

va

n@us.ibm.com



Kim Button – button@us.ibm.com

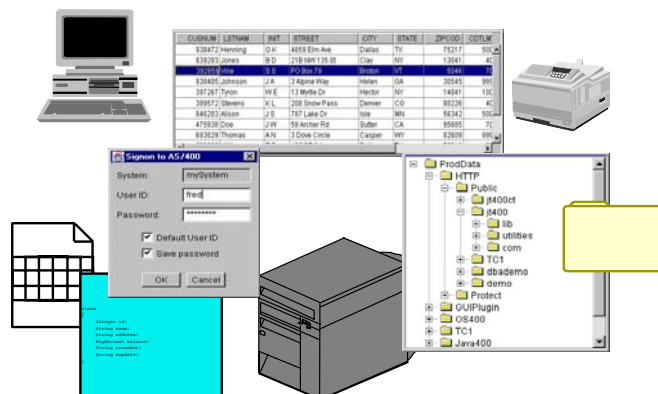
© Copyright IBM Corporation, 2008. All Rights Reserved.
This publication may refer to products that are not currently available in your country. IBM makes no commitment to make available any products referred to herein.

IBM Power Systems



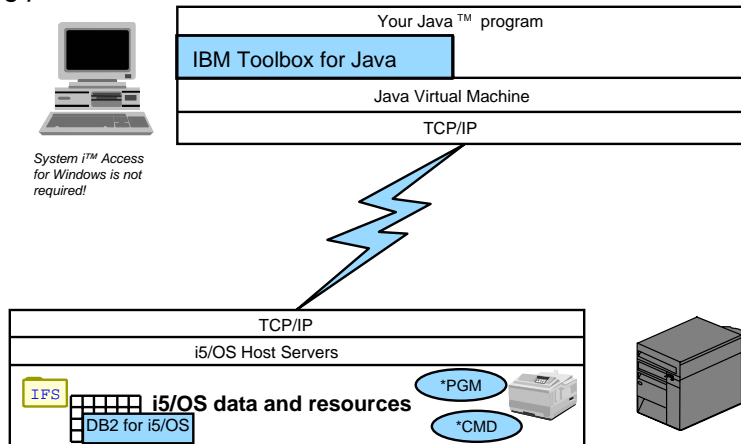
What is the Toolbox/JTOpen?

A set of Java classes and utilities which provide access to i5/OS® data and resources



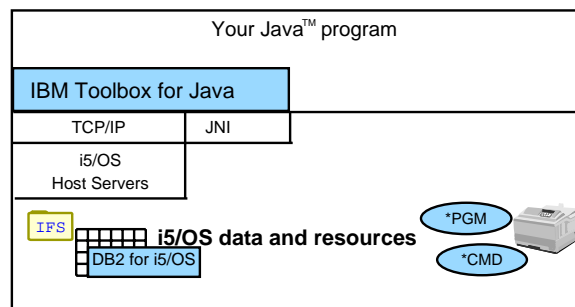
IBM Toolbox for Java

The big picture - Client/Server



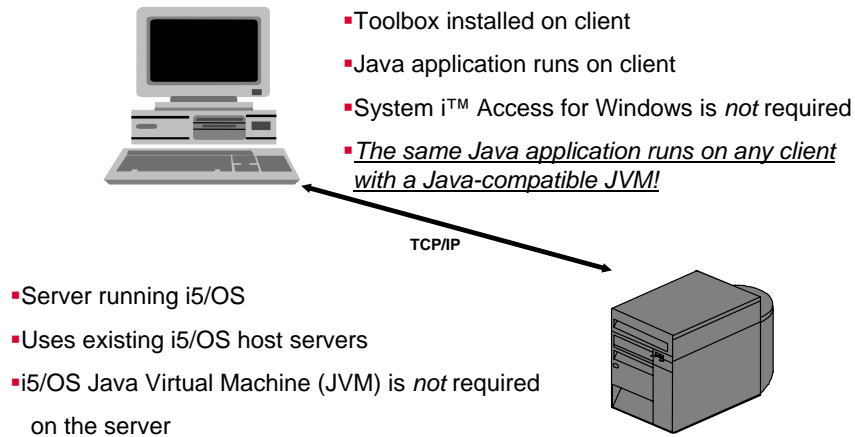
IBM Toolbox for Java

The big picture - Toolbox and data on same i5/OS



IBM Toolbox for Java

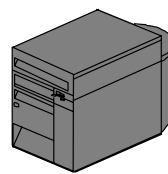
Using the Toolbox in client/server applications



