404025

IBM @server iSeries

Operations Navigator Plug-In Development

Greg McGuire
mcguireg@us.ibm.com

IBM @server. For the next generation of e-business.

Overview

IBM @server iSeries

- What does iSeries Navigator Plug-in Support offer me?
  - Integrating into iSeries Navigator
  - Using iSeries Navigator plug-in interfaces and services
- How can I build a Plug-in?
  - Installing and setting up of the tools, JDKs, classpath, etc.
  - Developing your Graphical User Interface (GUI)
  - Developing your Application
  - Getting and Putting data to the AS/400
  - Plugging your app into iSeries Navigator via the registry
  - Debugging
  - Installing your plug-in
- Where can I get more information?

IBM @server. For the next generation of e-business.

What does iSeries Navigator Plug-in Support offer?

IBM @server iSeries

- **Seamless Integration** into the iSeries GUI
  - Multiple integration points into iSeries Navigator
  - iSeries Naviator is the direction of the iSeries
- **Built-in Installation and Distribution** for your application
  - Uses iSeries Access for Windows install from the iSeries file system
- **Easier Management** using iSeries Access for Windows services
  - Installing fixes and code upgrades
  - Uninstalling the application
  - Using multi-national language support
- **Flexible Options** for building your plug-in
  - Java, C++, and Visual Basic plug-in support in Op Nav
  - This presentation focuses on Java as the preferred method

IBM @server. For the next generation of e-

Some Background Information

IBM @server iSeries

- iSeries Navigator is shipped as part of iSeries Access for Windows at no additional charge.
- Toolbox for Java is API's, GUIs and tools for connecting to the iSeries via Java and is shipped with iSeries Access for Windows
- Business Partners may provide **Plug-ins**
  - set of resources, classes, interfaces, and other "cross-references" that allow OpNav to be extended without changing code
  - Allows third parties to build applications without knowing internals
    - Plug-ins can be C++, Java or VisualBasic, however, Java is recommended

IBM @server. For the next generation of e-business.

Plugin Development

IBM @server iSeries

- Can take a staged approach to development
  1. Good first step is to create a new context menu that launches your application
     - Easier to do, not as tightly integrated
  2. Then add Application to Op Nav to make it more visible

IBM @server. For the next generation of e-business.

What does Op Nav Plug-in Support offer?

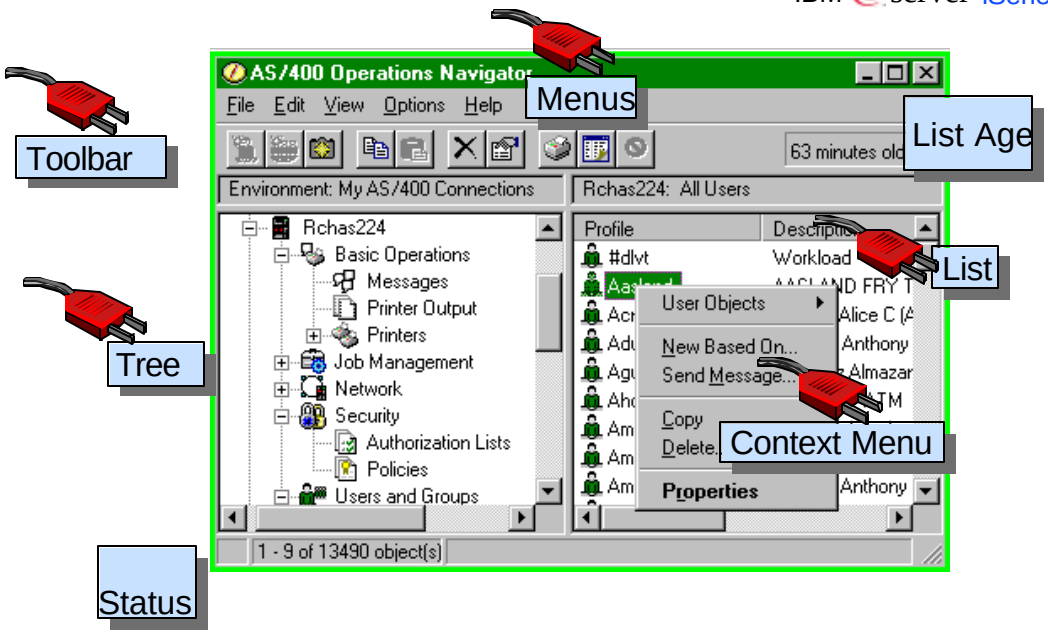
IBM @server iSeries

Seamless Integration

IBM @server. For the next generation of e-business.

OpNav User Interface Elements

IBM server iSeries



IBM @server. For the next generation of e-business.

User Interface Characteristics of iSeries Navigator (notes)

IBM server iSeries

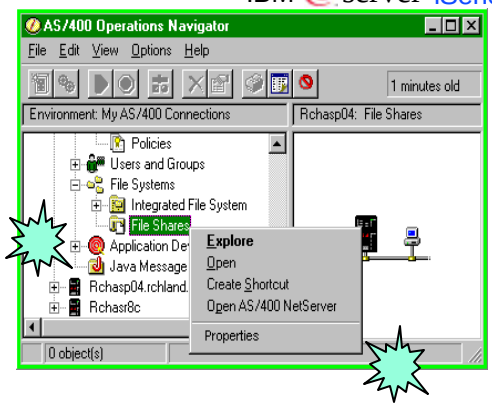
- Functions are arranged in a hierarchy
- User works with lists of objects that represent iSeries resources
- Lists are "snapshots" of server activity
- Actions are defined on each object type
- Actions may be unique or common (e.g. Copy, Paste, Delete, Properties)
- Actions may be enabled or disabled based on the object's state
- Online help is available for each action
- When the user selects an action, Operations Navigator displays a dialog.
  - confirmation
  - single panel that contains additional data
  - property sheet or wizard
- When the user clicks OK, the requested action is performed.
  - If action was successful, the main OpNav window and status area are refreshed.
  - If an error occurred, an error message is displayed.
  - Return is made to the dialog to allow user to correct the error.

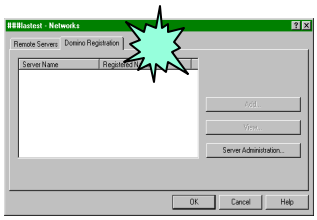
IBM @server. For the next generation of e-business.

Attaching to the Tree

IBM @server iSeries

- Add your folder to the tree
  - At any container
- Add your action to another object's menu
- Add your property page to another object's property sheet.



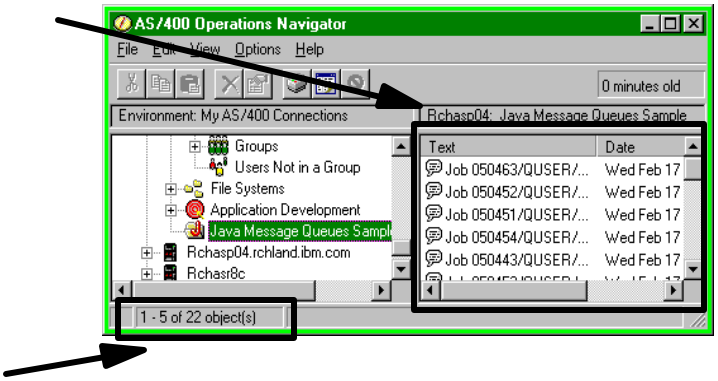


IBM @server. For the next generation of e-business.

Lists and Status

IBM @server iSeries

- Enumerate objects
- Manage objects
- Manage columns
- Refresh levels
  - Object
  - List
  - Tree (everything)
- Status messages

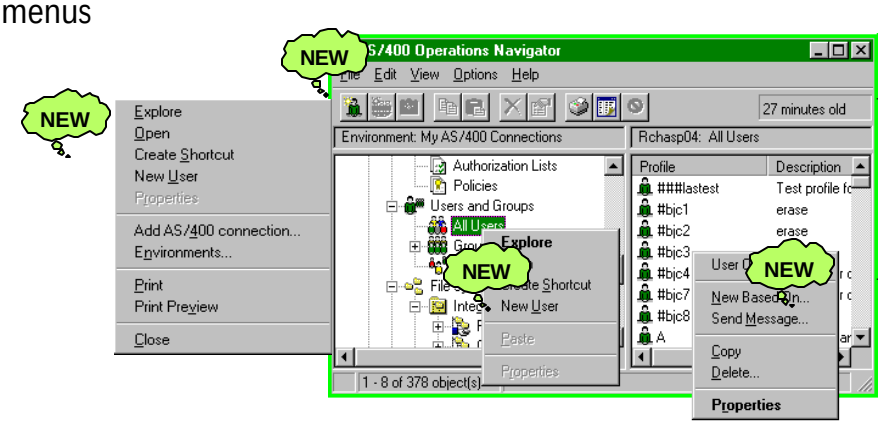


IBM @server. For the next generation of e-business.

Add New Menu Items and Buttons

IBM @server iSeries

- Menu Items drive
  - Pull down menus
  - Context menus
  - Buttons



Menus and Buttons *(Notes)*

IBM @server iSeries

- Depends on currently selected item
- May depend on other things
  Security etc..
- Each menu item has associated action or command to invoke
  Commands may be reused
- Pull-down and context menus have same content
  same menu handler called with different flags
- Toolbar buttons also utilize menu handler

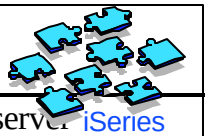
What does Op Nav Plug-in Support offer?

IBM @server iSeries

Op Nav Plug-in Interfaces and Services

IBM @server. For the next generation of e-business.

How does it work?



