AS/400e

IBM Graphical Toolbox for Java (A component of the AS/400 Toolbox for Java)

Scott Sylvester

AS/400 Software Development

(sylvests@us.ibm.com)

IBM

Agenda

- **Discuss building graphical user interfaces using Java and building AS/400 Java toolbox program call objects**
  - **A set of tools and runtime framework for building GUIs using XML that are rendered using Java**
  - **A runtime framework for building AS/400 program call toolbox objects using XML**

- **Discuss AS/400 Operations Navigator plug-in enhancements**
  - **AS/400 Operations Navigator plug-in support for Java GUI components**
  - **AS/400 Operations Navigator plug-in support for VisualBasic GUI components**

- **Discuss running and packaging of the various components**

IBM Graphical toolbox for Java

- Two new features packaged with the AS/400 Toolbox for Java (Mod 2 ships with V4R4)
  - IBM Graphical Toolbox for Java
    - Tools and a runtime framework to develop Graphical User Interfaces (GUIs) in Java using the eXtensible Markup Language (XML)
  - Program Call Markup Language (PCML) and runtime framework
    - An XML language and framework for building and using AS/400 Java Toolbox program calls

- In V4R4M0, Client Access Express now has the capability to selectively install the AS/400 Toolbox for Java

IBM Graphical toolbox content

- The Graphical Toolbox consists of the following tools to help you create GUI panels in Java for retrieving, storing, and processing information to/from the AS/400:
  - GUIBuilder - You can use the GUI Builder to quickly and easily create GUIs in a WYSIWYG fashion
  - Resource Script (RC) converter - use the Resource Script converter to easily convert existing Windows-based GUIs to Java

- The GUIs you create can be used to complete:
  - Applets
  - Java applications
  - Plug into the AS/400 Operations Navigator

- Both of the Graphical toolbox tools build the GUIs by creating panels represented in the Panel Definition Markup Language (PDML)
  - PDML is a tag language based upon the eXtensible Markup Lanaguage (XML)

IBM Graphical Toolbox technology highlights

- **Graphical User Interfaces**
  - A platform and technology independent representation of graphical user interfaces based on the Extensible Markup Language (XML) - *Panel Definition Markup Language (PDML)*
  - A pure-Java framework for interpreting the PDML language and constructing graphical user interfaces based on the Java Foundation Classes (JFC)
 - A set of tools for building GUIs in XML
  - Based on Java Beans

- **AS/400 Program call data objects**
  - An XML language to assist in the building AS/400 toolbox objects that represent AS/400 program call parameters - *Program Call Markup Language (PCML)*
  - A pure-Java framework for interpreting the PCML language and retrieving/storing information using the AS/400 Toolbox for Java