IBM Toolbox for Java

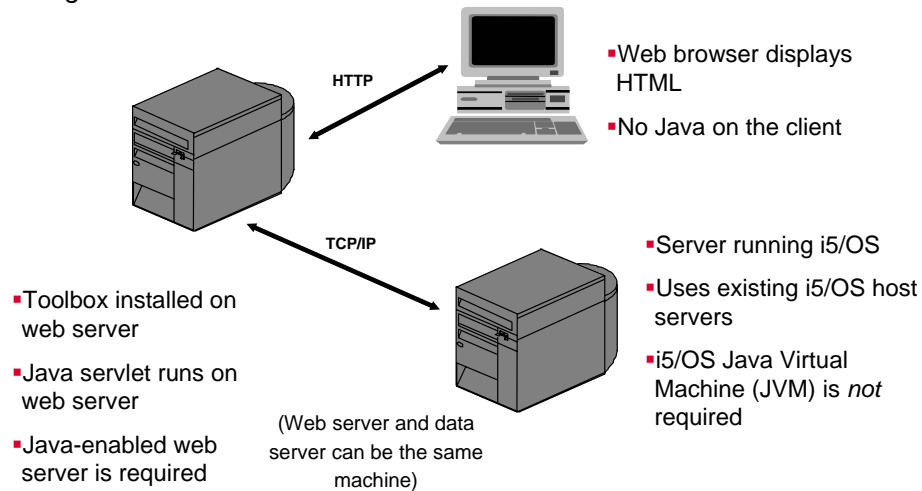
Using the Toolbox in server applications

- Toolbox installed on i5/OS
- Java application runs on i5/OS
- Use Toolbox instead of JNI
- "Local" sockets used to communicate between Toolbox and servers
- Direct API calls used in some cases to avoid the servers
- i5/OS with Java Virtual Machine (JVM) is required



IBM Toolbox for Java

Using the Toolbox in Internet-based servlets



IBM Toolbox for Java

Supported platforms:

- i5/OS, Linux®, Windows®, AIX®, Solaris, Netscape Communicator, Microsoft Internet Explorer

Requires Java 1.1.8 or greater, and supports Java 2

Divided into packages:

- com.ibm.as400.access - APIs for accessing i5/OS data and resources
- com.ibm.as400.resource - Framework for accessing list-based data
- com.ibm.as400.vaccess - GUI components
- com.ibm.as400.data - Program call markup language
- com.ibm.as400.ui.* - Graphical Toolbox
- com.ibm.as400.util.* - HTML, XSL-FO, and Servlet components
- com.ibm.as400.micro.* - APIs for wireless devices
- utilities - utility classes such as JarMaker, JPing, RunJavaApplication, AboutToolbox

IBM Toolbox for Java

Packaging

Licensed program 5722-JC1 (V5R3/V5R4) or 5761-JC1 (V6R1) (no additional charge)

Downloadable from JTOpen website (no additional charge)

Ships with i5/OS

Jar files:

- jt400.jar - Base function + GUI components
- jt400Native.jar - Base function only, intended for use on i5/OS JVM
- jt400Proxy.jar - Proxy support, subset of jt400.jar
- jt400Servlet.jar - HTML, XSL-FO, and Servlet components
- jt400Micro.jar - Wireless support
- uitools.jar, jui400.jar, util400.jar, x4j400.jar - Graphical Toolbox
- tes.jar - System Debugger



Use the JarMaker utility to reduce the size of jt400.jar or any other jar file

9

© 2008 IBM Corporation

JTOpen (Open Source)

All of the primary Toolbox packages are open source!

<http://jt400.sourceforge.net>

- Part of IBM's open source development community
- Use source as a debug tool
- Submit new function under the IBM Public License (IPL)
- Modify source for your use
- Submit problem reports and bug fixes

Two versions of the Toolbox:

- Licensed program
 - Supported by IBM
 - Only IBM developed code
- Open source version
 - Supported by open source community
 - Now officially supported by IBM Service!
 - Includes source from non-IBM contributors
 - New functions and fixes available here first!



10

© 2008 IBM Corporation

IBM Toolbox for Java

Popular Toolbox Functions

- Database access via JDBC
- Database access via a record-level I/O and DDS interface
- Command Call
- Program Call via both Java code and XML
- Data Queues / User Spaces / Data Areas
- Access files in iSeries Integrated File System
- Access Print object (spooled files, printers, queues, ...)
- Access iSeries objects (Jobs, Users, System Values, etc.)
- Built-in automatic data conversion
- HTML / Servlet wrappers
- Wireless APIs
- XML-based GUI Builder
- Components are Java Beans

IBM Toolbox for Java

i5/OS Functions Built on the Toolbox

- System i Navigator and Management Central
- System i Access for Web
- System i Connect (B2B)
- IBM Host On Demand
- Plus Many More...

IBM Toolbox for Java

Access Classes: Low-level Java APIs to Access Data

- User Authentication and Identification
- Command Call
- Connection Pools
- Clustered Hashtables
- Data Area
- Data Description
- Data Conversion
- Data Queues
- Environment Variables
- FTP
- IFS
- JDBC
- Jobs
- Messages
- NetServer
- Print
- Permissions
- Program Call
- Record-level Database Access
- Save File
- System Status
- System Values
- Users and Groups
- User Space

Infrastructure

"The AS400 object"

Represents a connection to the i5/OS

Provides a sign-on GUI

- Password caching available
- Change password GUI when appropriate

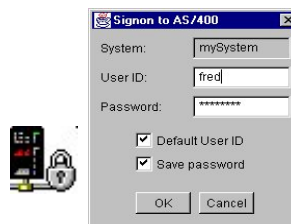
Controls conversations with server jobs

- Multiple users and multiple conversations
- Implicit and explicit connections

Provides Secure Sockets Layer (SSL) communication

- Encryption and server authentication

Most Toolbox classes use the AS400 object



```
AS400 sys = new AS400();
AS400 sys2 = new AS400("mySystem");
AS400 sys3 = new AS400("mySystem",
    "myUID", "myPWD");
```

```
CommandCall cc = new
CommandCall(sys);
```

JDBC

The Java standard for database access

Write Java programs in terms of standard JDBC interfaces, then plug in **any** JDBC driver - to work with **any** database!

