment.

Distributing code to remote sites is also easy using Application Development Manager/400. You can distribute applications, application fixes, and application enhancements to one or more remote AS/400 systems using the Export Part (EXPPART) command. You can also receive parts from remote systems, using the Receive Part (RCVPART) command.

To summarize, an application development team using this feature can:

- Define a flexible environment in which production and maintenance versions of an application can be managed simultaneously.

- Organize several developers to work on the same application at the same time.

- Build or compile an application based on the components that have changed and the relationships among the components. The components are compiled in the correct order, ensuring that no level checks occur.

- Create and maintain several versions of an application.

- Distribute parts to, or receive parts from remote sites.

- Maintain an archive of up to five back-level versions of each source part.

- Manage lists of all of the parts that have changed for each fix or enhancement.

From the PDM interface, you can access objects that are stored either in libraries or Application Development Manager/400 projects. You can choose options to start tools from Application Development ToolSet/400 to work with projects, groups, and parts.

Application Development Manager/400 is also integrated with the AS/400 system and ADTS CS for OS/400. You can use Application Development Manager/400 to:

- Store VRPG Client applications in the Application Development Manager/400 projects on the AS/400 system. This integration gives VRPG Client the change management and version control capabilities of Application Development Manager/400.

- Build parts that you have created or changed using CODE/400.

Application Dictionary Services/400 Feature

The Application Dictionary Services/400 (AppDict Services/400) feature is an impact analysis tool that speeds up the analysis of applications. Programmers can quickly find out where changes should be made and what the changes should be, as well as the effect of these changes on other programs.

AppDict Services/400 stores the information about the application objects and their relationships to other application objects in a **dictionary**. A dictionary extracts the cross-reference information about all of the objects in an application's libraries, and saves it in a set of database files stored in a library. As application objects and their relationships to other objects change, *the information in the dictionary is automatically updated.*

AppDict Services/400 is integrated with other tools in Application Development ToolSet for OS/400, such as: Application Development Manager/400, Report Layout Utility (RLU), Screen Design Aid (SDA), and Source Entry Utility (SEU).

AppDict Services/400 is especially useful for maintaining and tracking the significantly larger number of program parts that are associated with creating programs in an Integrated Language Environment (ILE). ILE programs tend to use many more parts than traditional OPM programs because they:

- Are modular
- Can use multiple programming languages
- Reuse components, rather than duplicate code across programs

To design and maintain your application, you can use the Application Dictionary Services/400 feature as follows:

- To see where fields are used in the objects in your application, and what files and programs would be affected by a change to a field.

- To change a field and to ensure that the change occurs in all objects that refer to the field. AppDict Services/400 recompiles all files or programs that are affected by the change, or locates and recreates all ILE and service programs affected by the change, from previously compiled modules.

- To identify certain physical files as field reference files.

- To search for strings in a source member, or to scan for externally described field names in RPG/400 programs to learn whether a change to the field has real impact.

- To identify the programs that are calling a specific program, and the programs that are being called by a specific program.

- To identify the programs that bind a specific object, and the objects that are bound by a specific program.

- To create physical files based on field reference files without having to write DDS source.

- To build Structured Query Language (SQL) table files based on field reference files, and build SQL index files based on the SQL table files and physical files, while recompiling only the affected parts.

Using AppDict Services/400, you can record information about objects that reside in libraries or in Application Development Manager/400 groups. You can choose to record all the groups in a project or only selected ones. All of the AppDict Services/400 functions described above are available for Application Development Manager/400 groups except for the last two: creating physical files based on the field reference files, and building SQL tables and index files.

AppDict Services/400 provides a background job that uses system journaling support to automatically update the AppDict Services/400 dictionaries whenever the documented libraries are updated.

When you create, copy, delete, change, rename, or move objects (or check in, check out, promote, or import Application Development Manager/400 parts) inside or outside of the AppDict Services/400 environment, the AppDict Services/400 dictionaries are automatically updated. The information in the dictionaries is also updated with each action, thereby keeping it synchronized.

Application Dictionary Services/400 and PDM both provide lists of objects that you can change using other ADTS for OS/400 utilities.

Server Access Programs

The server access programs for ADTS CS for OS/400 are shipped with ADTS for OS/400. ADTS CS for OS/400 consists of the following two features:

- CODE/400
- VRPG Client

You can also purchase either of these two features separately, along with the base ADTS CS for OS/400 product.

The server access programs are found on the ADTS for OS/400 program tape, and, therefore, are automatically loaded on your system as part of the installation. These programs are the host component of CODE/400 and of VRPG Client. They are necessary because they:

- Contain communication code

- Contain source for initialization of the ADTS CS for OS/400 features

- Contain source for the *CODE/400 Self-Study Guide*

- Save you installation time when you install CODE/400 and VRPG Client

- Give you the flexibility to install CODE/400 and VRPG Client at any time

- Allow users with ADTS CS for OS/400 installed on a portable workstation to use the product on any AS/400 with ADTS for OS/400 installed

The server access programs contain communication code to allow a programmable workstation to access services and data that reside on the AS/400 host. They are the means by which CODE/400 and VRPG Client access the required AS/400 services and data.