IBM Graphical Toolbox GUIBuilder and Presentation Manager
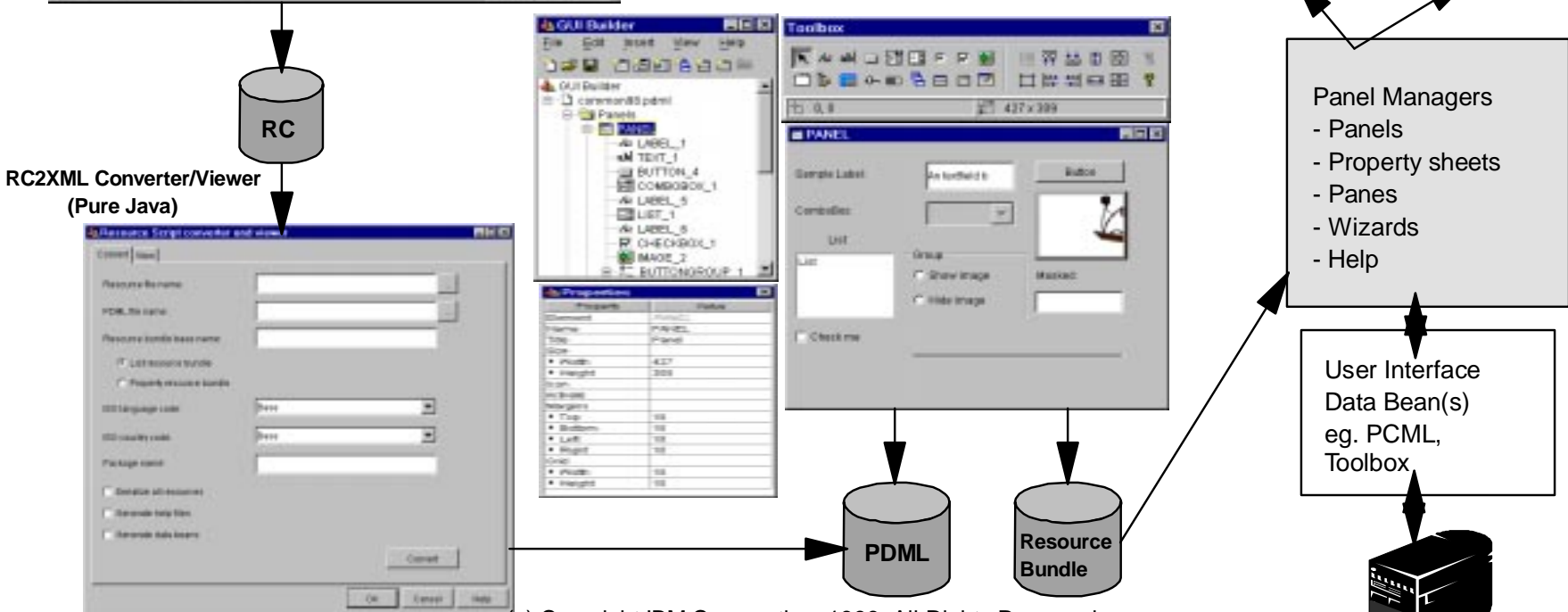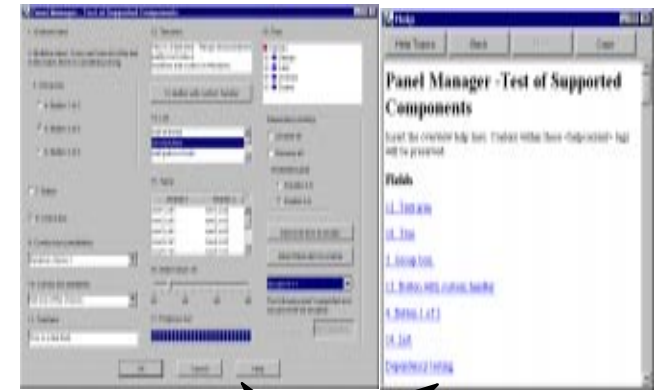
GUI Development

(Build-time)

End User Interface
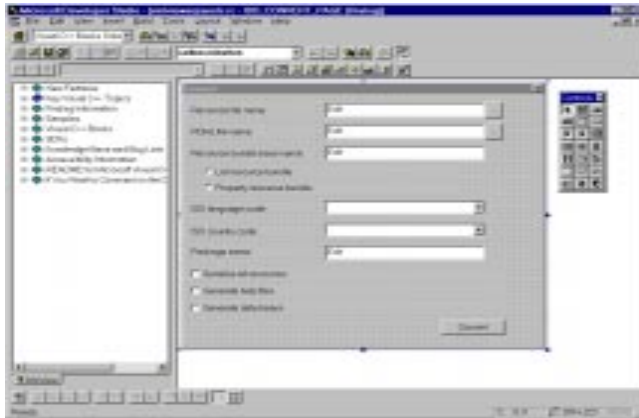
(Run-time)

MS VC++/VisualAge
(Resource Editor)

OR

PDML GUIBuilder
(Pure Java)

RC

RC2XML Converter/Viewer
(Pure Java)

Panel Managers
- Panels
- Property sheets
- Panes
- Wizards
- Help

User Interface
Data Bean(s)
eg. PCML,
Toolbox

PDML

**Resource
Bundle**

AS/400 XML Panel Definition Markup Language

- **Panel Definition Markup Language (PDML) is a tag language defined using the eXtensible Markup Language standard (XML)**
  - Extensible Markup Language (XML) is a evolving industry standard defined by the World Wide Web Consortium (W3C)
    - ▶ Subset of the Standardized Generalized Markup Language (SGML) that is targeted for use on the Web
  - XML defines the grammar for creating your own markup language for a specific purpose
  - XML grammar is checked to be well-formed and validated via an XML parser
  - Define a set of tags similar to HTML
  - Parsed into its constituent parts by the IBM XML parser (written in Java) for use by Java beans/applets/applications (available from IBM Alphaworks - www.alphaworks.ibm.com)
  - XML information available at: www.software.ibm.com/xml

Why AS/400 XML Panel Definition Markup Language?