- Java gives you platform independence, JDBC gives you database independence

java.sql package in Java Developers Kit

SQL is used extensively

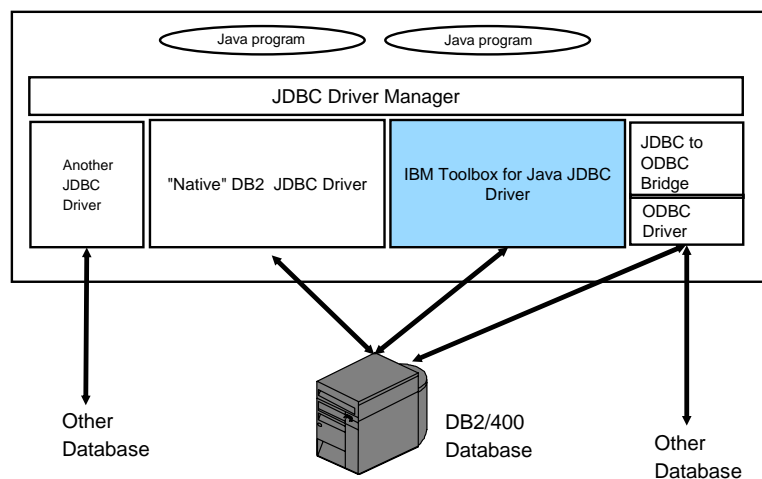
- Based on X/Open SQL Call Level Interface

Also supports:

- Database definitions, manipulations, and queries
- Stored procedures
- Catalog methods
- Transactions (commit, rollback, isolation levels, distributed)

JDBC

The Java standard for SQL database access



JDBC

i5/OS JDBC driver choices

Toolbox JDBC driver (*com.ibm.as400.access.AS400JDBCDriver*)

- Communicates with the database using TCP/IP
- Great for:
 - client/server applications
 - applets
 - servlets, where the web server and data are not on the same i5/OS

JDBC 4.0 support in JTOpen

“Native” DB2 JDBC driver (*com.ibm.db2.jdbc.app.DB2Driver*)

- Communicates with the database using direct CLI calls
- Great for:
 - server applications
 - servlets, where the web server and data on same i5/OS

```
DriverManager.registerDriver(...);
Connection c = DriverManager.getConnection(...);
Statement select = c.createStatement();
ResultSet rs = select.executeQuery("SELECT * FROM ...");
while (rs.next())
    System.out.println(rs.getString(column));
```

Record-level database access

Fast access to i5/OS database files

Provides access to database files:

- Access records sequentially, by record number, or by key
- Physical and logical file members are described by a RecordFormat
- Support for locking
- Support for transactions
- Familiar paradigm for RPG programmers
- Limited System/36 SSP file capability, too!

```
AS400 system = new AS400("mySystem");
SequentialFile file = new SequentialFile(system, "/QSYS.LIB/MYLIB.LIB/MYFILE.FILE");
file.setRecordFormat(...);
file.open(...);
Record r = file.readNext();
```

Integrated file system

File input, output, and more

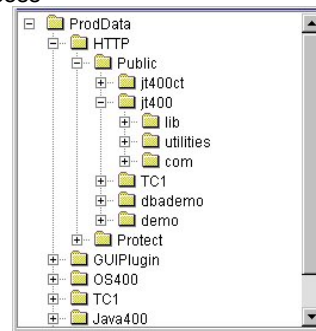
Implements standard Java input/output (*java.io*) classes:

- Read and write data sequentially or via random access
- Create, delete, and rename files and directories
- List the contents of a directory

```
AS400 system = new AS400();

IFSFileOutputStream s = new
    IFSFileOutputStream(system, "/a.a");
byte[] data = new byte[n];
s.write(data);

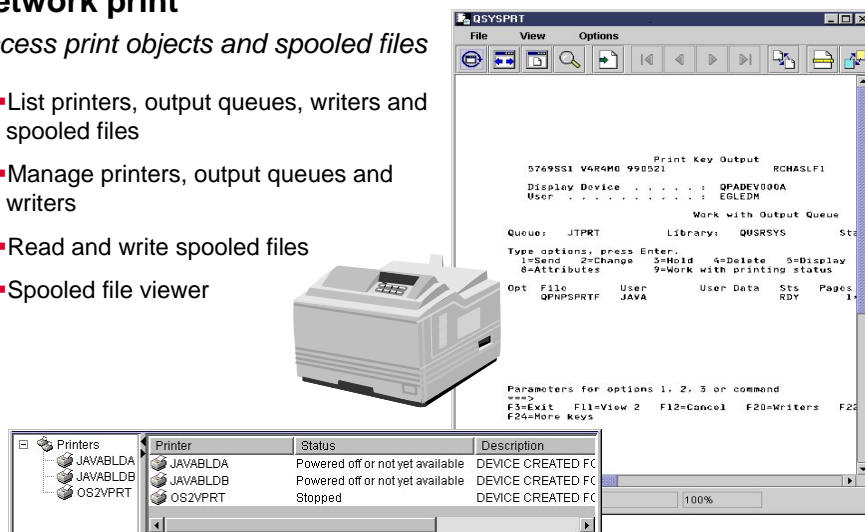
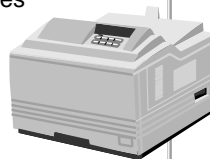
IFSTextFileOutputStream s2 = new
    IFSTextFileOutputStream(system, "/b.b", 37);
s2.write("Hi Mom in EBCDIC");
```



