Wired for Management: Managed Objects ToolKit

Intel Corporation

September 30, 1997

intel®

*Third-party brands and names are the property of their respective owners

Agenda

- What is the ToolKit?
- Why use the ToolKit?
- How to use the ToolKit
  - DMI 2.0 Example and demo
- ToolKit controls
- ToolKit road map

*Third-party brands and names are the property of their respective owners

What Is the ToolKit?

- A free collection of ActiveX* controls that access WfM technologies which allow the user to create management applications quickly

*Third-party brands and names are the property of their respective owners

What Is the ToolKit?

What Can You Do With It?

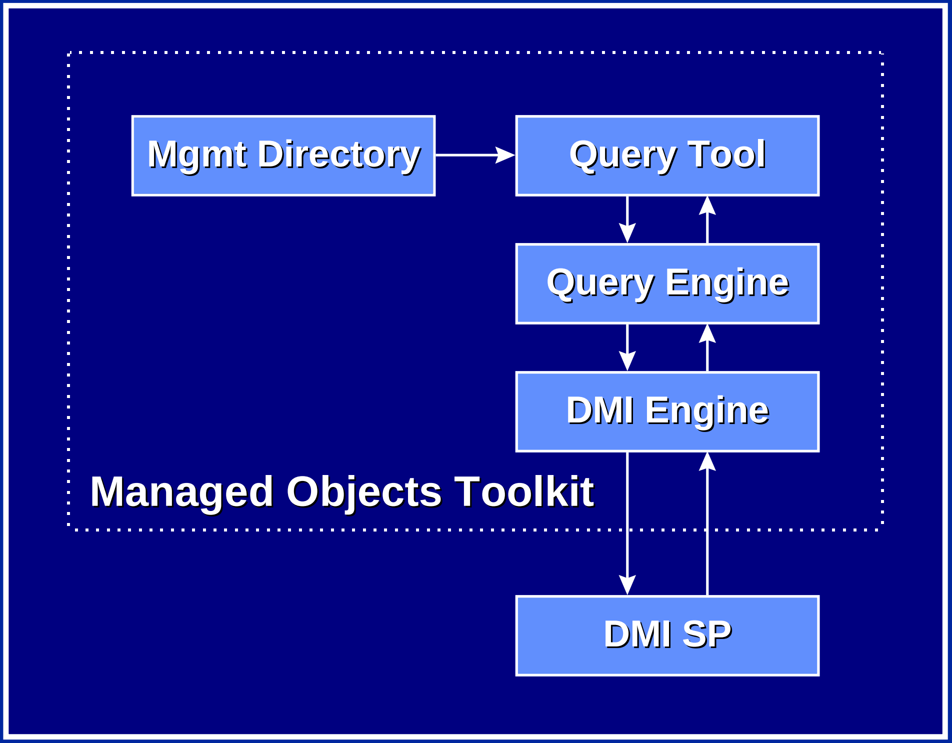
- Quickly create:
  - WfM information browsers
  - Alert processing tools
  - Instrumentation-specific tools
  - Applets which integrate with existing management frameworks

*Third-party brands and names are the property of their respective owners

What Is the ToolKit?