- ‣ The PDML was created to:
  - Provide a representation of GUIs that is platform and UI technology independent
  - Remove the need to write/generate Java code for creating and displaying GUIs
  - Remove the need for each change to the panel layout requiring recompilation of Java code
  - Provide a representation of graphical user interfaces that can adapt to a device's user interface display capabilities
  - Provide a representation of graphical user interfaces that other user interface technology can be converted to (i.e. .RC files)

- ‣ The PDML provides the following benefits:
  - GUIs are represented in a descriptive tag language providing MRI separation
  - Developers can have precise control over the sizes and positions of user interface components
  - Scaling of GUIs for different screen resolutions
  - Changes to the panel layout do not require any code to be recompiled/regenerated
  - Unique versions of a given panel may be easily created for each national language lanaguage supported by an application developer.
  - National language translators do not need specialized skills to modify the panel definitions

IBM Graphical Toolbox GUI build-time contents

- 3 ways to build panels in PDML:
  - Windows tools can be used to create panels
    - ► Pure-Java tools supplied to convert .RC files to .PDML and Resource Bundle (Property/List)
    - ► PDML preserved for items directly edited (i.e. comments, supplemental tags (enable/disable), etc.)
  - Pure-Java WYSIWYG GUI builder can be used to create panels
    - ► Produces .PDML and Resource Bundle (Property/List)
    - ► PDML preserved for items directly edited (i.e. supplemental tags (enable/disable), etc.)
  - Manually type in PDML and Resource Bundle support
- Panels can be serialized for better performance at run-time
- MRI separation for Internationalization via resource bundles (Property/List)
- HTML On-line help skeletons automatically generated for each panel
- Additional PDML tags supported for simpler GUI panel building (PropertySheet, Wizard, Nested panels, Enable/Disable, Selected/Deselected, Data validation, etc.)

IBM Graphical Toolbox PDML GUI RC Converter
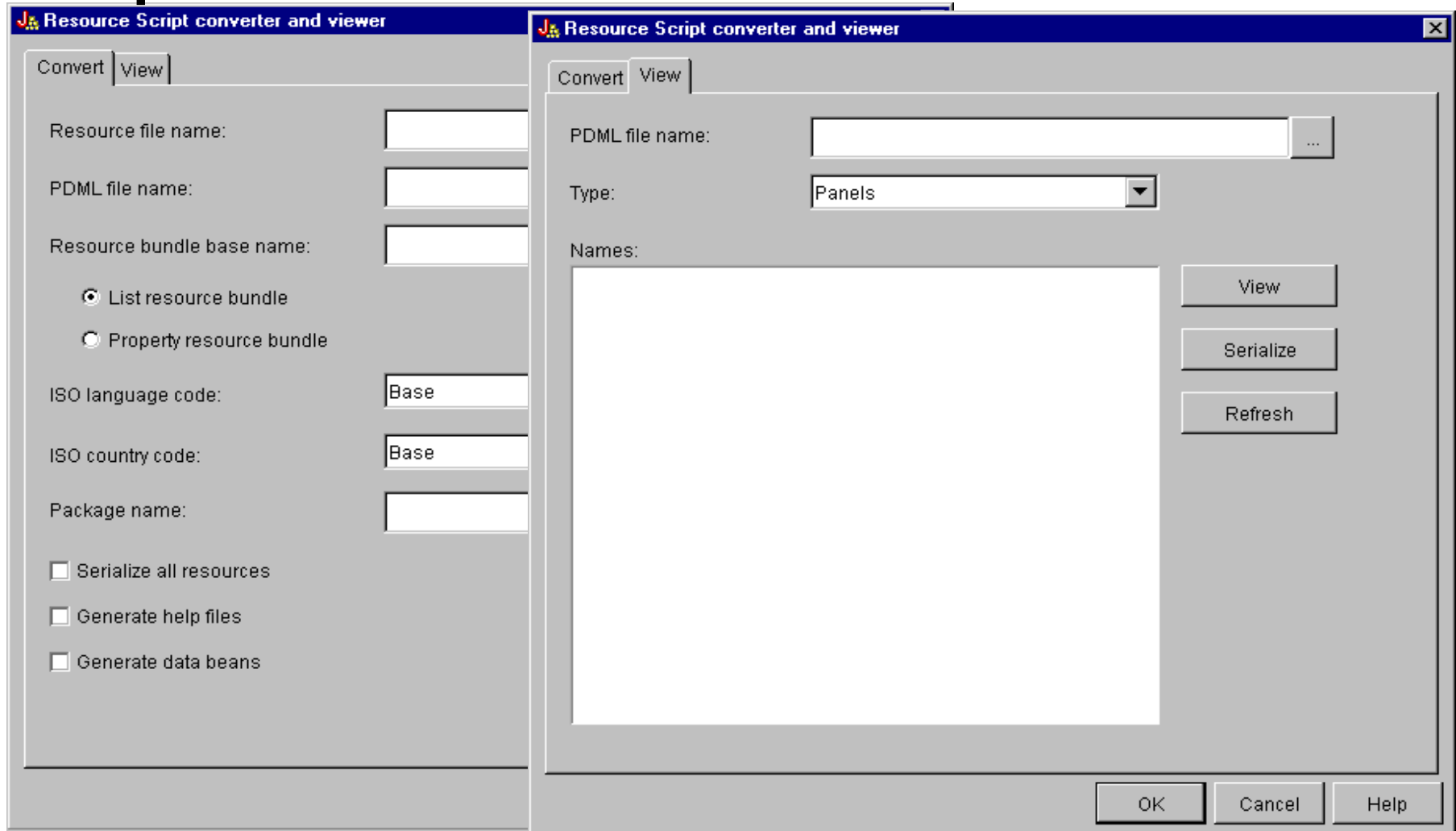
- .RC files converted to PDML using RC2XML converter/viewer "pure-java" tool (Microsoft and VisualAge for Windows)
- Converter preserves PDML tags manually added
- Resource Bundles can be created in any ISO language and country locale
- Panels can be converted and viewed
- Command line version also available for batch conversions