As well, the server access programs are the programs required to start and run the host portion of these workstation features. These programs contain the source that initializes CODE/400 and VRPG Client. The server access programs do not provide you with CODE/400 and VRPG Client. You need to purchase these features separately, or together as the ADTS CS for OS/400 product.

If you are a programmer using CODE/400 and you want to work through the programs in the *CODE/400 Self-Study Guide*, you need the server access programs because they contain some of the necessary source code.

When you purchase CODE/400, VRPG Client, or both as ADTS CS for OS/400, you need only install the client portion of the product because the server access programs are installed when ADTS for OS/400 is installed. Thus, both CODE/400 and VRPG Client can be installed on your programmable workstation in one installation procedure with minimal disruption to your work.

Client Components of the Workstation Products

The client components of the workstation products are:

- Base product

    - Communication programs
    - Editor
    - WorkFrame/2

    Note: If want to purchase either of CODE/400 or VRPG Client separately, you also need to purchase the base ADTS CS for OS/400 product described here.

- CoOperative Development Environment/400 (CODE/400)

    - DDS Design Utility (DSU)
    - RPG/400 program verifier
    - Compile Interface
    - Debug Tool

- AS/400 VRPG Client (VRPG Client)

    - GUI Designer
    - Compiler
    - Debugger

For more information about ADTS CS for OS/400 and its features, refer to Chapter 3, "Workstation Development Environment for Host and Client-Server Applications" on page 21.

Chapter 2. Getting the Most Out of ADTS for OS/400

This chapter provides a high-level description of how to use the tools in the ADTS for OS/400 product to obtain optimal effectiveness and efficiency. It provides a description of:

- How to access the tools

- How the Application Development Manager/400 feature helps you store and keep track of programs

- How the Application Development Manager/400 feature helps you manage changes and multiple versions of applications

- Editing program source using SEU

- Compiling members using PDM

- Building parts using Application Development Manager/400

- Debugging programs using ISDB

- Comparing files using FCMU

- Merging files using FCMU

- Designing screens, menus, and reports using SDA and RLU

- Performing impact analysis using AppDict Services/400

- Working with lists

How the Components of ADTS for OS/400 Work Together

After both ADTS for OS/400 and Application Development Manager/400 are installed, you can access everything you need from the AS/400 Programming Development Manager main menu. Through options in PDM, you can access functions in other components, including Application Development Manager/400. PDM and Application Development Manager/400 together are the main working environment.

Figure 3 shows the AS/400 Programming Development Manager main menu with Application Development Manager/400 options:
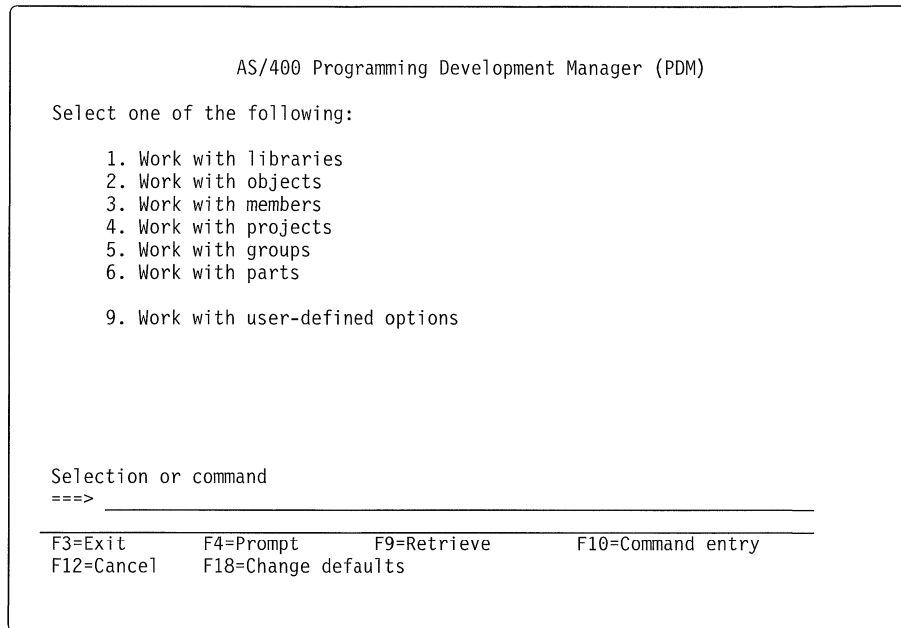
```
                    AS/400 Programming Development Manager (PDM)

    Select one of the following:

            1. Work with libraries
            2. Work with objects
            3. Work with members
            4. Work with projects
            5. Work with groups
            6. Work with parts

            9. Work with user-defined options




    Selection or command
    ===> _____

    F3=Exit       F4=Prompt       F9=Retrieve      F10=Command entry
    F12=Cancel    F18=Change defaults
```

Figure 3. The AS/400 Programming Development Manager (PDM) Main Menu

If Application Development Manager/400 is not installed, all the functionality of PDM is still available except for the control and management of versioned parts. That is, options 4, 5, and 6 do not appear when Application Development Manager/400 is not installed.