IBM @server iSeries

- The Operations Navigator user interface framework is designed to let users work with lists of AS/400 resources and to perform actions on them.
- The architecture of the Plug-In feature reflects this user interface design by defining interfaces for working with
  - lists of objects in a hierarchy,
  - for defining actions on those objects,
  - and to handle drag-and-drop operations.
- You implement a set of predefined classes and methods, which are invoked by Operations Navigator in response to a particular user action.

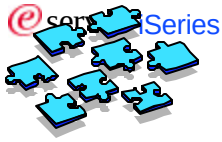
IBM @server. For the next generation of e-business.

OpNav Building Blocks

Op Nav communicates with the Java plug-in using the following interfaces that you implement:

- Interfaces
  - ActionsManager - for menus, toolbars and buttons
  - ListManager - for listing objects
  - DropTargetManager - for handling drag and drop
  - PropertiesManager
  - TasksManager
- Resources
  - MRI dll - displays your icon in the Op Nav tree
- Classes
  - jar files

OpNav Building Blocks

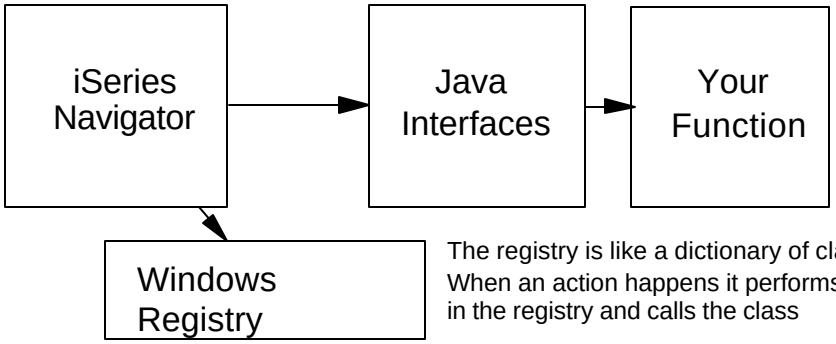


- Common services provided 'for free' by Op Nav
  - Refresh item / list - Auto refresh automatically refreshes the list of objects based on a time specified by the user
  - Set status text -Displays how many items are listed and how old the data is
  - Flyover help - Provides the capability for help when the cursor points to a field. You provide the help text.

How does Op Nav know about your Plug-in?

IBM @server iSeries

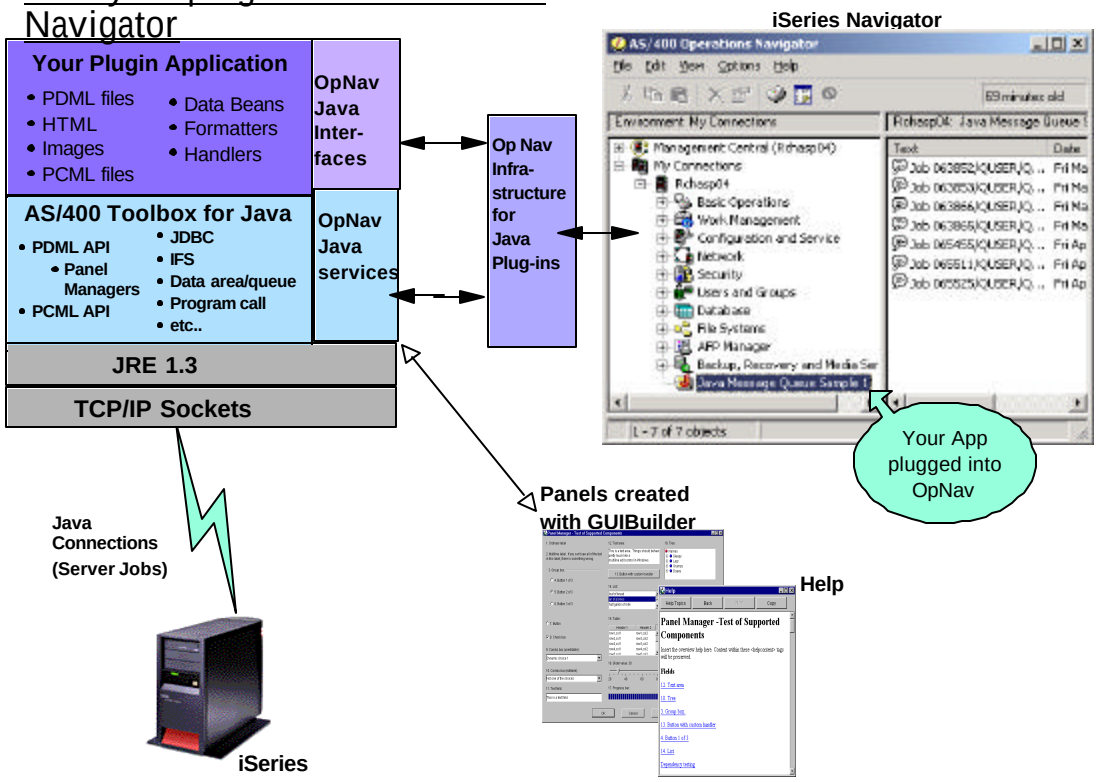
- Registry Entries - provide cross-references to make your plugin known to Op Nav
- You use standard interfaces and provide Windows registry entries
- Whenever an action occurs it looks in the Registry



The registry is like a dictionary of classes
When an action happens it performs a lookup in the registry and calls the class

IBM @server. For the next generation of e-business.

How your plugin fits with iSeries Navigator



OpNav Plug-in Interfaces

IBM @server iSeries

- Interface types
  - Context menu handlers
  - Drag drop hander
  - Property sheet handlers
  - Data servers / List managers
  - Taskpad managers

IBM @server. For the next generation of e-business.

Standard Interfaces (Notes)

IBM @server iSeries

Java interfaces parallel those used in C++

Function	Java Interface	C++ COM Interface
Context Menu Display and Command Execution	ActionsManager queryActions actionSelected	IContextMenu QueryContextMenu InvokeCommand
Enumerating Object Lists	ListManager open getItemCount itemAt getAttributes getColumnData ...Etc.	IA4HierarchyFolder Open GetItemCount ItemAt GetAttributesOf GetColumnDataItem ...
Handling Drag and Drop	DropTargetManager dragEnter dragOver dragExit	IDropTarget DragEnter DragOver DragLeave
Taskpad	TasksManager	IA4TasksManager

IBM @server. For the next generation of e-business.

ActionsManager Interface

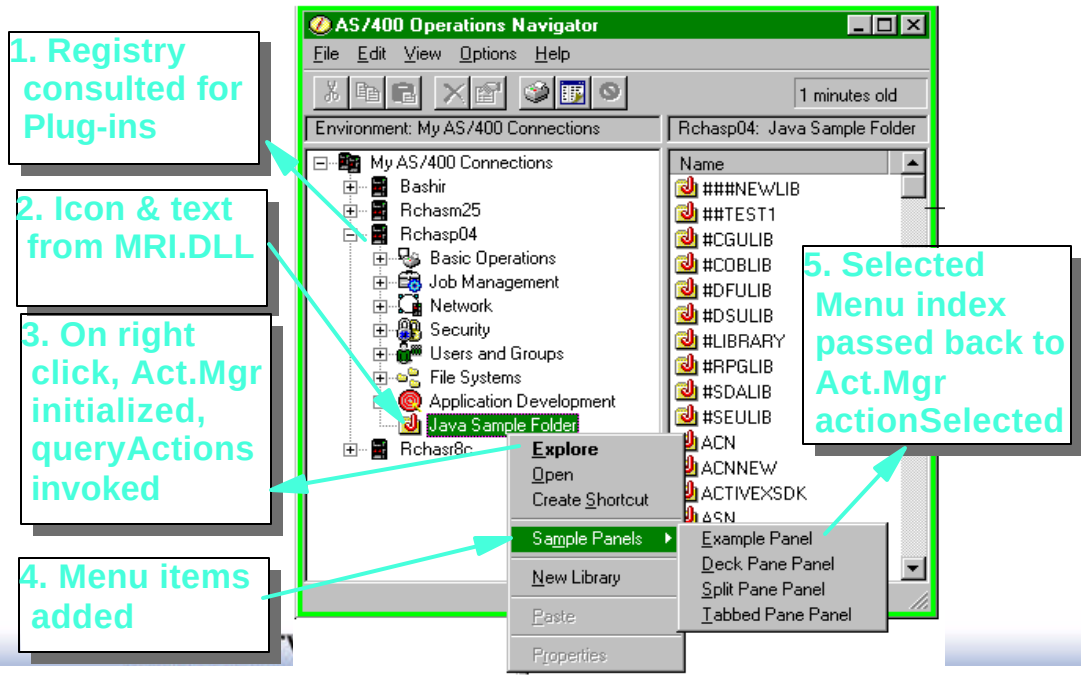
IBM @server iSeries

- ActionsManager provides lists of user actions to a user interface and also performs actions requested by the user
- To display menu choices and act on menu requests, use the ActionsManager interface
  - **initialize** - passes in the object selected
  - **queryActions** - lets you add menu items for the object
  - **actionSelected** - tells you which menu item was selected

IBM @server. For the next generation of e-business.

ActionsManager Diagram

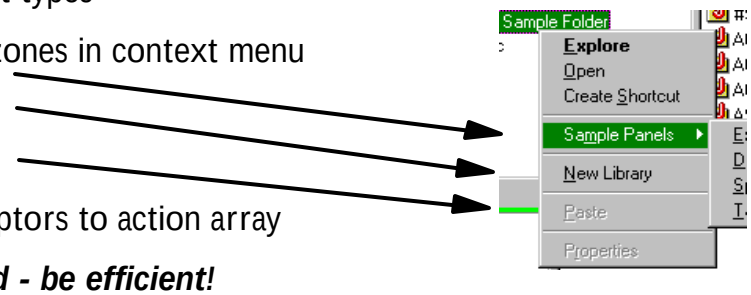
IBM @server iSeries



queryActions for ActionsManager

IBM @server iSeries

- **queryActions** (building context menu)
  - get the object type
  - handle context for drag-drop first
  - check for different types
  - check for action zones in context menu
    - CUSTOM
    - CREATION
    - STANDARD
  - add ActionDescriptors to action array
  - ***frequently called - be efficient!***



IBM @server. For the next generation of e-business.