IBM Graphical Toolbox PDML GUIBuilder

- **Panels can be built using "pure-java" WYSIWYG GUI builder**
- **GUI builder reads and writes PDML and resource bundle using runtime package**
- **Resource Bundles can be created in any ISO language and country locale**
- **Properties edited via property editor (non-UI tags also supported, i.e. Selected/Deselected, etc.)**
- **GUI components are shown in tree below panel**
- **Panels converted via RC converter can be read into GUI Builder and then changed or the GUIBuilder can RC files directly**
- **Multiple PDML files can be edited from within the GUIBuilder**
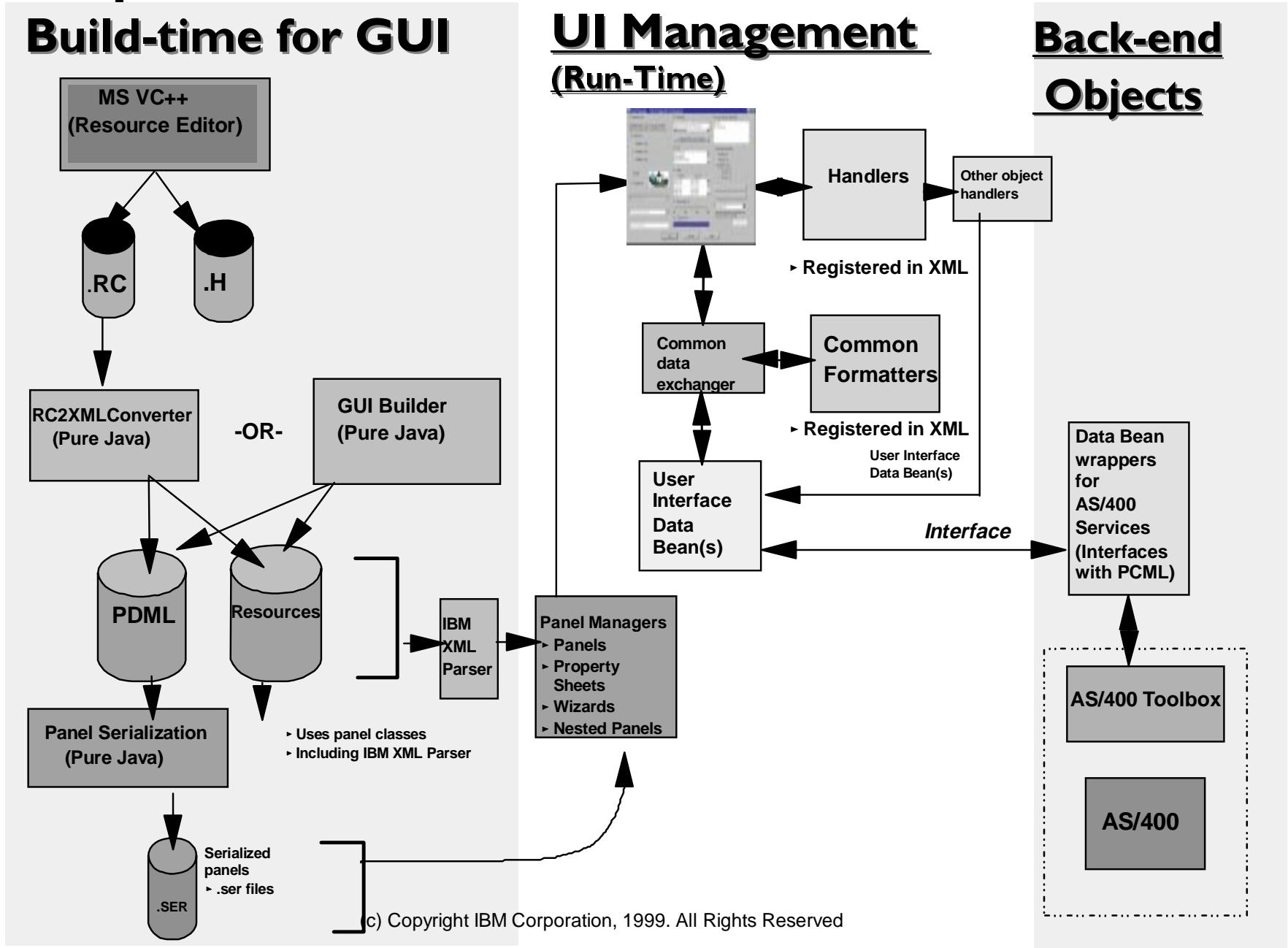- **GUIs can be previewed from within the GUI Builder**