From member lists in PDM, you can directly access the Screen Design Aid, Source Entry Utility, Report Layout Utility, Interactive Source Debugger, Data File Utility, and the File Compare and Merge Utility. Also, by entering an option next to the listed member, you can direct PDM to call the appropriate CL command or the selected utility.

Storing and Keeping Track of Programs

Applications often consist of several modules, especially if they are ILE applications. The application development team's traditional sequence of create, edit and compile is enhanced using Application Development Manager/400 and Application Dictionary Services/400. Application Development Manager/400 manages the parts of the application programs. Application Dictionary Services/400 maintains data dictionaries to track changes made to parts. Parts have system-supplied types, such as RPGSRC, RPGINC, MODULE, or FILE, or they can have user-defined types. Parts can also have a system-supplied language, such as RPGLE or SQLRPG, or a user-defined language for a user-defined part type.

Before you can use Application Development Manager/400 to perform part-development tasks, a project hierarchy must exist. You must also be enrolled in the project and have update authority to a group. This requirement controls who can change an application.

For information on how to create project hierarchies and how to work with them, refer to the *ADTS/400: Application Development Manager/400 Introduction and Planning Guide*. It discusses the planning needed to take advantage of Application Development Manager/400 functions.

Figure 4 shows application development under the control of Application Development Manager/400 and comparable application development in the OS/400 environment.
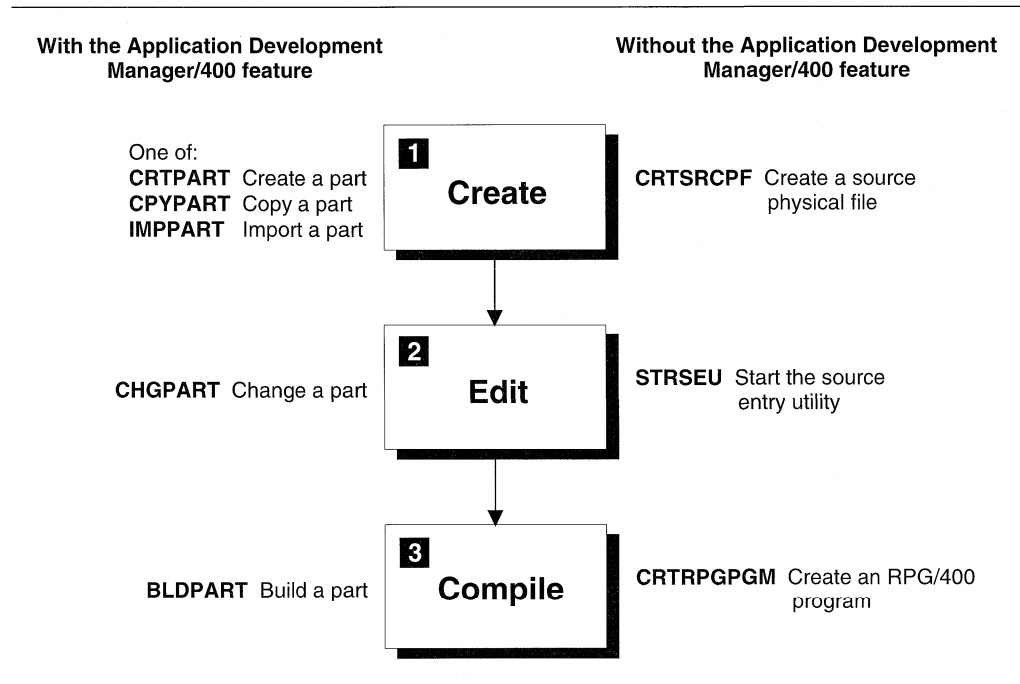

With the Application Development Manager/400 feature | **Without the Application Development Manager/400 feature**

One of:
CRTPART Create a part
CPYPART Copy a part
IMPPART Import a part

1 Create

CRTSRCPF Create a source physical file

CHGPART Change a part

2 Edit

STRSEU Start the source entry utility

BLDPART Build a part

3 Compile

CRTRPGPGM Create an RPG/400 program

Figure 4. Application Development Command Sequence

The left side of the figure shows the Application Development Manager/400 part-development commands you use. The right side of the figure shows the OS/400 commands you use if you are not developing parts under Application Development Manager/400 control.

Managing Changes and Multiple Versions of Applications

You can easily manage changes to one or more versions of your OPM or ILE application using Application Development Manager/400 and Application Dictionary Services/400. Impact analysis and data dictionaries help you to monitor changes. Source-code change management monitors part development and tells you when a change to a part should be made to a part in another version. Part builds are triggered after parts are changed. Application Development Manager/400 parts can also be accessed from CODE/400.

Application Dictionary Services/400 provides you with data dictionaries that are constantly being updated to reflect any changes you make to your libraries. With impact analysis, you can quickly identify the changes and the components they affect before you make the change. To proceed with the change, you use Application Dictionary Services/400 to access Application Development Manager/400 to check out the appropriate parts, or to access SEU if you are making changes to parts that have source members.

