



Java, J2EE, Web Services and SOA *What's new for i5/OS*

Bruce Wassell
IBM Innovation Centre
Hursley Park



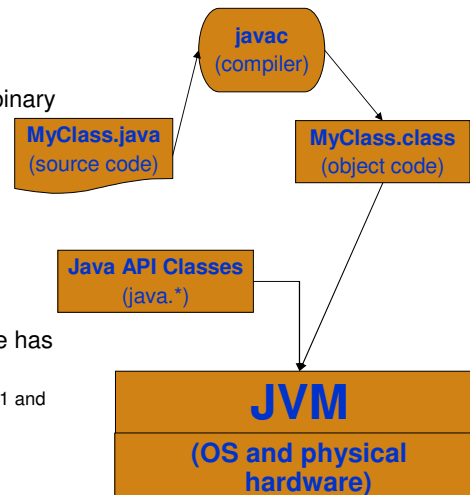
Agenda

1. The latest on DB2 Web Query
2. RPG and COBOL now, where next for i5/OS development?
3. **JAVA, J2EE, Web Services and SOA**
4. Brighten up the front end: V7.0 of HATS, WebFacing and WDHT
5. Getting into the Web: JSF, EGL and PHP for i5/OS
6. What about .NET? Interoperability with i5/OS and DB2

Java

Java and the Java Virtual Machine

- Java compiles into platform-neutral binary format
 - “Byte code” in “class files”
- The Java “API” has seen major improvements over the years
 - New syntax, new classes
- By contrast, the Java Virtual Machine has changed very little
 - Biggest changes occurred between JDK 1.1 and JDK 1.2 releases



The latest JDK for i5/OS – JDK 1.5 ('Java 5')

- Generics
- Enhanced 'for' Loop
- Static Import
- Metadata (Annotations)
- Typesafe Enums
- Autoboxing/Unboxing
- Varargs
- Formatted output (printf)
- Concurrency APIs (java.util.concurrent)
- JVM Monitoring and Management API (java.lang.management)
- JVMTI (JVM Tool Interface)
- Bytecode Instrumentation (java.lang.instrument)

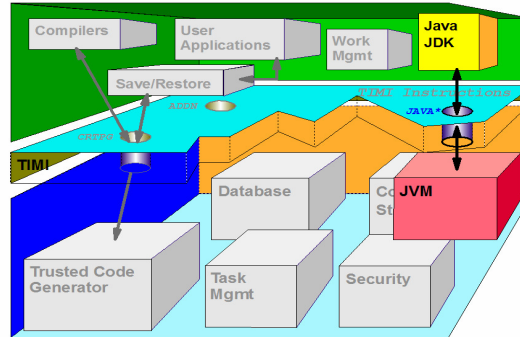
▪ For a complete description of these new features, see:
<http://java.sun.com/j2se/1.5.0/docs/relnotes/features.html>

i5/OS support for J2SE

- **64-bit "Classic" JVM**
 - SDK versions 1.3, 1.4 and 5.0 supported
 - Best where large memory footprints and upward scalability required
 - Some powerful – but i5/OS-unique - tooling
- **32-bit IBM Technology for Java**
 - Implementation of IBM's cross-platform Java ('J9') technology
 - So cross-platform tooling supported
 - SDK 5.0 and i5/OS V5R4 only
 - Smaller memory footprint
- Both include a "native" JDBC driver for local DB2 for i5/OS access