IBM Graphical Toolbox GUI run-time contents

- Java runtime framework for rendering GUIs using the Java Foundation Classes (JFC)
  - Parsing of PDML (using the IBM XML parser integrated) or Serialized PDML
  - Panel layout manager supplied that supports dialog units or pixels
  - Data exchanger provided for automatic data exchange between user interface panel and Java beans via introspection
  - GUI managers supplied for rendering:
    - Panels, PropertySheets (TabbedPanes), Wizards, Multiple Panes - DeckPanes, TabbedPanes, SplitPanes, On-line help HTML information
  - Data formatter classes supplied for data validation (can be extended)
  - User interface component level event handlers supplied (can be extended)
  - Look and feel handled at runtime by GUI managers via Pluggable Look and Feel (PLAF) in JFC
    - "Windows looks like Windows"
    - "AIX looks like AIX"

AS/400 Data access using PCML

- Java runtime framework for supporting distributed AS/400 program call to AS/400 APIs using the AS/400 Toolbox for Java
  - Parsing of PCML (using the IBM XML parser integrated) or Serialized PCML
  - Java classes are supplied that use the PCML and AS/400 Toolbox for Java classes
  - The AS/400 Distributed Program Call (DPC) server is used to handle the remote program calls
  - For increased performance, a serialized version of the parsed PCML can be used to construct the remote program call
  - The PCML program call support can run on the client or the AS/400 JVM
  - A *ProgramCallDocument* class is supplied with the following functions
    - ▶ ProgramCallDocument.setValue - sets an input parameter value
    - ▶ ProgramCallDocument.callProgram - calls a program on the AS/400
    - ▶ ProgramCallDocument.getValue - gets an output value

IBM Graphical Toolbox functional flow

Build-time for GUI

UI Management (Run-Time)

Back-end Objects

MS VC++ (Resource Editor)

.RC

.H

RC2XMLConverter (Pure Java)

-OR-

GUI Builder (Pure Java)

PDML

Resources

IBM XML Parser

Panel Managers
- Panels
- Property Sheets
- Wizards
- Nested Panels

Panel Serialization (Pure Java)

▸ Uses panel classes
▸ Including IBM XML Parser

Serialized panels
▸ .ser files

.SER

Handlers

Other object handlers

▸ Registered in XML

Common data exchanger

Common Formatters

▸ Registered in XML

User Interface Data Bean(s)

User Interface Data Bean(s)

Interface

Data Bean wrappers for AS/400 Services (Interfaces with PCML)

AS/400 Toolbox

AS/400

IBM Graphical Toolbox for Java Installation

- The Graphical toolbox is installed with AS/400 Toolbox for Java

- The AS/400 Toolbox for Java can be installed via two methods
  - Install the AS/400 Toolbox for Java with Client Access Express
  - Or, install just the AS/400 Toolbox for Java (newest versions available from the web page www.as400.ibm.com/java)