When working on Application Development Manager/400 projects, you use the check out function to prevent multiple developers from working on the same source at the same time. This function keeps program changes from being lost or over-written.

As your application changes over time, you can change the hierarchy of the Application Development Manager/400 projects to reflect these changes. For example, when work on the first version of an application is complete, development of a second version of the application can begin. A notify function lets you know when a part that you are requesting is already being changed in another version of the application, or when it exists in another branch of the project hierarchy. This is useful information when corrections to a production version of a part have to be propagated to the subsequent version of the part.

Another advantage of Application Development Manager/400 is the powerful build function, which compiles (or builds) the components of an application that have changed or that are affected by a change to another component. The build feature uses information generated by the compilers to compile objects in the correct order. It will compile the DDS source first, and then the programs. This feature eliminates level checks and missed compilation of programs.

Once all of your applications and application fixes have been built, they can easily be distributed to remote sites.

Editing Program Source

After you have checked out the part to be changed, you can edit its source using the Source Entry Utility (SEU). SEU is a full-screen editor that you can use to:

- Create and update the source members of a source file

- Edit Application Development Manager/400 parts

- List, edit, browse, print, or delete the existing members in a source physical file

- Use other editing functions, such as finding, changing, split-session editing and browsing, prompting, and syntax-checking

Compiling Members

To compile, you can use either of the following:

- The traditional method of typing the CL command on the command line

- PDM member lists

    You can choose the compile option by typing the option number next to the member you want to compile. PDM then submits a CL command depending on the member type. You can change the parameters before submitting the compile command by pressing F4 instead of the Enter key. PDM displays the CL prompt so you can alter the default parameters.

Building Parts

When you work with Application Development Manager/400 objects, you build parts instead of compile members. One of the advantages of Application Development Manager/400 is that while you build, Application Development Manager/400 recompiles all application parts that are affected by the changed parts.

After the build is completed, messages are displayed to indicate whether or not the part is built successfully. If the build is not successful, you look at the build report or compiler listing to find the errors. You can then use SEU to make the corrections. After you complete the corrections, you can then promote the parts through the project hierarchy.

If you change multiple parts, you can build the application as a whole rather than each part individually. The build report indicates whether or not the files that changed are compiled successfully.

Debugging Programs

After you code and compile your program, often you will need to debug it. The Interactive Source Debugger utility allows you to view your program interactively while the debug function is operating. You can set conditional breakpoints so you can track which conditions are triggered at which times. You can also set breakpoints that will stop the processing of the program if a specified statement has been encountered a specified number of times. This feature makes it easier to discover infinite loops in your code.

Comparing Files

If you are using multiple versions of a program, you may want to compare one copy of a file with another. The compare function of the File Compare and Merge Utility (FCMU) identifies the differences between two source physical files and performs the comparison at a level of detail which you determine. You can do the comparison at the record, line, or word level. A report is issued so you can print the results of the comparison.

Merging Source Files

When you customize programs that you purchase from a vendor, and then receive updates to that program, you are faced with the daunting task of integrating the licensed program changes with your customized code. There are many other scenarios when a merge of two sets of updates may be required.

With the merge function of the File Compare and Merge Utility, you can combine two sets of updates into one file.

You can perform the merge in batch or interactively. If you run the merge in batch the updates in the maintenance version are moved into the target version, and any conflicts that are found are logged in a report so that you can manually edit the member to your satisfaction. If you run the merge in interactive mode, you can choose which updates to keep (either your customized code or the changes in the updated program source). The Source Entry Utility is used in interactive mode so that you can edit your source directly, with the full function of SEU available to you.

Designing Screens, Menus, and Reports

You sometimes define screens, menus, and reports outside of your programs using externally described files. You use data description specifications (DDS) to do this.

SDA and RLU help you work with your DDS source in a WYSIWYG (what-you-see-is-what-you-get) manner. SDA helps you design menus and displays, and RLU helps you design reports.

These utilities offer many advantages over the traditional method of designing menus, displays, and reports using an editor, because you can:

- See what your displays, menus, and reports look like at the design and layout stage.

- Produce data description specifications (DDS) from your design.

- Design without an extensive knowledge of DDS coding forms, keywords, or syntax. Related keywords are grouped to help you make choices.

- Select fields from existing database files when designing a display or a report.

- Test displays with the data and status of the condition indicators specified for each test.

- Create prototype reports.

- Create a display from the DDS source statements that SDA creates, or a report from the DDS source statements that RLU creates.

- Change your screen design directly, without changing the DDS directly.

Performing Impact Analysis Before Making a Change

Before applying a change to your application, you can use Application Dictionary Services/400 to determine how many programs or files will be affected by the change. After you have a list of the programs that must be changed, you can use tools from ADTS for OS/400 to change them. The Application Dictionary Services/400 dictionaries are updated to record the change.

For step-by-step instructions on performing impact analysis using the Application Dictionary Services/400, refer to the *Application Dictionary Services/400 User's Guide*.