Toolbars

IBM @server iSeries

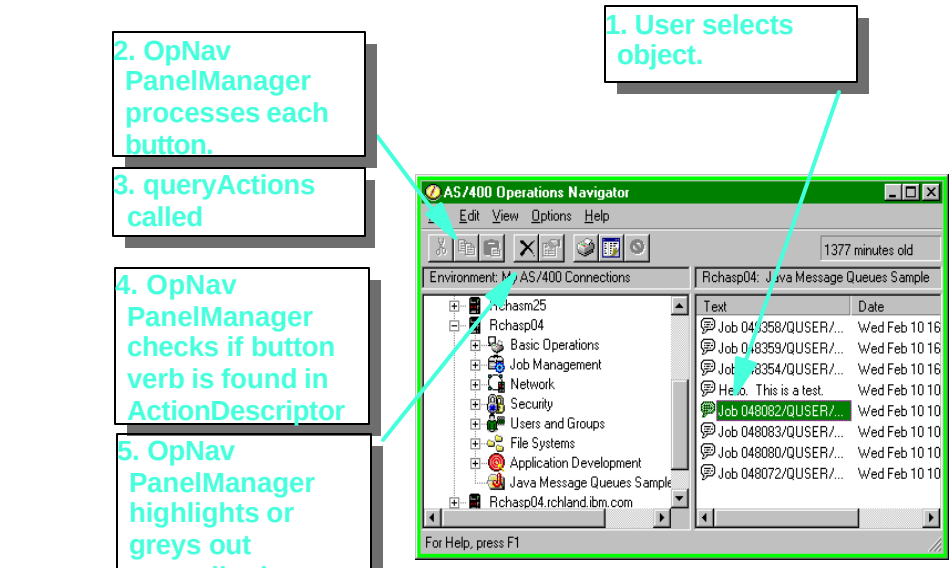
- You provide ...
  - toolbar bitmap (.bmp files)
  - toolbar information (text and hover help)
- queryActions is called for each button
- clicking the toolbar button generates the action
- actionSelected called with identifier of action
  - can be same action as menu action

IBM @server. For the next generation of e-business.

Toolbar Diagram

IBM eserver iSeries

How does the toolbar work?

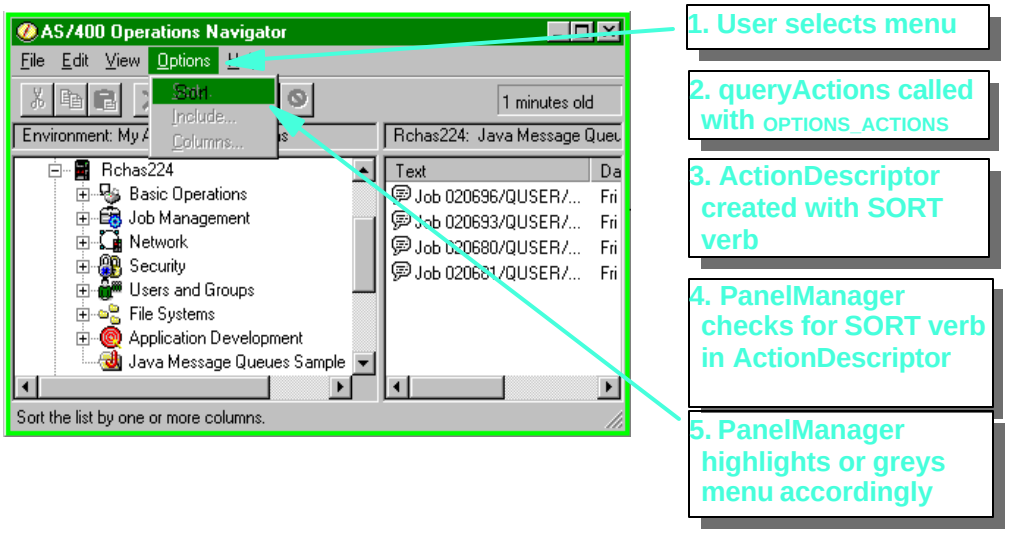


IBM eserver. For the next generation of e-business.

Options Diagram

IBM eserver iSeries

How does the Option menu work?



IBM eserver. For the next generation of e-business.

ListManager Interface

IBM @server iSeries

- A ListManager implementation constructs and manages a list of objects to a user interface,
- Required to populate OpNav component lists
  - **initialize** - identifies the container object to be enumerated
  - **open** - obtain and construct the objects (called on separate "data" thread - ***do long stuff here!***)
  - **getStatus** - returns the current status of the list
  - **getErrorMessage** - if getStatus is LIST_ERROR
  - **getItemCount** - should return the total number of object in the list
  - **itemAt** - returns ItemIdentifier of nth object

IBM @server. For the next generation of e-business.

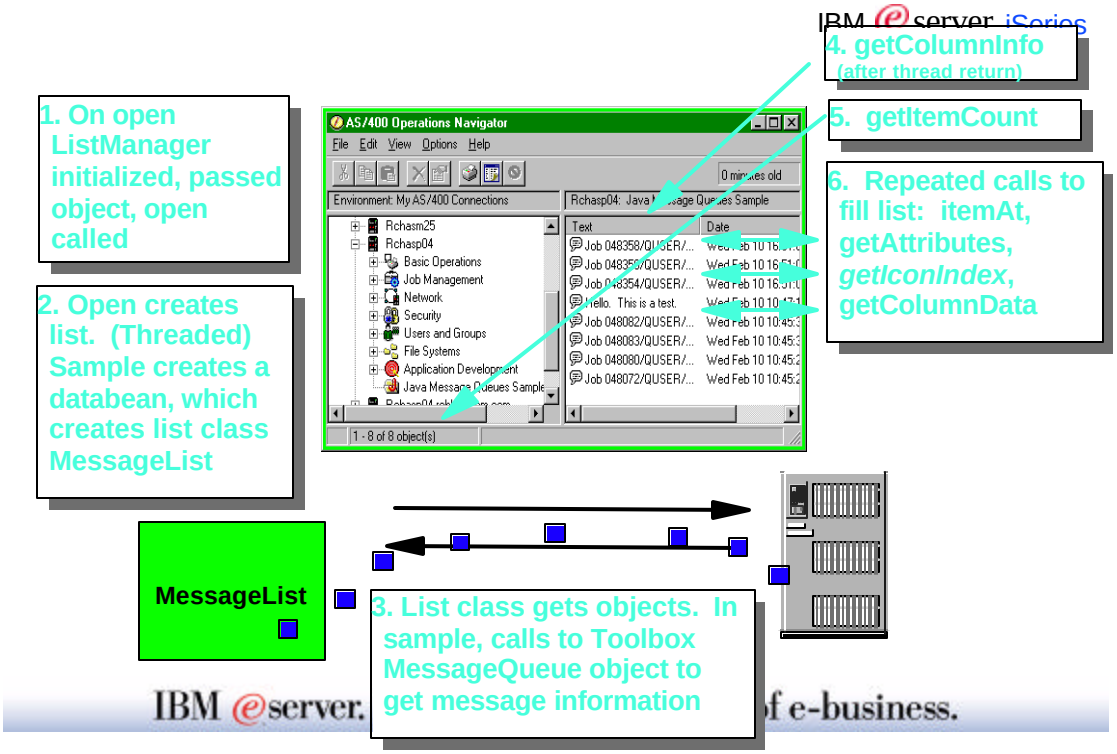
ListManager Interface

IBM @server iSeries

- **getAttributes** - attribute flags for nth object
- **getColumnInfo** - returns column headings
- **getColumnData** - returns data for column list
- **getListObject** - returns cached list object (e.g. for another interface)
- **close** - frees all resources
- **prepareToExit** - last chance to clean up.
- ***getIconIndex*** *- separate, non-interface method currently provides icon for nth object*

IBM @server. For the next generation of e-business.

ListManager Diagram



DropTargetManager Interface

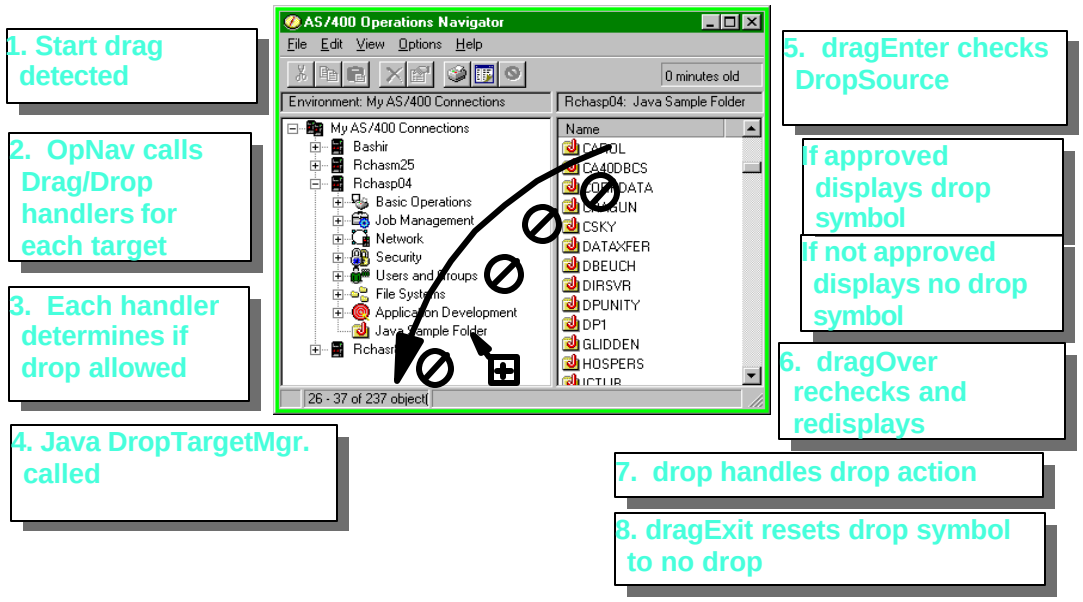
IBM @server iSeries

- Handles drag and drop processing
  - **initialize** - identifies target of drag/drop operation
  - **dragEnter** - indicates whether a drop can be accepted and the action that will take place on the drop
  - **dragOver** - called while the user continues over drop target (usually the same as dragEnter)
  - **dragExit** - user left the target or cancelled
  - **drop** - user indicates drop. Action is taken.