Network print

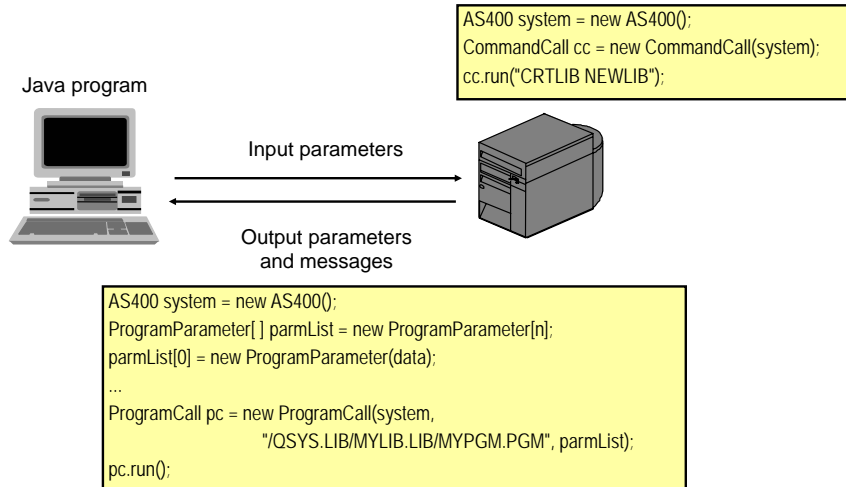
Access print objects and spooled files

- List printers, output queues, writers and spooled files
- Manage printers, output queues and writers
- Read and write spooled files
- Spooled file viewer



Command call and program call

Make use of legacy code and system APIs

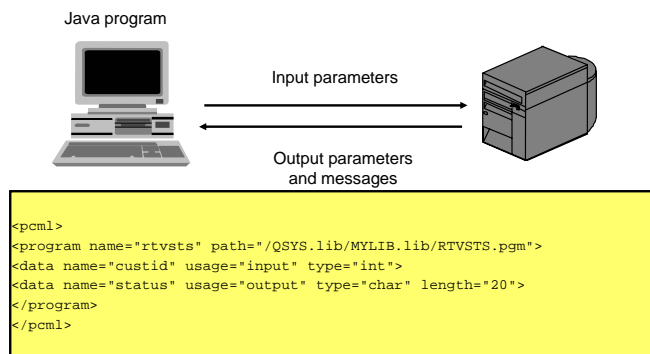


Program Call Markup Language (PCML)

Describe program calls using XML

Automates program call parameter and structure definition

Simplifies data description and conversion



Interprogram Communication

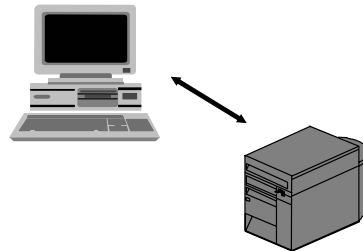
Data queues, Data areas, User spaces, ...

Transfer data between programs using:

- Data areas
- Data queues (keyed or sequential)
- User spaces

Partner can be Java program, traditional i5/OS program, on i5/OS or another client

Java program



```
AS400 system = new AS400();
UserSpace us = new UserSpace(system, "/QSYS.LIB/MYLIB.LIB/MYDATA.USRSPC");
byte[] data = new byte[1024];
us.read(data, 0);
```

RFML (Record Format Markup Language)

Very similar to PCML (Program Call Markup Language)

While PCML is designed only for Program Parameters, RFML is useful for parsing/composing:

- Data queue entries
- User spaces
- Physical file records
- Data buffers

Specify record formats using XML; get/set field values

Segregate the data layout from the program logic

RFML vs. FieldDescription

Example: Composing a customer record

Using RFML:

```
import com.ibm.as400.data.RecordFormatDocument;

RecordFormatDocument rfmlDoc =
    new RecordFormatDocument("customer");

( In a separate file named "customer.rfml": )

<rfml version="4.0" ccid="37">
  <recordformat name="cusrec">
    <data name="cusnum" type="int" length="2" precision="16"/>
    <data name="lstnam" type="char" length="8"/>
    <data name="baldue" type="zoned" length="6" precision="2"/>
  </recordformat>
</rfml>
```