Working With Lists of Libraries, Objects, Members, Projects, Groups, Parts

The Programming Development Manager utility builds lists. Using lists to work from means that you can:

- Work with items in the list without typing in CL commands
- Work with several items in the list at the same time
- Create subsets of lists for faster retrieval and recognition
- Start other utilities to work with selected items
- Access Application Development Manager/400 through an interface

You do not need to know the exact syntax of CL commands that you want to use. For example, to compile an RPG program using PDM, you only need to type 14 (Compile) next to the source of the RPG program that you want to compile, and then press the Enter key, or on an OS/400 command line, you can type the Create RPG Program (CRTRPGPGM) command and then press F4 to enter all the required parameters. Using PDM, you can easily run multiple commands at the same time from any list. You can also create subsets of lists based on an object's attribute, such as its name or type.

PDM provides options to copy, delete, move, rename, compile, and recreate OS/400 objects. It also provides you with options for starting other host-based tools such as ISDB, FCMU, SEU, SDA, RLU, and DFU. Other functions, such as the Find string, are available from the same list. All PDM functions available to your object types can be used from the same list too. In addition to the options provided by PDM, you can define your own options. Refer to *ADTS/400: Programming Development Manager* for information on how to create them.

PDM also functions as an interface to Application Development Manager/400 functions, so you do not need to type CL commands on the command line. When you choose to work with projects, groups, or parts, you are presented with a list from which you can select options to perform any of the Application Development Manager/400 commands. All the other PDM functions, such as creating subsetted lists and user-defined options, are still available.

Chapter 3. Workstation Development Environment for Host and Client-Server Applications

This chapter describes Application Development ToolSet Client Server for OS/400 (ADTS CS for OS/400) including the following:

- Common components of CODE/400 and VRPG Client
- Features and benefits of CODE/400
- Components in CODE/400
- Features and benefits of VRPG Client
- Components in VRPG Client

When you purchase ADTS CS for OS/400, you are getting CODE/400 and VRPG Client in one package. You can buy either of these features separately as well, along with the base ADTS CS for OS/400 product.

ADTS CS for OS/400 provides a workstation-based development environment for maintaining current AS/400 applications and for creating new client-server applications. ADTS CS for OS/400 runs on the OS/2 operating system on your workstation and contains the CODE/400 and AS/400 VRPG Client features.

Working in the OS/2 environment allows you to take full advantage of OS/2's multitasking capabilities. Working on a GUI desktop allows you to manipulate objects faster and easier than by using commands. And, of course, working on the desktop means that host resource is freed for more demanding activities, such as network and application processing.

The CODE/400 feature lets you create applications on the workstation that will run on the host, and lets you maintain current applications. The VRPG Client feature lets you create new GUI client-server RPG applications in a visual application development environment that can only be run on the workstation.

For an explanation of how you can use CODE/400 and VRPG Client together, see *Introduction to Application Development ToolSet Client Server for OS/400.* You receive this book when you purchase ADTS CS for OS/400.

Figure 5 on page 22 shows your application development environment with ADTS CS for OS/400 installed, that is, with both the CODE/400 and VRPG Client features installed. It also illustrates the relationships between ADTS CS for OS/400 and the components of ADTS for OS/400.
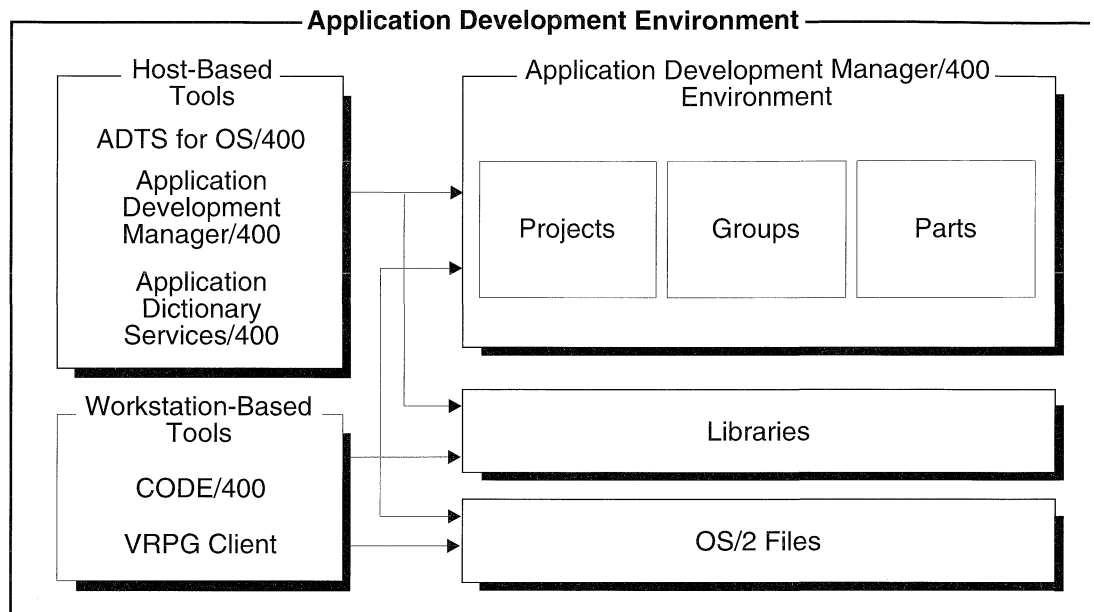
```
┌─────────────────── Application Development Environment ───────────────────┐
│  ┌──── Host-Based ────┐          ┌── Application Development Manager/400 ──┐ │
│  │        Tools       │          │            Environment                  │ │
│  │  ADTS for OS/400   │          │  ┌──────────┐ ┌──────────┐ ┌──────────┐ │ │
│  │  Application       │─────────▶│  │          │ │          │ │          │ │ │
│  │  Development       │          │  │ Projects │ │  Groups  │ │  Parts   │ │ │
│  │  Manager/400       │          │  │          │ │          │ │          │ │ │
│  │  Application       │          │  └──────────┘ └──────────┘ └──────────┘ │ │
│  │  Dictionary        │          └─────────────────────────────────────────┘ │
│  │  Services/400      │                                                       │
│  └────────────────────┘          ┌─────────────────────────────────────────┐ │
│  ┌─ Workstation-Based ┐          │                Libraries                 │ │
│  │        Tools       │─────────▶│                                          │ │
│  │  CODE/400          │          └─────────────────────────────────────────┘ │
│  │                    │          ┌─────────────────────────────────────────┐ │
│  │  VRPG Client       │─────────▶│                OS/2 Files                │ │
│  └────────────────────┘          └─────────────────────────────────────────┘ │
└───────────────────────────────────────────────────────────────────────────────┘
```