IBM @server. For the next generation of e-business.

DropTargetManager Diagram

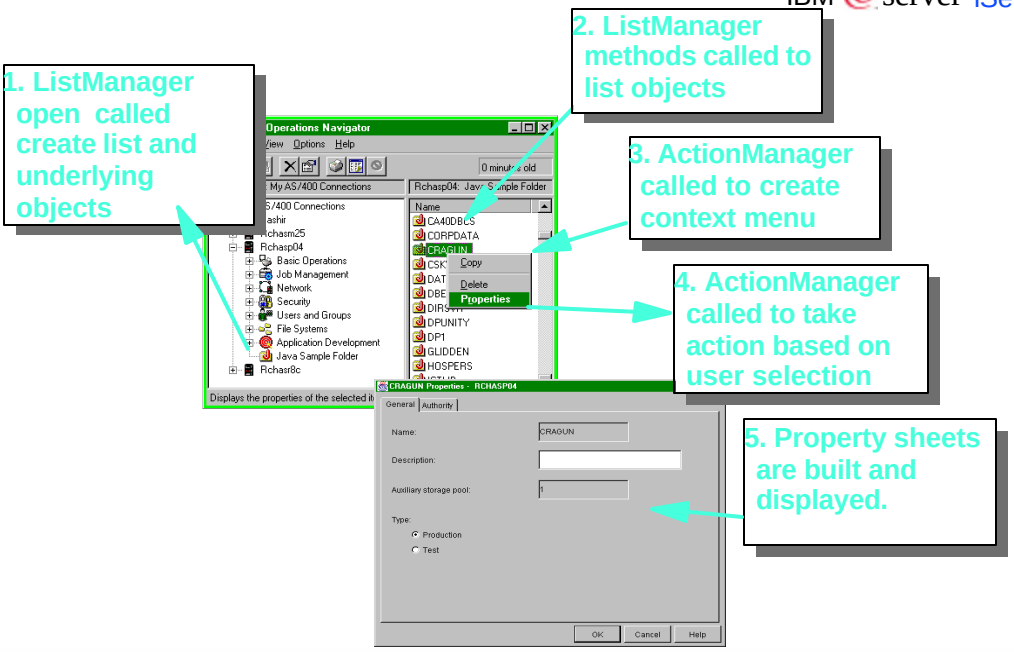
IBM @server iSeries



IBM @server. For the next generation of e-business.

PropertyPage Diagram

IBM @server iSeries



IBM @server. For the next generation of e-business.

Programmatic Identifier (ProgID)

IBM @server iSeries

- Used in registry file and in the directory structure of the client
- Indentifies your Plug-in uniquely in the registry
- Form: **vendor . component**
- Example: Lotus.Domino, IBM.Sample

IBM @server. For the next generation of e-business.

Sample registry entry (Notes)

IBM @server iSeries

```
; Define the primary registry key for the plugin

[HKEY_CLASSES_ROOT\IBM.AS400.Network\3RD PARTY EXTENSIONS\IBM.MsgQueueSample4]
"Type"="Plugin"
"MinimumRISCRelease"="ANY"
"MinimumIMPIRelease"="NONE"
"ProductID"="NONE"
"ServerEntryPoint"="NONE"
"NLS"="c:\\MsgQueueSamples\\win32\\mri\\MessageQueuesMRI.dll"
"NameID"=dword:00000001
"DescriptionID"=dword:00000002
"JavaPath"="c:\\MsgQueueSamples;"
"JavaMRI"=""
;-------------------------------------------------------------------
; Add key that indicates that the plug-in supports SSL
[HKEY_CLASSES_ROOT\IBM.AS400.Network\3RD PARTY EXTENSIONS\IBM.MsgQueueSample4\SSL]
"Support Level"=dword:00000001
;-------------------------------------------------------------------
; Register a new folder

[HKEY_CLASSES_ROOT\IBM.AS400.Network\3RD PARTY EXTENSIONS\IBM.MsgQueueSample4\folders\Sample4]
"Attributes"=hex:00,01,00,a0
"CLSID"="{1827A856-9C20-11d1-96C3-00062912C9B2}"
"JavaClass"="com.ibm.as400.opnav.MsgQueueSample4.MqListManager"
"Parent"="AS4"
```

IBM @server. For the next generation of e-business.

(Sample Registry Entry continued) Notes

IBM @server iSeries

"NameID"=dword:0000000f
"DescriptionID"=dword:00000010
"DefaultIconIndex"=dword:00000001
"OpenIconIndex"=dword:00000000
;---
; **Register a context menu handler for the new folder and its objects**

[HKEY_CLASSES_ROOT\IBM.AS400.Network\3RD PARTY EXTENSIONS\IBM.MsgQueueSample4\shellex\Sample4***ContextMenuHandlers**\{1827A857-9C20-11d1-96C3-00062912C9B2}]
"JavaClass"="com.ibm.as400.opnav.MsgQueueSample4.MqActionsManager"
;---
; **Register a drag/drop context menu handler for the new folder and its objects**

[HKEY_CLASSES_ROOT\IBM.AS400.Network\3RD PARTY EXTENSIONS\IBM.MsgQueueSample4\shellex\Sample4***DragDropHandlers**\{1827A857-9C20-11d1-96C3-00062912C9B2}]
"JavaClass"="com.ibm.as400.opnav.MsgQueueSample4.MqActionsManager"
;---
; **Register a drop handler for the new folder and its objects**

[HKEY_CLASSES_ROOT\IBM.AS400.Network\3RD PARTY EXTENSIONS\IBM.MsgQueueSample4\shellex\Sample4* **\DropHandler**]
@="{1827A857-9C20-11d1-96C3-00062912C9B2}"
"JavaClass"="com.ibm.as400.opnav.MsgQueueSample4.MqDropTargetManager"

IBM @server. For the next generation of e-business.

How can I build a plug-in?

IBM @server iSeries

Overview

IBM @server. For the next generation of e-business.

Plugin Development Cookbook

Steps to developing your plugin

IBM @server iSeries

1. Set up the tools, jdks, classpath, etc.
2. Develop your Graphical User Interface (GUI) with the GUIBuilder
3. Develop your Application
4. Getting and Putting data to the iSeries (using PCML and the Toolbox for Java)
5. Plug your app into the iSeries Navigator via the registry
6. Debugging
7. Installing your plug-in

IBM @server. For the next generation of e-business.

How can I build a plug-in?

IBM @server iSeries

Step 1: Set up the tools

IBM @server. For the next generation of e-business.

Setup Your Development Environment

IBM @server iSeries

- Install iSeries Access for Windows
  - Select Toolbox for Java from Selective install menu
- Run CheckVersion to apply GA service pack
- Install the Java Developer's Kit (JDK 1.3 for V5R1) IBM's developerWorks website: http://www-106.ibm.com/developerworks/
- Set up classpath:

```
jdk1.3\lib\;
ClassPath=
 .;C:\java;C:\jdk1.3;
 C:\Program Files\IBM\ClientAccess\jt400\lib\uitools.jar;
  C:\Program Files\IBM\Client Access\classes\jopnav.jar;
 C:\Program Files\IBM\Client Access\jt400\lib\jt400.jar;
 C:\Program Files\IBM\Client Access\jt400\lib\jui400.jar;
 C:\Program Files\IBM\Client Access\jt400\lib\data400.jar;
 C:\Program Files\IBM\Client Access\jt400\lib\util400.jar;
 C:\Program Files\IBM\Client Access\jt400\lib\x4j400.jar;
  C:\Program Files\IBM\Client Access\JRE\Lib\jhall.jar;
Path=...;C:\jdk1.3\bin;
```

IBM @server. For the next generation of e-business.

Required JAR Files

IBM @server iSeries

- **uitools.jar** Contains the GUI Builder and Resource Script Converter tools.
- **jui400.jar** Contains the runtime API for the Graphical Toolbox. Java programs use this API to display the panels constructed using the tools. May be redistributed.
- **data400.jar** Contains the runtime API for the Program Call Markup Language (PCML). Java programs use this API to call iSeries programs whose parameters and return values are identified using PCML. May be redistributed.
- **util400.jar** Contains utility classes for formatting iSeries data and handling AS/400 messages. May be redistributed.
- **x4j400.jar** Contains the XML parser used by the API classes to interpret PDML and PCML documents.
- **jt400.jar** Contains the Toolbox for Java jar .zip file
- **jhall.jar** Contains Java Help
- **jopnav.jar** Contains OpNav infterfaces that each component implements

IBM @server. For the next generation of e-business.

How can I build a plug-in?