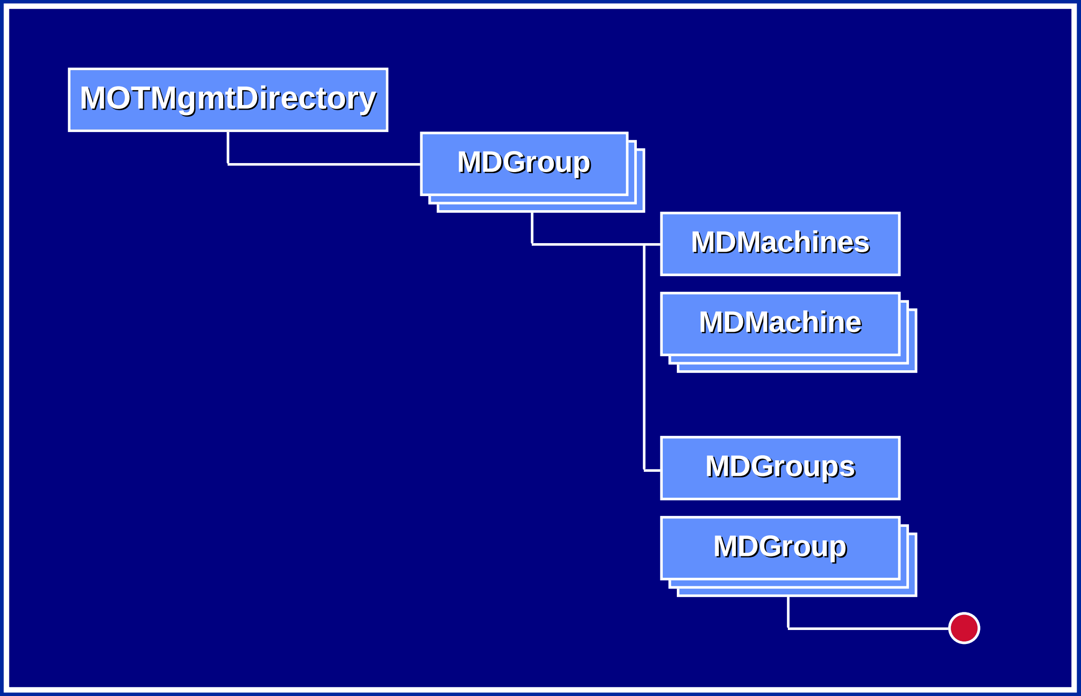
Current ToolKit Contents

- Management directory
- Query tool
- Query engine
- DMI engine



*Third-party brands and names are the property of their respective owners

Management Directory



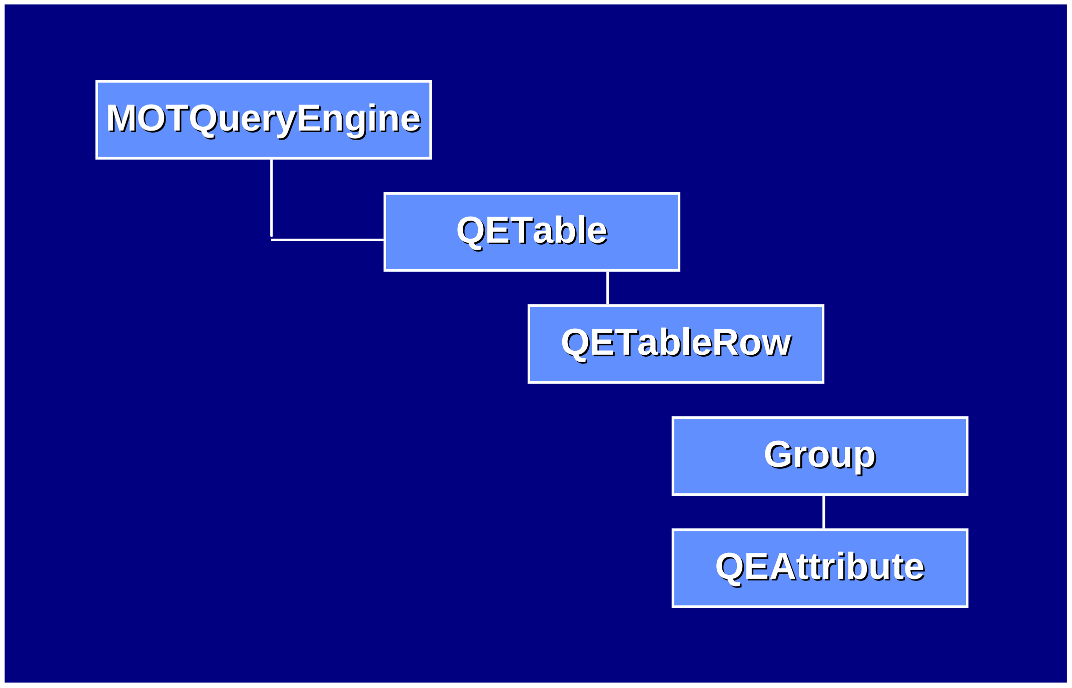
- Provides a simple way to visually model the manageable network
- Allows discovery to be attached

*Third-party brands and names are the property of their respective owners

Query Tool

- Purpose
  - Allow user to graphically build complex queries on multiple remote DMI-enabled machines
- Features
  - OLE drop target for management nodes
  - Create persistent queries
  - Drag results to an OLE drop site

Query Engine



- Provides way to build complex queries across a network

*Third-party brands and names are the property of their respective owners

Why Use the ToolKit?