Figure 5. Application Development Environment with Workstation Tools

Common Components of CODE/400 and VRPG Client

When you install ADTS CS for OS/400, you install components that CODE/400 and VRPG Client share. These components are the Editor, WorkFrame/2, and the communication mechanism between the workstation and the host. These shared components provide a common development environment for both host and client-server applications. With these components you can:

- Program in RPG, both on the client and the server.

- Use the same high-function editor for the development of server or client applications.

- Use WorkFrame/2 and its predefined action profiles. Tools for both host application development (editor, debugger, and designer for data files, screens, and reports) and client application development (GUI designer, editor, and client debugger) can be initiated from this user-friendly WorkFrame/2 development environment.

- Improve the response time of applications because they run on the client instead of the server.

CODE/400

CODE/400 combines the productivity of the PC environment with the integrity of the AS/400 environment for developing applications.

Features

CODE/400 is an integrated and cohesive edit, compile, and debug utility. It runs on a programmable workstation using the graphical user interface of OS/2, and offloads processing time from the AS/400 host.

CODE/400 provides:

- A language-sensitive editor with prompting and color tokenizing that highlights different language constructs.

- A source level debug tool with many advanced debugging functions.

- A utility (Data Description Specifications Design Utility - DSU) to help you create data description specifications (DDS).

- The ability to compile, find errors, correct errors, and re-compile without leaving the editor.

- An interface to host compilers.

- An RPG/400 program verifier that checks for syntax and semantic errors.

- Language support for Integrated Language Environment (ILE) RPG/400, ILE RPG/400 SQL, ILE CL, original program model (OPM) RPG/400, OPM RPG/400 SQL, OPM CL, OPM COBOL/400, ILE COBOL/400, OPM COBOL/400 SQL, and ILE COBOL/400 SQL.

- WorkFrame/2.

- The ability to edit and verify program files on a workstation in disconnect mode. Later, these files can be uploaded to the host for compiling and debugging.

- An interface to Application Development Manager/400.

Benefits

Using CODE/400 you can:

- Take full advantage of the OS/2 desktop, including multitasking, mouse and icon manipulation, windowing, and simultaneous connection to multiple hosts

- Reduce system load on the AS/400 machine

- Find syntax and logic errors before compile time

- Reduce application development cycle time

- Write and verify code without having to be connected to a host

Editing with CODE/400

The editor offers several functions designed to optimize coding efficiency for each language being edited, before the compiler is used.

With the editor you can:

- Use visual aids for code structure
- Customize and program its functions
- Use interactive syntax checking
- Detect errors earlier in the coding cycle
- Use concurrent multiple edit sessions with cut and paste capabilities
- Gain rapid access to online language-sensitive help
- Display only particular types of lines

Compiling with CODE/400

You can verify the syntax and semantics of program source on your workstation using the RPG/400 program verifier before sending the source to the host compiler.

Debugging with CODE/400

The Debug Tool provides functions for debugging, tracing, setting breakpoints, monitoring, and changing the values of variables.

To follow the execution of the program or quickly identify errors in code, you can monitor source code or compiler listings on the workstation while the program is running on the host. While you are debugging your program, you can change the source of the program in the editor.

DDS Design Utility Functions

The DDS Design Utility (DSU) feature helps you develop display files, printer files, and database definition files for use on the AS/400 system. DSU has full DDS graphical keyword support.