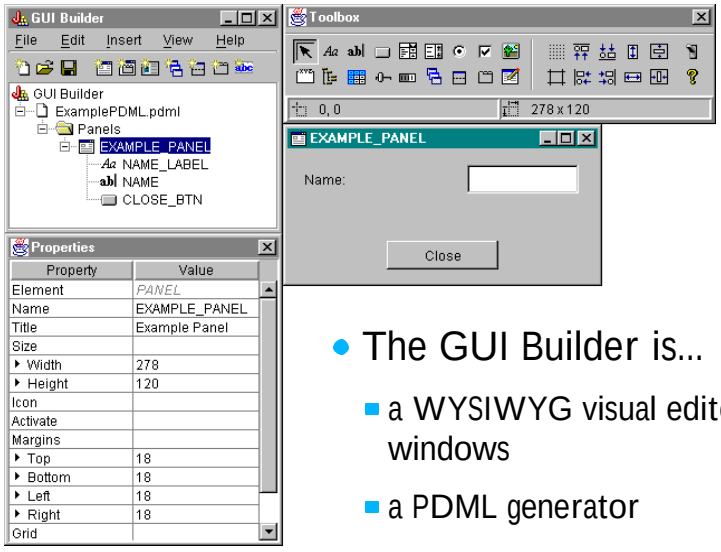
IBM eserver iSeries

Step 2: Develop the Graphical User Interface

IBM eserver. For the next generation of e-business.

Simpler Java Dialogs with GUIBuilder

IBM eserver iSeries



- The GUI Builder is...
  - a WYSIWYG visual editor to build Java windows
  - a PDML generator
  - the starting point for all Java development!!

IBM eserver. For the next generation of e-business.

Development and Tools

IBM @server iSeries

- Easily create graphical user interface panels in Java with the Graphical Toolbox
  - GUIBUILDER and Resource Script Converter
- Incorporate the panels into your Java applications, applets, or iSeries Navigator plug-ins
- Program Call Markup Language (PCML) for iSeries data access
- Graphical Toolbox and PCML are packaged with the Toolbox for Java
  - Installed via Client Access selective install or a web download

IBM @server. For the next generation of e-business.

What is PDML?

IBM @server iSeries

- PDML is a language that IBM developed using XML. It's similar in structure to HTML and SGML.
- Here are some examples of PDML tags:
  - <panel> -- defines a panel
  - <title> -- specifies the title of the panel or field
  - <size> -- specifies the size of the panel or field
  - <label> -- defines a label on the panel (static text field)
  - <location> -- specifies the location of the field on the panel
  - <button> -- defines a button on the panel
  - <textfield> -- defines a textfield on the panel
  - and many more...

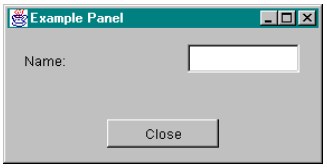
IBM @server. For the next generation of e-business.

PDML Example

IBM @server iSeries

Here's what the PDML looks like for the simple panel shown below:

```
<PDML version="1.0" source="JAVA" basescreensize="1024x768">

    <PANEL name="EXAMPLE_PANEL">
        <TITLE>EXAMPLE_PANEL</TITLE>
        <SIZE>278,120</SIZE>
        <LABEL name="NAME_LABEL" disabled="no">
            <TITLE>NAME_LABEL</TITLE>
            <LOCATION>15,20</LOCATION>
            <SIZE>100,19</SIZE>
        </LABEL>
        <TEXTFIELD name="NAME" masked="no" editable="yes"
                   disabled="no">
            <TITLE>NAME</TITLE>
            <LOCATION>161,14</LOCATION>
            <SIZE>100,26</SIZE>
        </TEXTFIELD>
        <BUTTON name="CLOSE_BTN" disabled="no">
            <TITLE>CLOSE_BTN</TITLE>
            <LOCATION>89,83</LOCATION>
            <SIZE>100,26</SIZE>
        </BUTTON>
    </PANEL>

</PDML>
```



IBM @server. For the next generation of e-business.

What can I do with the GUI Builder?

IBM @server iSeries

- Using the GUI Builder, you can:
  - Create GUIs
  - Manage the translatable text
  - Generate Help skeleton
  - Generate Java Beans for data management

IBM @server. For the next generation of e-business.

What are the advantages for using the GUI Builder? (Notes)

IBM @server iSeries

- 2 Other methods to create Java panels:
  - Write raw AWT code (without a graphical tool)
    - (+) Panel components automatically resize for each NLV
    - (-) Panel layout is defined in the Java code
    - (-) Requires many edit, compile, and debug cycles before getting it right
  - Use graphical tools that generate AWT code
    - (-) Panel component locations and sizes are absolute
    - (-) Panel layout is defined in the Java code
    - (-) Translators must know Java to resize panels for each language

IBM @server. For the next generation of e-business.

What are the advantages for using the GUI Builder? (Notes)

IBM @server iSeries

- How does that compare with the GUI Builder?
  - Use GUI Builder
    - (-) Panel component locations and sizes are absolute
      - but that's OK cecause they are not defined in code
    - (+) Panel layout is defined in PDML (no compilation required)
    - (+) Fits well into the existing translation process:
      - Structure is similar to the Windows .rc files
      - Translators don't change Java code (fewer bugs)
      - Translators don't need to know Java to resize panels
    - (+) Provides additional functionality, such as generating the help and data beans
    - (+) If user interfact technologies chnage, the UI definition doesn't need to change

IBM @server. For the next generation of e-business.

Relationship between the GUI Builder & Your IDE?

IBM @server iSeries

- GUI Builder
  - Builds the end-user interface -- the panels
- Java Development Environment (e.g. Visual Age, Symantec Cafe, etc.)
  - Builds, compiles, and manages the Java code
  - Do not use it to build the end-user interface -- the panels (use the GUI Builder instead)

IBM @server. For the next generation of e-business.

Creating Java Windows

IBM @server iSeries

The GUI Builder allows you to create different types of Java windows:

- **Panels** -- simple dialogs
- **Property Sheets** -- Notebook pages with tabs at the top
- **Wizards** -- multiple pages displayed sequentially
- **Tabbed Panes** -- multiple pages with tabs at the top, bottom, right or left
- **Split Panes** -- multiple pages displayed adjacent horizontally or vertically
- **Deck Panes** -- multiple pages layed on top of each other
- **Menus** -- menu bars containing menu items
- **Toolbars** -- toolbars with menu items that can link to existing menu items in the menu bar
- **Context Menus** -- context menu definitions that can be used to pop-up context menus for components

IBM @server. For the next generation of e-business.

Managing translatable text

IBM @server iSeries

- The GUI Builder provides the String Table function in order to manage translatable text.
- Use the String table for:
  - Messages
  - Menu text and status bar help
  - Additional status bar information (e.g. User Bob created)
- What about the text on the panels?
  - The field text on each panel is translatable.
  - The field text does not need to be added to the string table.
  - The field text is stored in the same file as the string table information. During translation, it will also get translated.

IBM @server. For the next generation of e-business.

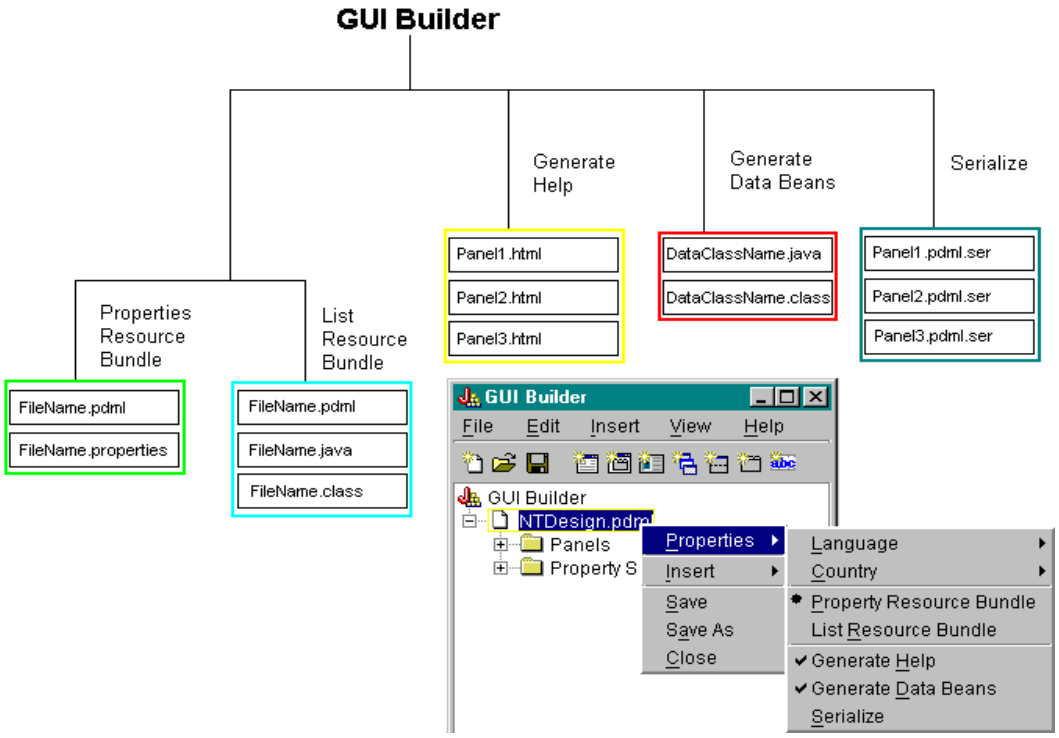
How can I build a plug-in?

IBM @server iSeries

Step 3: Develop your Application

IBM @server. For the next generation of e-business.

Files generated by GUI Builder



Files Generated by GUIBuilder *(Notes)*

IBM @server iSeries

► Properties Resource Bundle

By default, it uses the Properties Resource Bundle which will create two files for you:

- **FileName.pdml -- holds the panel definition information, such as the type of fields, their location, etc..**
- **FileName.properties -- holds all of the translatable text**

► List Resource Bundle

Used to enhance performance of the application.

Compiles the panel definitions and translatable text, so performance is better at run-time.