Without RFML:

```
import com.ibm.as400.access.AS400Text;
import com.ibm.as400.access.AS400UnsignedBin2;
import com.ibm.as400.access.AS400ZonedDecimal;
import com.ibm.as400.access.BinaryFieldDescription;
import com.ibm.as400.access.CharacterFieldDescription;
import com.ibm.as400.access.RecordFormat;
import com.ibm.as400.access.ZonedDecimalFieldDescription;

RecordFormat recFmt = new RecordFormat("cusrec");

AS400UnsignedBin2 conv1 = new AS400UnsignedBin2();
BinaryFieldDescription desc1 = new BinaryFieldDescription(conv1, "cusnum");
recFmt.addFieldDescription(desc1);

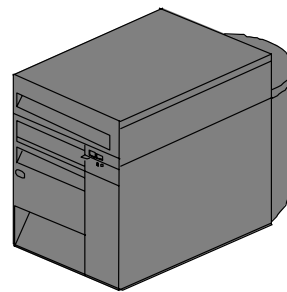
AS400Text conv2 = new AS400Text(8, 37);
CharacterFieldDescription desc2 = new CharacterFieldDescription(conv2, "lstnam");
recFmt.addFieldDescription(desc2);

AS400ZonedDecimal conv3 = new AS400ZonedDecimal(6, 2);
ZonedDecimalFieldDescription desc3 = new ZonedDecimalFieldDescription(conv3,
    "baldue");
recFmt.addFieldDescription(desc3);
```

Server Objects

Jobs, Users, System Values, ...

- List i5/OS jobs
- List i5/OS users and groups
- Display and change system values
- Manage message queues
- Manage user permissions to objects



```
AS400 system = new AS400();
SystemValue sv = new SystemValue(system, "QDATE");
System.out.println(sv.getValue());
```

Data description and conversion

Converts between Java data and i5/OS data

Java data type		i5/OS data type
Object[]		Array
short	↔	2 byte binary
int		2 byte unsigned binary
Int	↔	4 byte binary
long		4 byte unsigned binary
long		8 byte binary
byte[]	↔	Byte array
float		4 byte floating point
double		8 byte floating point
BigDecimal	↔	Packed decimal
BigDecimal		Zoned decimal
Object[]		Structure
String	↔	Text

Handles all code page, byte order, and data conversion issues!

Data description and conversion

Record formats

Access data in the record by field name

Convert data automatically for:

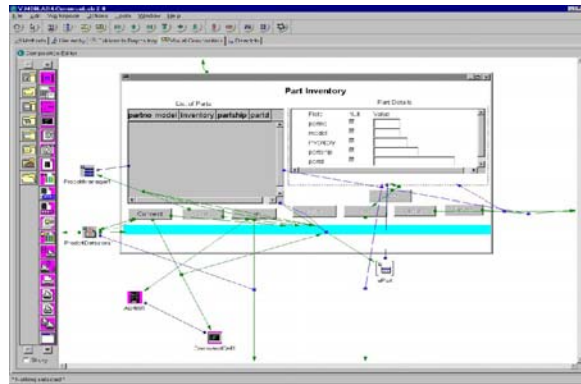
- Program calls
- Data queues
- Record-level database access

```
BinaryFieldDescription customerNumber = new BinaryFieldDescription(new AS400Bin4(),
                                                                    "CUSTOMER_NUMBER");
CharacterFieldDescription customerName = new CharacterFieldDescription
    (new AS400Text(20, system), "CUSTOMER_NAME");
RecordFormat recordFormat = new RecordFormat();
recordFormat.addFieldDescription(customerNumber);
recordFormat.addFieldDescription(customerName);
Record data = recordFormat.getNewRecord(dataQueue.read().getData());
Integer I = (INTEGER) data.getField("CUSTOMER_NUMBER");
String name = (String) data.getField("CUSTOMER_NAME");
```

IBM Toolbox for Java

Visual development environments

Most Toolbox public classes are Java Beans. With visual development tools like WDSC (WebSphere Developer Studio client), no coding necessary!



HTML and Servlet classes

Web components create tables and forms

Provides access to database files:

- Access database file with Record Level Access or SQL via JDBC
- Includes Meta Data

Provides classes to display data:

- Display data in tables or forms
- Toolbox provides converters that will produce HTML tables or forms based on the row data

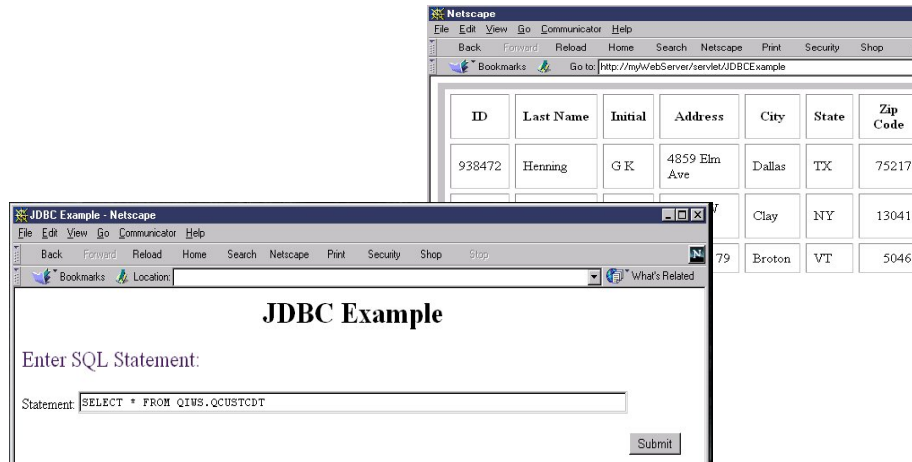
```
HTMLTableConverter converter = new HTMLTableConverter();

ResultSet resultSet = statement.getResultSet();
SQLResultSetRowData rowdata = new SQLResultSetRowData(resultSet);

String[] html = converter.convert(rowdata);
out.println(html[0]);
```