- Quick and easy access to WfM technology
- Shields users from changes in technology
- Integrated with most development environments and applications

Saves a lot of time!

intel®

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

- Determine the task
- Choose your environment
    - C/C++*, VB*, VBA*, VBScript*, Java*, Jscript*
    - Browsers, Office Suites, Dev Studios
- Develop solution with ToolKit

*Third-party brands and names are the property of their respective owners

Demonstration

Using the ToolKit to Query Multiple Platforms via DMI 2.0

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

DMI 2.0 Example

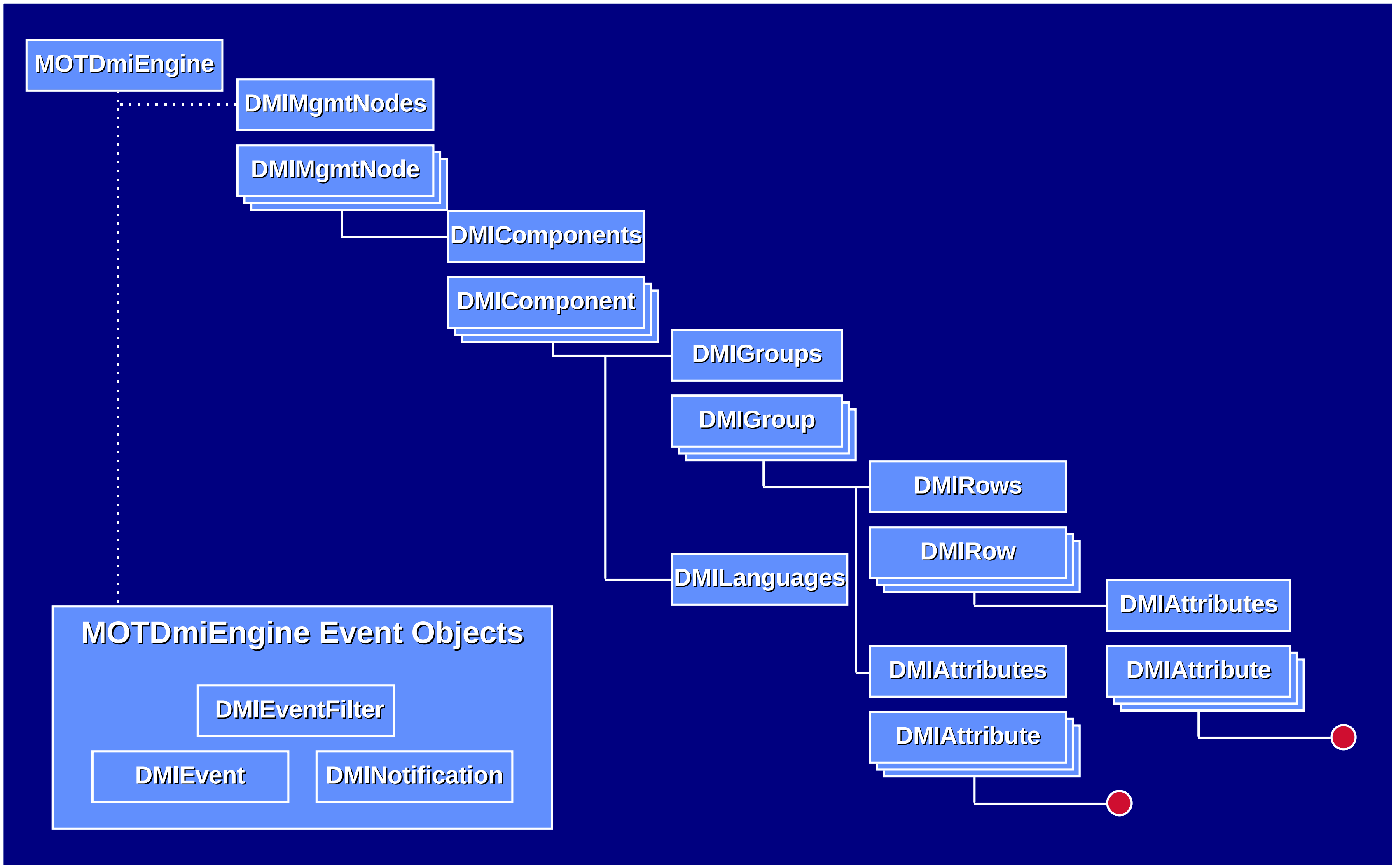
- Describe DMI engine design
  - DMI data objects
  - DMI data object collections
  - DMI data object persistence
  - DMI events