- **However, it will slow down performance in the GUI Builder when previewing panels.**

This bundle will create three files for you:

- **FileName.pdml -- holds the panel definition information, such as the type of fields, their location, etc..**
- **FileName.java -- holds all of the translatable text**
- **FileName.class** -- compiled version of FileName.java

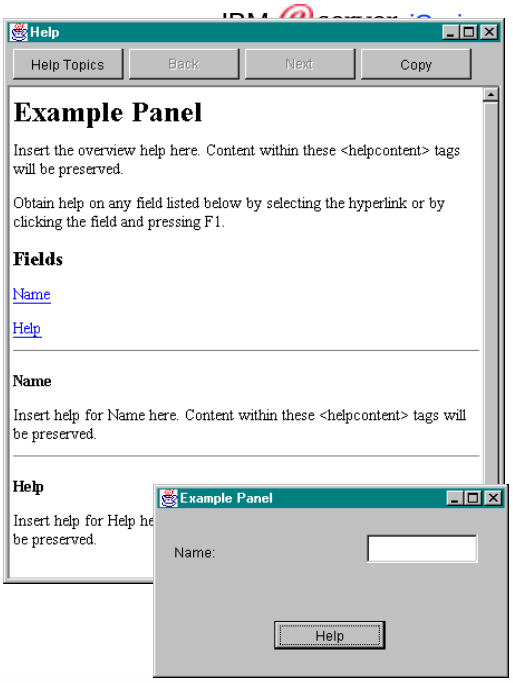
► Serialization

- Compresses the information for each window in the PDML file.
- Enhances performance of your Java panels.
- Creates these files for you:
  - **PanelName.pdml.ser** (for each panel in the PDML file)

IBM @server. For the next generation of e-business.

Help Generation

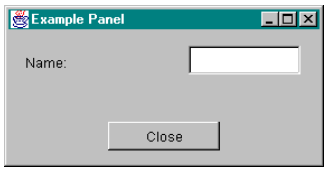
- Creates help skeletons for each panel in the PDML file. This help skeleton will appear when the user click on the F1 or Help button.
- Does not overwrite any existing help in help skelton
- The help file cannot be tested in the GUI Builder. You must load the file in a Java application for the help to appear.
- Creates these files for you:
  - **PanelName.html** (for each panel in the PDML file)



IBM @server. For the next generation of e-business.

Files generated by GUI Builder

IBM @server iSeries



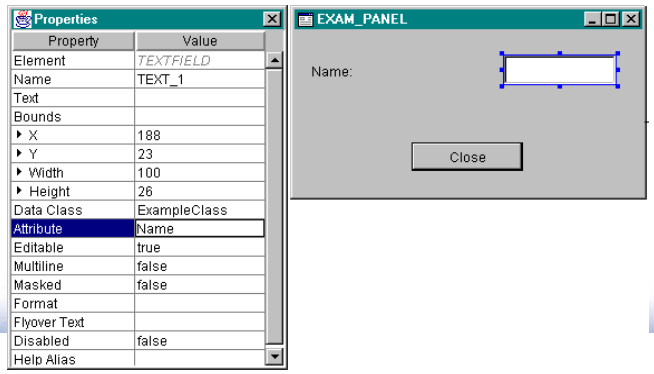
- Generating Data Beans
  - Generates data bean skeletons (get and set methods) for fields in the PDML.
  - Makes it easy for you to get started on the Java code for the panels.
  - Creates these files for you:
    - **DataClassName.java** (for each data class specified in the PDML file) -- file with get and set methods for each attribute in the data class
    - **DataClassName.class** (for each data class specified in the PDML file) -- compiled version of the corresponding .java file

IBM @server. For the next generation of e-business.

Data Bean Generation

IBM eserver iSeries

- Data Class and Attributes
  - used to send and receive information to and from individual fields on the panels.
  - Specify the Data Class and Attribute in the Properties
  - Only generates Databeans if
    - Generate databeans is set to "True" at save
    - Dataclass & attribute fields are specified



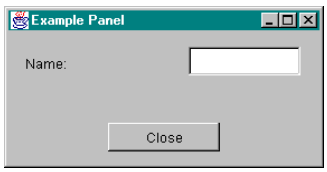
IBM eserver.

Data Bean Generation

IBM eserver iSeries

Example of a data bean skeleton (get and set methods)

Data Class == ExampleClass
Attribute == Name



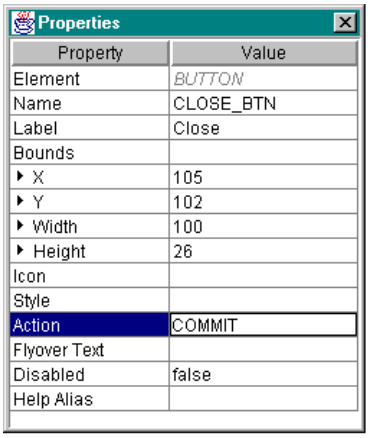

```
import com.ibm.as400.ui.framework.java.*;

public class ExampleClass extends Object implements DataBean
{
   private String m_sName;

  public String getName()
  {      return m_sName;    }

  public void setName(String s)
  {      m_sName = s;   }

  public Capabilities getCapabilities()
  {      return null;   }

  public void verifyChanges()
  {   }

  public void save()
  {   }

  public void load()
  {      m_sName = "";   }
}
```

IBM eserver. For the next generation of e-business.

Button Handling

IBM @server iSeries

- Button Actions
  - Specify each button's Action handler in the Properties (by specifying the class name)
  - Generates a button handler skeleton
  - Three Button Actions are available without having to write a button handler (must specify in Properties):
    - COMMIT -- For OK buttons
    - CANCEL -- For Cancel buttons
    - HELP -- For Help buttons



IBM @server. For the next generation of e-business.

How can I build a plug-in?

IBM @server iSeries

Step 4: Getting Data to / from the iSeries

IBM @server. For the next generation of e-business.

Getting data to / from the iSeries

IBM @server iSeries

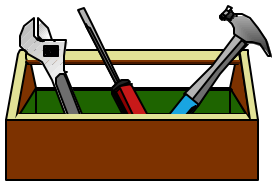
- Toolbox for Java objects
  - CL commands - The Toolbox for Java provides a facility to run CL commands
  - JDBC - the Toolbox for Java provides the drive
  - iSeries API's
  - PCML - to make it easier to call iSeries API's

IBM @server. For the next generation of e-business.

Toolbox for Java Objects

IBM @server iSeries

Cool objects already built for you...



- Users and groups
- User spaces
- Data queues
- Messages and message queues
- Digital certificates
- System values
- System status
- Object authority
- Jobs and job attributes
- Data areas, etc...

IBM @server. For the next generation of e-business.

IBM @server iSeries

PCML - Simpifying iSeries Data Access

IBM @server. For the next generation of e-business.

What is PCML?

IBM @server iSeries

- Program Call Markup Language
  - Simplifies the process of calling iSeries programs from Java programs
  - An XML-based language to define iSeries program interfaces
  - PCML describes an iSeries program interface
    - Used to drive automatic data translation
- com.ibm.as400.data.ProgramCallDocument
  - A Java class representing the interfaces described in a PCML source file
  - Performs data conversion and program calls to the iSeries

IBM @server. For the next generation of e-business.

More on PCML

IBM @server iSeries

- PCML is implemented as a package of java classes
  - com.ibm.as400.data (data400.jar)
- Built on top of the Toolbox for Java access package
  - com.ibm.as400.access (jt400.zip/jt400.jar)
- Shipped with Toolbox for Java

IBM @server. For the next generation of e-business.

Simplifying iSeries program calls

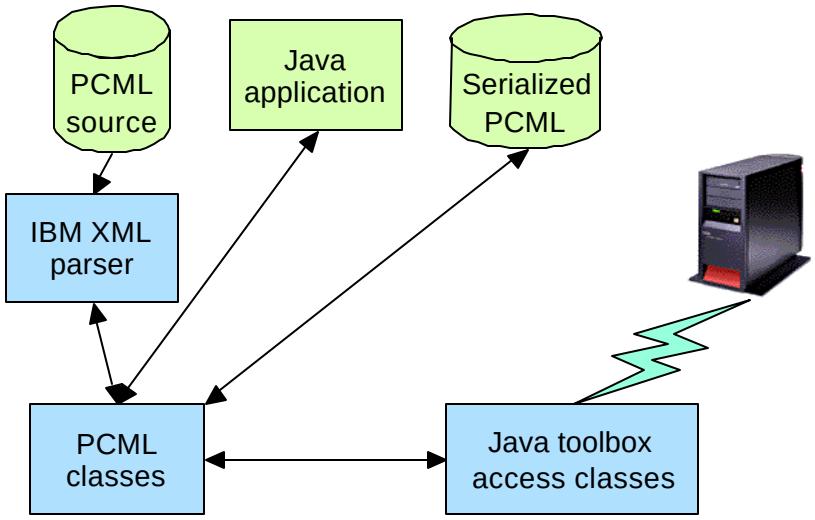
IBM @server iSeries

- Simplifies Java programs by handling complex relationships in iSeries data
  - Varying length character strings
  - Varying length structures
  - Varying sized arrays of fields and structures
  - Nested arrays
  - Varying locations of data -- offsets and displacements
  - Strings with run-time CCSID tagging
  - Parameters or fields within a structure that exist only at specific OS release levels

IBM @server. For the next generation of e-business.

Structure using PCML

IBM @server iSeries



IBM @server. For the next generation of e-business.

PCML Example - PCML Source Code

IBM @server iSeries

```
<pcml version="1.0">
  <program name="foobar"
           path="/QSYS.lib/JOE.lib/RABOOF.
    <data name="fred"  type="char" length=
    <data name="ethel" type="int"  length="4" />
    <struct name="lucyAndRicky">
      <data name="lucy" type="char" length="1" />
      <dat(1)name="ricky" type="int" length="4" />
    </stru(2)>
  </progra(3)
</pcml>
```

name= does not have to match the name of the actual program

<program> contains one tag for each parameter

IBM @server. For the next generation of e-business.