The DSU feature provides an interface for designing each type of file, produces DDS source based on the design, and optionally calls the OS/400 DDS compiler. A graphical interface helps you concentrate more on the design of displays and reports than on the DDS coding specifics. DDS errors can be checked anytime without compiling the source.

WorkFrame/2

WorkFrame/2 lets you work with OS/400 objects on your OS/2 desktop. It lets you manipulate OS/400 libraries, files, and members, or Application Development Manager/400 projects, groups, and parts using the object-oriented user interface of OS/2.

VRPG Client

VRPG Client offers a development environment for RPG application programmers to develop, maintain, and document their applications visually. You can preserve your skills and develop visual RPG applications with the graphical interface and logic on the programmable workstation. A function to import display files and convert them to VRPG GUI definitions is also provided.

Two of the components in VRPG Client are shared with CODE/400: VRPG Client uses the same editor as CODE/400 and also uses WorkFrame/2.

Features

VRPG Client has the following features:

- Ease of use and minimal learning curve
- GUI designer
- Help creation using Information Presentation Facility (IPF)
- Message creation
- RPG workstation compiler for client code
- Integrated language-sensitive editing support
- AS/400 database (DB2 for OS/400) access
- Powerful RPG source debugger

- Built-in communication to AS/400 system
- Application modernization support
- Host backup and recovery with workstation development
- Packaging user applications for both OS/2 and Windows 3.1

Benefits

With VRPG Client you can:

- Preserve your skills in RPG development
- Enhance your skills in RPG development with GUI applications
- Take full advantage of the OS/2 desktop
- Reduce system load on the AS/400 machine

Using the VRPG Client GUI Designer

The Graphical User Interface (GUI) Designer is similar to Screen Design Aid (SDA) and to DDS Design Utility (DSU). The GUI definitions are created for you from a visual interface that you manipulate.

Also, just as when using SDA, you have access to the AS/400 database. You can easily access referenced fields, and pick up attributes from the database.

You can import your host display files to convert them to a GUI. Though some adjustments will be necessary to complete the conversion, the import function saves time when you convert applications.

Using the VRPG Compiler

The RPG workstation compiler provides event-driven programming support. The language definitions allow you to create action subroutines that you can start at run time when an event is triggered through the user interface. New operation codes are available to change attributes of controls dynamically, and to obtain status information from attributes of controls at run time.

Storing Your VRPG Client Applications as Application Development Manager/400 Parts

The Application Development Manager/400 feature allows you to store and manage VRPG Client applications in Application Development Manager/400 projects on the AS/400 system. To do this, two part types are available: VRPGTXT and VRPGBIN.

VRPGTXT part types are stored in source physical files with the language of the part type being VRPG. VRPG Client text objects are stored as members with the member types set as their workstation file extension.

VRPGBIN part types are stored in data files with the language of the part being VRPG. This type of part includes VRPG Client binary objects.

Bibliography

The following publications are listed with their full titles and base order numbers.

The related Application Development ToolSet for OS/400 publications are:

- *ADTS/400: Advanced Printer Function*, SC09-1766
- *ADTS/400: Character Generator Utility*, SC09-1769
- *ADTS/400: Data File Utility*, SC09-1773
- *ADTS/400: File Compare and Merge Utility*, SC09-1772
- *ADTS/400: Interactive Source Debugger*, SC09-1897
- *ADTS/400: Programming Development Manager*, SC09-1771
- *ADTS/400: Report Layout Utility*, SC09-1767
- *ADTS/400: Screen Design Aid*, SC09-1768
- *ADTS/400: Screen Design Aid for the System/36 Environment*, SC09-1893
- *ADTS/400: Source Entry Utility*, SC09-1774
- *LPS: IBM Application Development ToolSet/400*, GC09-1884

The publications for the orderable features of Application Development ToolSet for OS/400 are:

- *ADTS/400: Application Development Manager/400 Introduction and Planning Guide*, GC09-1807
- *ADTS/400: Application Development Manager/400 User's Guide*, SC09-2133
- *ADTS/400: Application Development Manager/400 API Reference*, SC09-2180
- *ADTS/400: Application Development Manager/400 Self-Study Guide*, SC09-2138
- *ADTS/400: Application Dictionary Services/400 Self-Study Guide*, SC09-2086
- *ADTS/400: Application Dictionary Services/400 User's Guide*, SC09-2087

The related Application Development ToolSet Client Server for OS/400 publications are:

- *CODE/400 General Information*, GC09-1907
- *CODE/400 Debug Tool*, SC09-1905
- *CODE/400 Self-Study Guide*, SC09-1911
- *Getting Started with VRPG Client*, SC09-2195
- *Installing ADTS CS for OS/400*, SC09-2188
- *Introduction to Application Development ToolSet Client Server for OS/400*, GC09-2189
- *LPEX Command Reference*, SC09-1910
- *Publications Reference*, SC41-4003
- *VRPG Client Language Reference*, SC09-1847
- *VRPG Client Parts Reference*, SC09-1846
- *Getting Started with VRPG Client*, SC09-2195
- *Client/Server Visual Programming with VRPG Client*, SC09-2131