HTML and Servlet classes

Web components create tables and forms



HTML and Servlet classes

Web components create tree hierarchy

Provides classes to display the Integrated File System:

- Display contents of the Integrated File System
- Toolbox provides classes to create and display a customized and traversable tree

```
HTMLTree tree = new HTMLTree(HTTPrequest)

IFSJavaFile root = new IFSJavaFile(systemObject, "/QIBM");

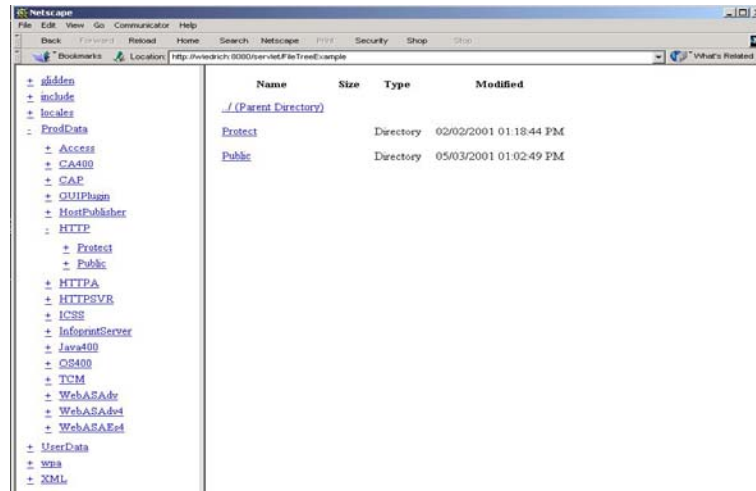
DirFilter filter = new DirFilter();

File[] dirList = root.listFiles(filter);

for (int i=0; i<dirList.length; i++)
{
    FileTreeElement node = new FileTreeElement(dirList[i]);
    tree.addElement(node);
}
```


HTML and Servlet classes

Web components create tree hierarchy



Toolbox Micro Edition

Terms

What is ToolboxME?

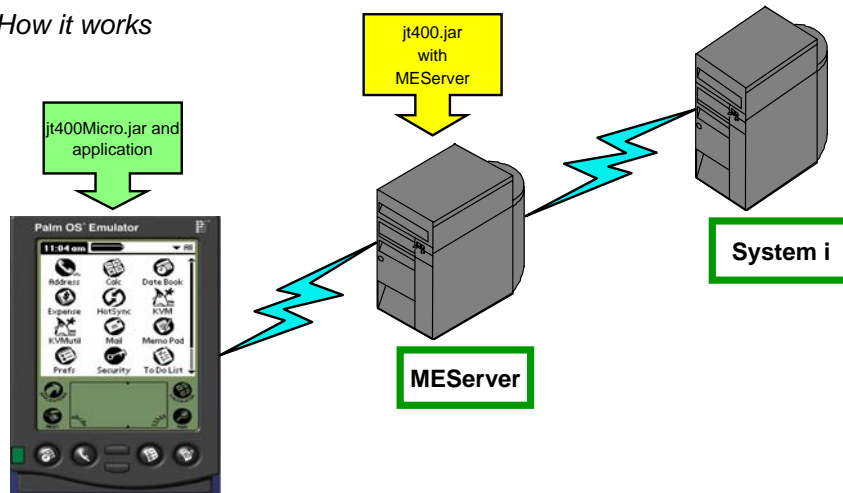
A subset of the Toolbox for Java classes which will provide access to i5/OS data and resources from a Tier 0 device

What is a Tier 0 device?

- **Tier 3:** the i5/OS server
- **Tier 2:** the application or web server
- **Tier 1:** the client desktop or laptop
- **Tier 0:** refers to the next level down. Tier 0 devices are predominantly thought of as web-enabled cell phones and personal digital assistants (PDAs).