PCML Example - Java code

IBM @server iSeries

```
AS400 as400 = new AS400();
try {
  ProgramCallDocument pcml =
      new ProgramCallDocument(as400, "example");

  pcml.callProgram("foobar");

  Object val = pcml.getValue("foobar.fred");
  val = pcml.getValue("foobar.lucyAndRicky.lucy");
}
catch (PcmlException pe) {}
```

Construct using an AS400 object and reference to PCML resource

Set input values and get output values using "qualified name"

IBM @server. For the next generation of e-business.

The ProgramCallDocument class

IBM @server iSeries

- PCML is implemented by the com.ibm.as400.data package
- The public class **`ProgramCallDocument`** does all the work
- Constructed with an iSeries (toolbox) object, and the name of the PCML resource

```
pcml = new ProgramCallDocument(myAs400,
           "qsyrusri");
```

PCML resource name

IBM @server. For the next generation of e-business.

The ProgramCallDocument class

IBM @server iSeries

- In the above example, "qsyrusri" is the name of a PCML resource to be located via the classpath.
  - Constructor first looks for serialized version of resource
    - **"qsyrusri.pcml.ser"**
  - If serialized version not found, source version must be found
    - **"qsyrusri.pcml"**
  - Resource name can be package qualified on constructor
    - **"com.ibm.mypkg.qsyrusri"**

IBM @server. For the next generation of e-business.

<data> Elements *(Notes)*

IBM @server iSeries

- **Data elements are accessed in Java by specifying the fully qualified name of the element.**
- **A qualified name consists of the qualified name of the element's parent, a period separator, followed by the element's `name=` attribute**

For example:

```
pcml.getValue("qsyrusri.receiver.userName");
```

Comes from:

```
<program name="qsyrusri">
  <struct name="receiver">
    <data name="userName">
```

IBM @server. For the next generation of e-business.

Data Types *(Notes)*

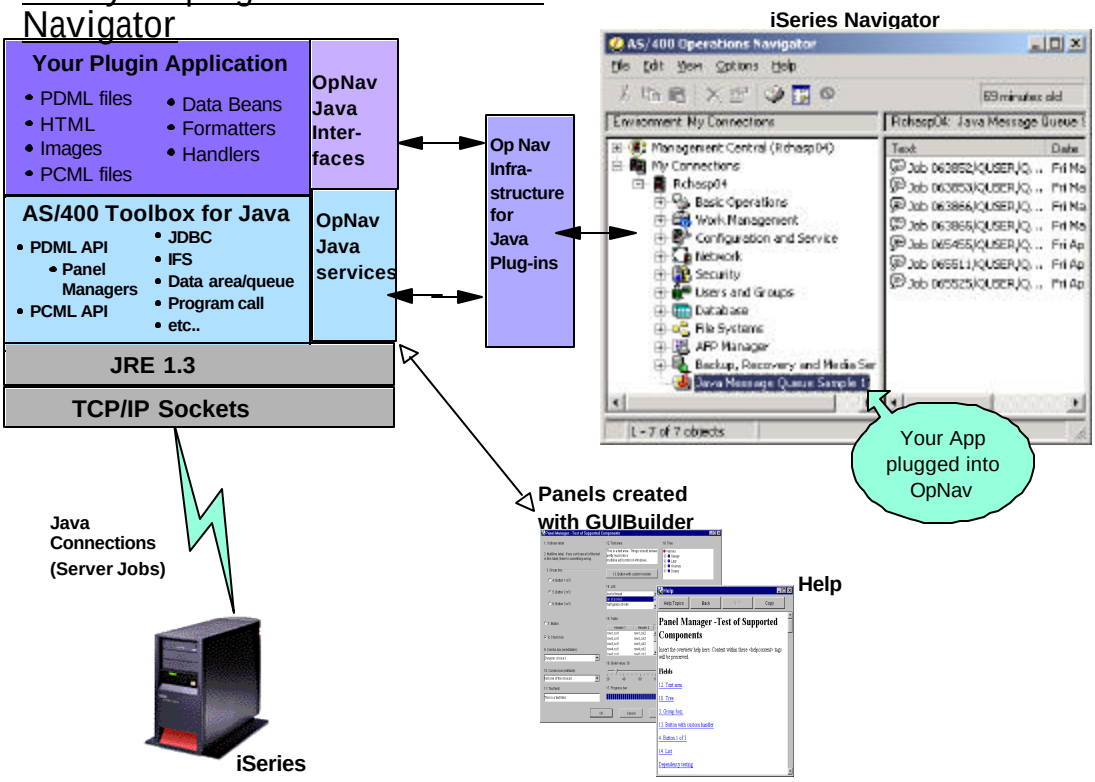
IBM @server iSeries

Java object is input to
`ProgramCallDocument.setValue()`
and returned from
`ProgramCallDocument.getValue()`

AS/400 type=	length=	precision=	Java object
char	0..65536 *data-name*		java.lang.String
int	2 4	15 (signed) 16 (unsigned) 31 (signed) 32 (unsigned)	java.lang.Short java.lang.Integer java.lang.Integer java.lang.Long
packed	1..31	0..*length*	java.math.BigDecimal
zoned	1..31	0..*length*	java.math.BigDecimal
float	2 4		java.lang.Float java.lang.Double
byte	0..65536 *data-name*		byte [ ]

IBM @server. For the next generation of e-business.

How your plugin fits with iSeries Navigator



How can I build a plug-in?

IBM @server iSeries

Step 5: Plug your applications into Op Nav using the registry

IBM @server. For the next generation of e-business.

Plugging your App Into the iSeries Navigator Registry

IBM @server iSeries

Files needed to make your plugin known to Op Nav

★ **Suggest copying setup.ini and filename. reg file from the sample code**

- **Setup.ini** -
  - File that provides the information needed to install, upgrade, and service an iSeries Navigator plug-in.
- **mri.dll -**
  - Provides the icon and text that is displayed for your plug-in (i.e. the icon and description in the iSeries Navigator tree)
  - Can be created by any tool that creates MFC mri.dlls
- **filename.reg -**
  - Contains the registry entries needed for an iSeries Navigator plug-in written in Java. The registry entries inform the Navigator that a Java plug-in exists, and identify the names of the Java classes which implement the interfaces defined for a plug-in.

IBM @server. For the next generation of e-business.

Excerpts from a Sample Setup.ini file *(Notes)*

IBM @server iSeries

Setup.ini - File that provides the information needed to install, upgrade, and service an Operations Navigator plug-in.

```
[Plugin Info]
Name=Java Message Queue Sample 3
NameDLL=MessageQueuesMRI.dll
NameResID=1
Description=Operations Navigator Java Plug-in sample 3
DescriptionDLL=MessageQueuesMRI.dll
DescriptionResID=2
Version=0
VendorID=IBM.JavaSample
SupportExpress=YES
;-----------------------------------------------------------------
; The Service section provides the following information to the
; Client Access Selective Setup and Login Service Check programs:
;[Service]
FixLevel=0
AdditionalSize=0
 ;File sections can be empty.  Those file sections are simply ignored.
;-----------------------------------------------------------------

; The Base Files section
;-----------------------------------------------------------------

[Base Files]

1=javasample.jar
```

IBM @server. For the next generation of e-business.

Excerpts from setup.ini continued...*(Notes)*

IBM @server iSeries

```
;-----------------------------------------------------------------
; The Java MRI Files section (example for Canadian French) - optional if you translate to other languages
;-----------------------------------------------------------------
[JavaMRI2981]
1=javasamplemri_fr_CA.jar

;-----------------------------------------------------------------
; The Help Files section
;-----------------------------------------------------------------
[Help Files]

;-----------------------------------------------------------------
; The Registry Files section - this is required
;-----------------------------------------------------------------
[Registry Files]
1=MsgQueueSample3.reg


;-----------------------------------------------------------------
; The Install Dependencies section - The AS/400 Java Toolbox is listed
; as a dependency below.
```

IBM @server. For the next generation of e-business.

How can I build a plug-in?

IBM @server iSeries

Step 6: Debugging your application

IBM @server. For the next generation of e-business.

Checklist of Most Common Problems

IBM @server iSeries

- ☑ **If the tools do not come up, the classpath is probably not specified correctly.**
- ☑ **Wrong version of Java**
- ☑ **Wrong version of Swing**
- ☑ **If help doesn't come up, ensure you are pointing to jhall.jar in your classpath**
  - ▶ C:\Program Files\IBM\Client Access\JRE\Lib\jhall.jar

Release	JRE	Swing	JavaHelp
V5R1	JRE 1.3	none - swing is included in JRE 1.3	jhall.jar 1.1.1
V4R5	JRE 1.1.8	Swing 1.1	
V4R4	JRE 1.1.7	Swing 1.0.3	

IBM @server. For the next generation of e-business.

Where to find error details

IBM @server iSeries

- Graphical Toolbox writes error messages to a log file specified by the application
- PCML writes error messages to a log file specified by the application
- iSeries Navigator writes error messages to the iSeries Access for Windows History Log & Detail Trace Log
- Uses Trace class from the Access package of the Java ToolBox (com.ibm.as400.access.Trace)
- If problems occur, obtain all log files!

IBM @server. For the next generation of e-business.

Serviceability API- *(Notes)*

IBM @server iSeries

Trace class methods for logging data:

log(int, String)
Logs a message in the specified category.

log(int, String, boolean)
Logs a message and a boolean value in the specified category.

log(int, String, byte[])
Logs a message and byte data in the specified category.

log(int, String, byte[], int, int)
Logs a message and byte data in the specified category.

log(int, String, int)
Logs a message and an integer value in the specified category.

log(int, String, Throwable)
Logs a message in the specified category.

log(int, Throwable)
Logs a message in the specified category.