Index

A

ADTS CS for OS/400
  See Application Development ToolSet Client Server for OS/400 (cl1s.)
ADTS for OS/400
  See Application Development ToolSet for OS/400 (adts44.)
ADTS/400
  See Application Development ToolSet/400 (ADTS)
Advanced Printer Function (APF)   8
APF
  See advanced printer function (APF)
AppDict Services/400
  See Application Dictionary Services/400 (AppDict Services/400)
application development
  DDS   2
  debugging   2
  designing screens, menus, reports   2
  improving management of   1
  speeding up   2
  using Application Dictionary Services/400   10
Application Development Manager/400
  accessing   9
  build function   16
  check out function   15
  compiling applications   16
  description   9
  maintaining application versions   9
  managing change   15
  notify function   16
  storing VRPG Client applications with   25
  uses   9
  working with AppDict Services/400   11
  working with PDM   19
Application Development ToolSet Client Server for OS/400 (ADTS CS for OS/400)
  common components   22
  introduction   21
Application Development ToolSet Client Server for OS/400 (cl1s.)
  components   12
Application Development ToolSet for OS/400 (ADTS for OS/400)
  components
    APF   8
    Application Development Manager/400   9
    Application Dictionary Services/400   10
    CGU   8
    DFU   7
    FCMU   5
    ISDB   6

Application Development ToolSet for OS/400 (ADTS for OS/400) *(continued)*
  components *(continued)*
    overview   3
    PDM   4
    RLU   7
    SDA   7
    SEU   6
    working together   13
  environment   13
  features   4
  figure   3
  function   1
  integration   13
  server access programs   11
  utilities   4
Application Dictionary Services/400 (AppDict Services/400)
  building SQL tables   11
  creating data dictionary   10
  data dictionaries   14, 15
  description   10
  finding relationships between objects   10
  functions of   10
  identifying field reference file   10
  impact analysis   10, 15
  performing impact analysis   18
  searching for string   10
  using   10
  working with Application Development Manager/400   11, 14, 15
AS/400 VRPG Client (VRPG Client)
  application storage with Application Development Manager/400   25
  benefits   25
  common components   22
  features   24
  graphical user interface (GUI)   24, 25
  help and message compiler   24
  VRPG compiler   25
  VRPGBIN part types   25
  VRPGTXT part types   25
automating compilation of application   2
automating compilation of applications   16

B

benefits
  CODE/400   23
breakpoint
  conditional   17
  uses   17

Communicating Your Comments to IBM

AS/400 Advanced Series
Introducing Application Development
ToolSet for OS/400 and the
AS/400 Server Access Programs
Version 3

Publication No. GC09-2088-00

If there is something you like—or dislike—about this book, please let us know. You can use one of the methods listed below to send your comments to IBM. If you want a reply, include your name, address, and telephone number. If you are communicating electronically, include the book title, publication number, page number, or topic you are commenting on.

The comments you send should only pertain to the information in this book and its presentation. To request additional publications or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give it to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:
  - United States and Canada: 416-448-6161
  - Other countries: (+1)-416-448-6161
- If you prefer to send comments electronically, use the network ID listed below. Be sure to include your entire network address if you wish a reply.
  - Internet: torrcf@vnet.ibm.com
  - IBMLink: toribm(torrcf)
  - IBM/PROFS: torolab4(torrcf)
  - IBMMAIL: ibmmail(caibmwt9)

Readers' Comments — We'd Like to Hear from You

AS/400 Advanced Series
Introducing Application Development
ToolSet for OS/400 and the
AS/400 Server Access Programs
Version 3

Publication No. GC09-2088-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____   Address _____

Company or Organization _____   _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
GC09-2088-00

IBM®

Fold and Tape   **Please do not staple**   Fold and Tape

PLACE
POSTAGE
STAMP
HERE

IBM Canada Ltd. Laboratory
Information Development
2G/345/1150/TOR
1150 EGLINTON AVENUE EAST
NORTH YORK   ONTARIO   CANADA   M3C 1H7

Fold and Tape   **Please do not staple**   Fold and Tape

GC09-2088-00

Readers' Comments — We'd Like to Hear from You

AS/400 Advanced Series
Introducing Application Development
ToolSet for OS/400 and the
AS/400 Server Access Programs
Version 3

Publication No. GC09-2088-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____   Address _____

Company or Organization _____

Phone No. _____

Fold and Tape   **Please do not staple**   Fold and Tape

PLACE
POSTAGE
STAMP
HERE

IBM Canada Ltd. Laboratory
Information Development
2G/345/1150/TOR
1150 EGLINTON AVENUE EAST
NORTH YORK   ONTARIO   CANADA   M3C 1H7

Fold and Tape   **Please do not staple**   Fold and Tape

Readers' Comments — We'd Like to Hear from You

GC09-2088-00

IBM ®

Cut o
Along

GC09-2088-00

Cut c
Along

)

)

)

IBM

Program Number: 5716-PW1