Toolbox Micro Edition

How it works



Toolbox Micro Edition

Supported Components

- AS400
- Command Call
- Program Call via PCML
- Data Queues
- JdbcMe



Graphical Toolbox

Describe GUI panels using XML

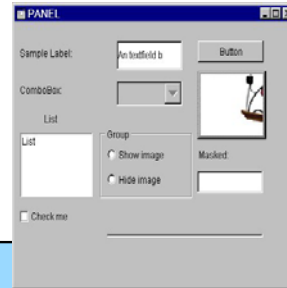
Panel Definition Markup Language (PDML)

- Simplifies GUI panel definition and layout

Resource script (RC) converter

- Converts Windows GUIs to Java

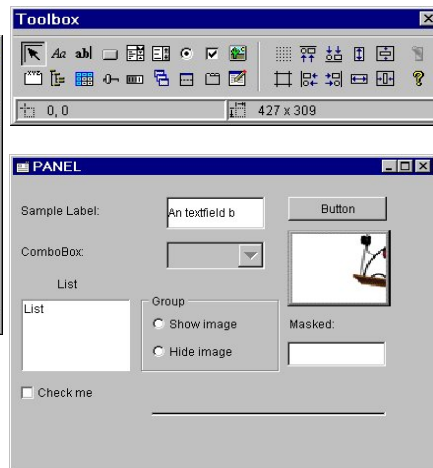
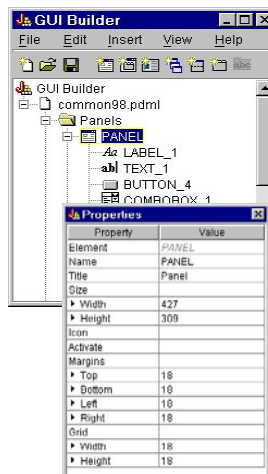
```
<pdml>
<panel name="order_entry" path="/QSYS.lib/MYLIB.lib/RTVSTS.pgm">
<title>Order Entry</title>
<button name="Ok" disabled="no">
  <title>Ok</title>
  <location>125,100</location>
  <size>100,26</size>
  <action>COMMIT</action>
</button>
</pdml>
```



Graphical Toolbox

GUI builder

- WYSIWYG panel definition
- Generates PDML code
- Now supports JavaHelp™



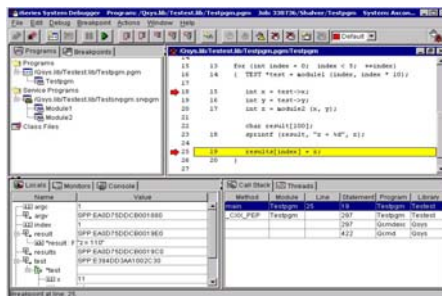
Other components

The list keeps growing!

- NetServer
- JarMaker
- i5/OS Messages
- Message files
- System status
- Proxy Support
- Save File
- Report Writer
- Servlets
- System Properties



System Debugger and Debug Manager



- Supports all ILE languages: C, C++, RPG, Java, Cobol, CL
- Point and click breakpoint manipulation in source code
- Automatic variable evaluation with mouse and local variable display
- Program call stack and thread display

• Requires JDK1.3 and tes.jar, jt400.jar, and jhall.jar

• Invoke with following: `java utilities.DebugMgr` OR `java utilities.Debug -s system -u user`

New in JTOpen and V6R1

Now available at

<http://www.ibm.com/systems/i/software/toolbox>

Enhancements and new classes

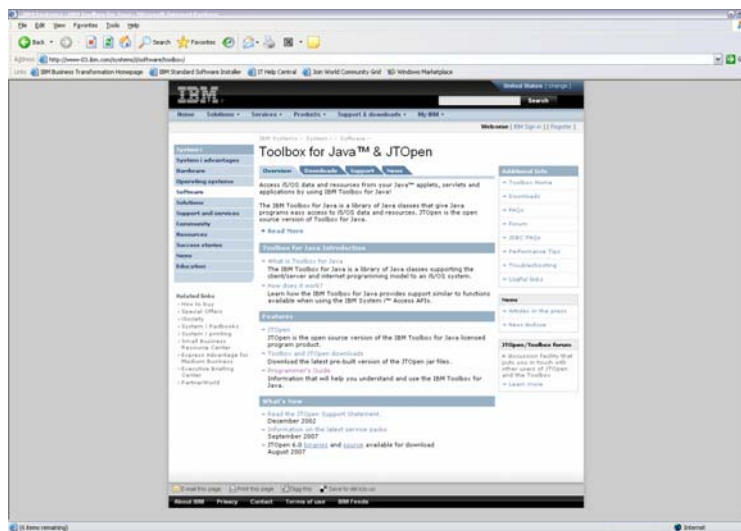
- JDBC enhancements including generated key support
- JDBC 4.0 support (JTOpen only)
- JDBC performance improvements
- AS400JDBCManagedConnectionPoolDataSource
- FileAttributes
- HistoryLog
- ObjectReferences
- UDFS



Plus

- CL command documentation generator
- Kerberos authentication is now supported through the use of JGSS
- Performance improvements in list processing (users, jobs, etc.)
- CL commands now Unicode enabled

IBM Toolbox for Java home page



IBM
Home Business solutions IT services Products Support & downloads My IBM

Search: Search: All topics

Contents

IBM Toolbox for Java

IBM Toolbox for Java™ is a set of Java classes that allow you to use Java programs to access data on your system. You can use these classes to write client/server applications, agents, and servlets that work with data on your system. You can also run Java applications that use the IBM Toolbox for Java classes on the System i™ Java virtual machine (JVM).