IBM @server. For the next generation of e-business.

How can I build a plug-in?

IBM @server iSeries

Step 7: Installing your plug-in

IBM @server. For the next generation of e-business.

Java Plug-in Installation

IBM @server iSeries

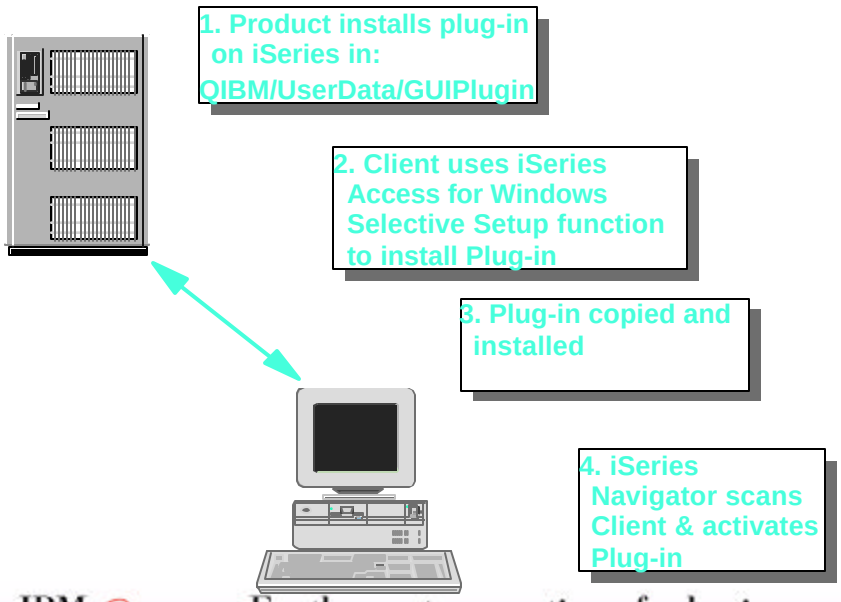
- iSeries Navigator Install
  - iSeries Navigator is shipped as part of iSeries Access for Windows at no additional charge.
- Your plug-in's install
  - Install from an iSeries using iSeries Access for Windows Selective Setup
  - Will have it's own install option
  - Installation of JRE, Swing and Toolbox for Java is handled by iSeries Access for Windows

Release	JRE	Swing	JavaHelp
V5R1	JRE 1.3	none - swing is included in JRE 1.3	jhall.jar 1.1.1
V4R5	JRE 1.1.8	Swing 1.1	
V4R4	JRE 1.1.7	Swing 1.0.3	

IBM @server. For the next generation of e-business.

Installation

IBM @server iSeries



IBM @server. For the next generation of e-business.

What's Needed for Install?

IBM @server iSeries

- **Product Installs plugin on the Server**
  1. Include files and resources in an OS/400 licensed program
  2. The licensed program should be installed into the IFS directory on the AS/400 in: QIBM/UserData/GUIPlugin
- **Setup.ini file - determines where the plugin will be installed in iSeries Navigator**
- **The plugin will be displayed in iSeries Access for Windows Selective Install**
  - The user checks the plugin to install it
- **When a user installs a plug-in, Selective Setup uses a setup file in the plug-in's subdirectory to identify and locate the files to be copied to the user's workstation.**
  - Java plug-ins, Selective Setup copies the plug-in's JAR files to the locations specified in the setup file.

IBM @server. For the next generation of e-business.

Install Notes

IBM @server iSeries

►**Product Installs plugin on the Server**

1. **Include files and resources in an OS/400 licensed program**
2. **The licensed program should be installed into the IFS directory on the AS/400 in:**

If your product only supports iSeries Access for Windows:
QIBM/UserData/OpNavPlugin

If your product supports the XD1 client & iSeries Access for Windows:
QIBM/UserData/GUIPlugin

IBM @server. For the next generation of e-business.

Scanning after Install

IBM @server iSeries

►After a plug-in has been installed, iSeries Navigator performs a "scan" function the first time the user selects an iSeries in the Navigator hierarchy.

►The scan operation uses information that is supplied by the plug-in about its software prerequisites to determine whether the plug-in's function should be activated for that iSeries.

►Following completion of the scan operation, the iSeries Navigator saves the status of the plug-in (whether to activate the plug-in for a particular AS/400) locally on the user's workstation.

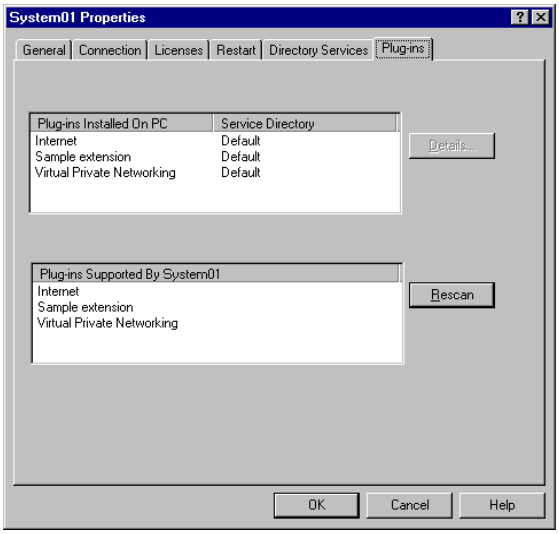
IBM @server. For the next generation of e-business.

Displaying list of installed Plugins

IBM @server iSeries

How do I know what plugins are installed?

1. In the Operations Navigator window, right-click on an AS/400 and select Properties.
2. In the Properties dialog, select the Plug-ins tab. The installed Plug-ins are listed.



IBM @server. For the next generation of e-business.

Where can I get more Information?

IBM @server iSeries

IBM @server. For the next generation of e-business.

Samples for developers

IBM @server iSeries

- The iSeries Access for Windows Toolkit contains samples for:
  - Java (recommended)
  - C++
  - Visual Basic

★ Note, the iSeries Access for Windows Toolkit is an optionally installable component of iSeries Access for Windows.

IBM @server. For the next generation of e-business.

Redbooks:

IBM @server iSeries

Redbooks available at: http://www.redbooks.ibm.com/

- **AS/400 XML in Action: PDML and PCML , SG24-5959-00**
  - The entire book is available on the web in PDF format.
- **Building AS/400 Client/Server Applications with Java SG24-2152-02**

IBM @server. For the next generation of e-business.

For more information.....

IBM @server iSeries

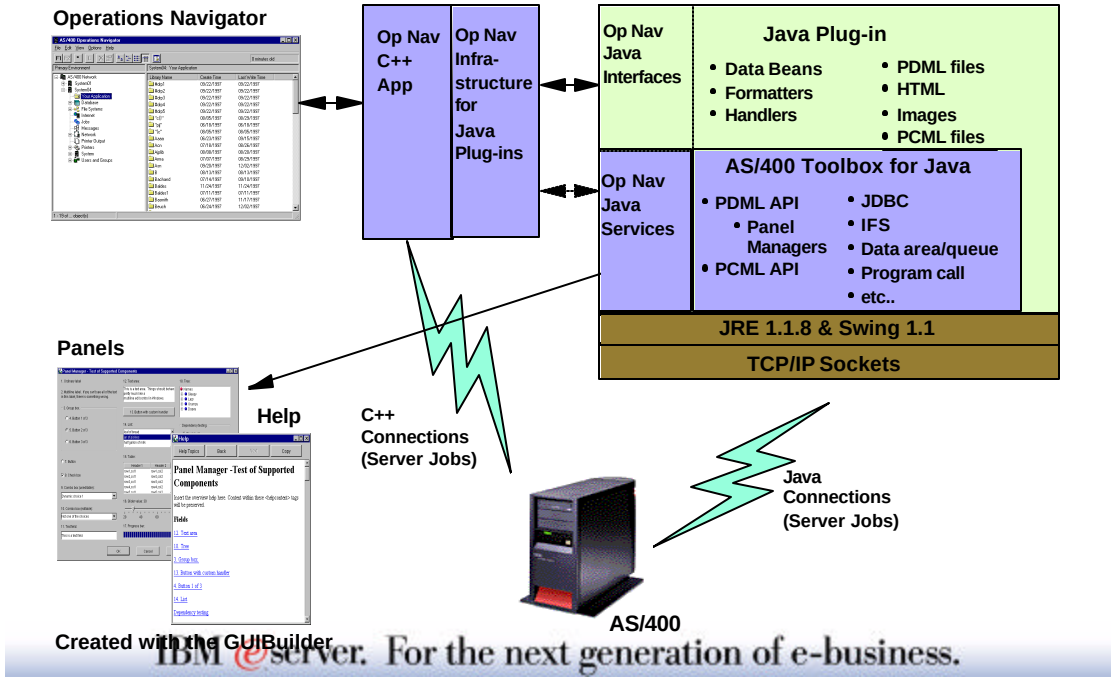
- On-line help for the GUI Builder and Resource Script Converter
- Info Center
  - For Plug-in information via the Client Access Programming links
    - http://publib.boulder.ibm.com/pubs/html/as400/v5r1/ic2924/info/index.htm
  - For the Graphical Toolbox and PCML via the Java and Toolbox for Java links
    - http://publib.boulder.ibm.com/pubs/html/as400/v5r1/ic2924/info/java/rzahh/toolbox.htm
    - Then take links for Graphical Toolbox or PCML
- Toolbox for Java website: http://www.as400.ibm.com/toolbox/
- iSeries Navigator website: http://www.as400.ibm.com/oper_nav/
- DeveloperWorks website (to obtain JDK):
  - http://www-106.ibm.com/developerworks/

IBM @server. For the next generation of e-business.

IBM @server iSeries

IBM @server. For the next generation of e-business.

Operations Navigator with Java Plug-ins



How your plugin fits with iSeries Navigator