- The AS/400 Toolbox for Java requires:
  - Java Runtime Environment (JRE) 1.1.6 or higher
  - Swing/JFC 1.0.3
  - Note, the JRE and Swing are installed with Client Access Express
  - If you are developing Java components, you will need the Java Developers Kit (JDK 1.1.6 or higher) and Swing/JFC 1.0.3

- The Graphical Toolbox is packaged with AS/400 Toolbox for Java - JC1

IBM Graphical Toolbox for Java Configuration and Commands

- The correct .JAR files must specified in the CLASSPATH
  - Java runtime (either from Client Access Express or JDK)
  - Swing/JFC, swingall.jar (Either from Client Access Express or Swing installation)
  - jt400\lib\jui400.jar (IBM Graphical toolbox runtime framework)
  - jt400\lib\util400.jar (IBM Graphical toolbox utility classes)
  - jt400\lib\uitools.jar (IBM Graphical toolbox GUI build-time)
  - jt400\lib\x4j400.jar (IBM XML parser)
  - jt400\lib\data400.jar (IBM AS/400 Program call data Access using PCML)
  - jt400\lib\jt400.jar (IBM AS/400 Toolbox for Java runtime)

- To invoke the GUI Builder from the command line:
  - java com.ibm.as400.ui.tools.GUIBuilder

- To invoke the Resource Script converter from the command line
  - java com.ibm.as400.ui.tools.PDMLViewer

IBM Graphical Toolbox for Java Documentation

- On-line HTML help for GUIBuilder and Resource Script Converter tools

- AS/400 Toolbox for Java documentation

- Client Access Express Toolkit

- Web samples

- Info-Center (used to be Client Access API and Technical Reference manual)

- URLs
  - http://www.ibm.com/as400/toolbox
  - http://iws.as400.ibm.com/toolbox/welcome.htm
  - http://infocntr/pubs/html/as400/v4r4/ic2924/info/java/rzahh/toolbox.html
    - ▶ Then go to the IBM Graphical Toolbox

IBM Graphical Toolbox for Java benefits

- **Through the Graphical toolbox**
  - You now have the ability to easily create GUIs that operate from platform to platform while eliminating the following:
    - ► The need to write Java GUI code
    - ► The need to recompile code after each change
  - As a result, less time is required for creating and managing GUIs

- **Through the PCML**
  - You now have the ability to describe your program calls to the AS/400 eliminating the following:
    - ► The need to write Java toolbox program call objects
    - ► The need to write Java code in your Java applications to construct AS/400 toolbox Java class objects for connecting to and retrieving information from an AS/400

IBM Graphical Toolbox for Java - Samples

AS/400 Operations Navigator Java and Visual Basic Plug-in support

- Java Runtime Environment (JRE) in Client Access
- Java Virtual Machine (JVM) will be started by the AS/400 Operations Navigator
- AS/400 Toolbox for Java runtime support (including new PDML graphical user interface runtime and PCML data access) will be included in Client Access
  - The PDML build-time tools will be included in the AS/400 Toolbox for Java
- AS/400 Operations Navigator Java enablement plug-in support includes:
  - Plugging into the AS/400 Operations Navigator tree (code can be written in Java)
  - Plugging into the AS/400 Operations Navigator lists (code can be written in Java)
  - Providing context menus for the AS/400 Operations Navigator (code can be written in Java)
  - Providing toolbars for the AS/400 Operations Navigator (code can be written in Java)
  - Graphical User Interface components can be written in Java
  - Remote AS/400 program calls can be built using the PCML

  (Note: AS/400 Operations Navigator components that provide tree items or toolbars require a windows resource DLL for the icons, bitmaps and strings)

IBM Graphical Toolbox for Java and AS/400 Operations Navigator Java enablement

Easy to implement graphical user interfaces in Java using XML as well as integrating them into the AS/400 Operations Navigator

Extending Simplicity to GUI development
in support of e-business

Notice

- This publication may refer to products that are not currently available in your country. IBM makes no commitment to make available any products referred to herein.
- Trademark and service marks:
  - AS/400, IBM, OS/400, AIX, OfficeVision, PROFS, OS/2, Facsimile Support/400 and APPN are trademarks of the IBM Corporation in the United States or other countries or both.
  - LOTUS, LOTUS Notes, cc:Mail, and cc:Mail for the Internet are trademarks of the LOTUS Development Corporation.
  - JAVA are registered trademarks of Sun Microsystems, Inc.
  - Microsoft, Windows, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.
  - Other company, product and service names may be trademarks or service marks of others.