Four subsystems provide DMI functionality

intel®

*Third-party brands and names are the property of their respective owners

DMI Engine Hierarchy



intel®

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

DMI Data Objects

- DMIMgmtNode
- DMIComponent
- DMIGroup
- DMIRow
- DMIAttribute

- Read() method allow random access to remote DMI 2.0 service provider

Models DMI 2.0 Data Model

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

DMI Data Object Example

```
Public Const RFSP                    2        // read from SP
Dim oComp as New DMIComponent             // object
Dim oRow as New DMIRow                    // object

// read method connects to SP and gets object
oComp.Read(“dce|tcpip|jdoe|12”, RFSP )
oRow.Read(“dce|tcpip|jdoe|12|5|1=“9”+3=“FOO””,
RFSP)
```

*Third-party brands and names are the property of their respective owners

Example

```
// Access objects from collections
Dim oA as DMIAttribute
Set oA = oMgmt.Components(1).Groups(4).Rows(1).Attributes(1)

Set oA = oMgmt.Components(“WIN DMI Service
     Layer”).Groups(“DMTF|ComponentID|001”).Rows(1).
     Attributes(“Manufacturer”)

oA.Value = “MyCorp”   // sets attribute on SP
```

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

DMI Data Object Collections

- Every DMI data object has a collection
  - E.g., DMIGroup → DMIGroups collection
- Collection methods and properties
  - Add(), Remove(), RemoveAll()
  - Count(), Item()[†]

[†] Default property

intel®

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

Collections Example

```
Dim oMN as New DMIMgmtNode    // define objects
Dim oComp as DMIComponent

oMN.Connect( “dce|tcpip|jdoe”)          // conn to SP
For each oComp in oMN.Components        // enum
comps
    MsgBox oComp.Name
Next
```

Easy to enumerate objects

*Third-party brands and names are the property of their respective owners

Example

```
// Add a component, group, & row
Dim oRow as New DMIRow
// build a valid row; user implemented func
BuildRow oRow

oMgmtNode.Components.Add(“C:\CP.MIF”)
oMgmtNode.Components(1).Groups.Add(“C:\GP.MIF”)
oMgmtNode.Component(1).Groups(1).Rows.Add oRow
```

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

DMI Data Object Persistence

- Every DMI data object can read and write its internal state to a file
- Each object supports a Read() and Write() method

Good for inventory
and asset control

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

Persistence Example

```
// read object from SP, write object to file

Const Public      RFSP      2
oGrp.Read("dce|tcpip|jdoe|6|5", RFSP)
oGrp.Write( "c:\group5.fil")
```

*Third-party brands and names are the property of their respective owners

How to Use the ToolKit

DMI Events

- Two types:
  - DMIEvents - Std Events
  - DMINotifications - Add/Remove of DMI Data Objects
- Easy to use

*Third-party brands and names are the property of their respective owners

Example

```
// Listen for events and notifications
// Container automatically generates entry points

MOTDmiEngine1.EnableEvents oMgmtNode
MOTDmiEngine1.EnableNotifications oMgmtNode

DoEvents

MOTDmiEngine1.DisableEvents oMgmtNode
MOTDmiEngine1.DisableNotifications oMgmtNode
```

*Third-party brands and names are the property of their respective owners

Future Direction

- Possible next set of WfM tools
  - Meta Alert Monitor Control
  - Service Incident Control
  - Service Boot Admin Control
  - WfM Discovery Control
- Possible Java and CIM tools to come

Additional WfM technologies to come!

*Third-party brands and names are the property of their respective owners

Summary

- Collection of WfM ActiveX controls
- Saves significant time
- Controls interoperate
- Easy to use
- Help design the future via feedback

*Third-party brands and names are the property of their respective owners

Call to Action

- Get the WfM Managed Objects ToolKit
- Use it to access WfM information
- Create management applications
- Send feedback to:
    - wfm_tool_kit@ccm.jf.intel.com

intel®

*Third-party brands and names are the property of their respective owners

Collateral

- DMI 2.0 specification
- Managed PC Web site
  - http://www.intel.com/managedpc
- WFM SDK

intel®

*Third-party brands and names are the property of their respective owners