IBM Toolbox for Java uses the J2SE™ [Host Servlets](#) as access points to the system. Because IBM Toolbox for Java uses communication functions built into Java, you do not need to use IBM System i™ Access for Windows™ to use IBM Toolbox for Java. Each server runs in a separate job on the server, and each server job sends and receives data streams on a socket connection.

Note: By using the code examples, you agree to the terms of the [Code Sample and disclaimer information](#).

What's new for V6R1
This topic highlights the changes made to IBM Toolbox for Java in V6R1.

PDF file for IBM Toolbox for Java
You can view and print a PDF file of this information.

Class finder
Use the class finder to find information about the Toolbox for Java classes. You can find particular classes by searching the class name and description or by displaying a list of classes.

Installing and managing IBM Toolbox for Java
Using IBM Toolbox for Java makes it easier to write client Java [applets](#), [Servlets](#), and applications that access system resources, data, and programs.

IBM Toolbox for Java classes
The IBM Toolbox for Java classes are categorized, like all Java classes, into packages. Each package provides a certain kind of functionality.

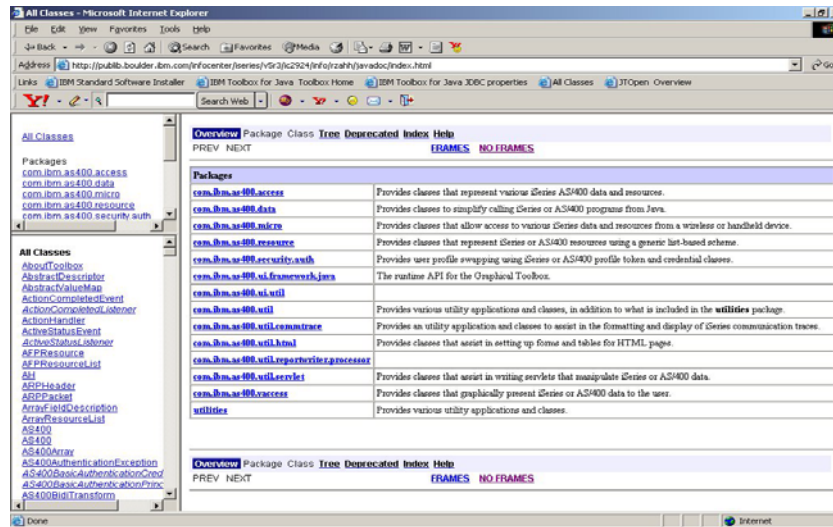
JavaBeans
Use the JavaBeans to find detailed information about the IBM Toolbox for Java classes.

Graphical Toolbox and PDI
The Graphical Toolbox, a set of UI tools, enables you to create custom user interface panels in Java.

IBM Toolbox for Java beans
JavaBeans™ are reusable software components that are written in Java. The component is a piece of program code that provides a well-defined, functional unit, which can be as small as a label for a button on a window or as large as an entire application.

JDBC
JDBC is an application programming interface (API) included in the Java platform that enables Java programs to connect to a wide range of databases.

Javadoc



References

Where can I get more information?

<http://www.ibm.com/systems/i/software/toolbox/>

- News, downloads, FAQs, service packs, articles, COMMON labs

<http://jt400.sourceforge.net/>

- JTOpen - open source, bug reporting, feature requests

<http://www.ibm.com/servers/eserver/support/series/index.html>

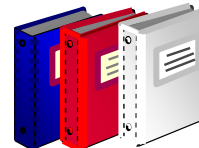
- System i5 Technical Forums - including IBM Toolbox for Java/JTOpen Forum

IBM Toolbox for Java Programmers Guide

- Shipped with the IBM Toolbox for Java
- Contains overview, full API documentation (javadoc), and code examples
- Available in the System i™ Information Center
- Link off of the Toolbox home page

Building AS/400 Client/Server Applications with Java

- Redbook SG24-2152-02



IBM Certification Testing – Here at COMMON!!

➤ Where and When ?

- Delta Island F
- 8:30 – 5:00 Tuesday and Wednesday
- 8:30 – 12:00 Thursday

➤ What's in it for me ?

- Portable credential
- Proof that you can "Walk the Talk"
- Peer and Employer recognition
- Industry recognition

➤ How much does it cost ?

- **NOT** \$190 that you pay at Prometric testing centers
- **Special Discounted price of \$95 !!**

➤ What tests are available?

- System i (of course)
- All "other" System Group platform tests (System p, x, z, and Storage)
- All Software Group tests

Just arrived at **COMMON**

UPDATED

System Operator
System Administrator
ILE RPG Programmer
certification tests



See Laura Calley in the
 Certification Lab

Trademarks and Disclaimers

© IBM Corporation 1994-2007. All rights reserved.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

Trademarks of International Business Machines Corporation in the United States, other countries, or both can be found on the World Wide Web at <http://www.ibm.com/legal/copytrade.shtml>.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

The customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Some information addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Prices are suggested U.S. list prices and are subject to change without notice. Starting price may not include a hard drive, operating system or other features. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Photographs shown may be engineering prototypes. Changes may be incorporated in production models.