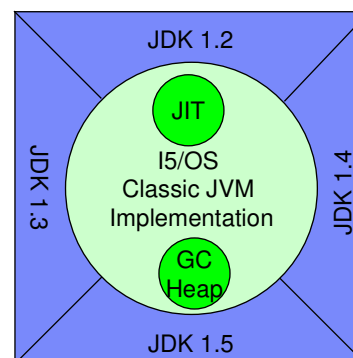
The i5/OS “Classic” JVM Implementation

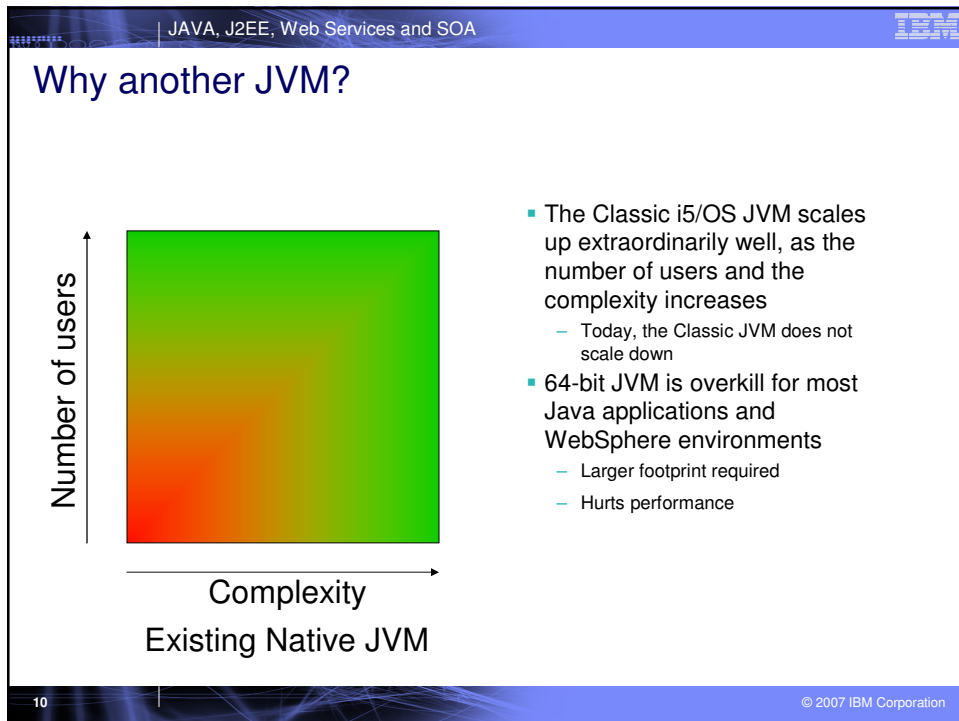
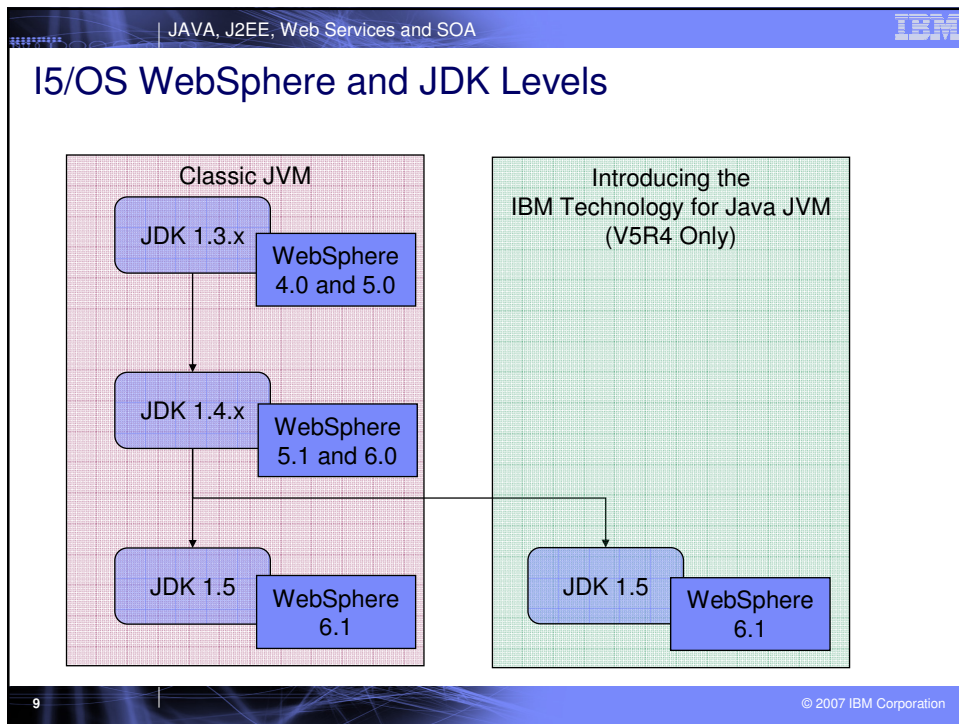
- First Java release on AS/400: V4R2
- Below the MI, each of the MI instructions is “implemented” in the machine
 - To add Java to the mix, a set of special purpose Java-related instructions were developed
 - The JVM does its work primarily below the MI
- Classic JVM is a 64-bit implementation of the JVM.
 - Written by IBM Rochester specifically for the System i.

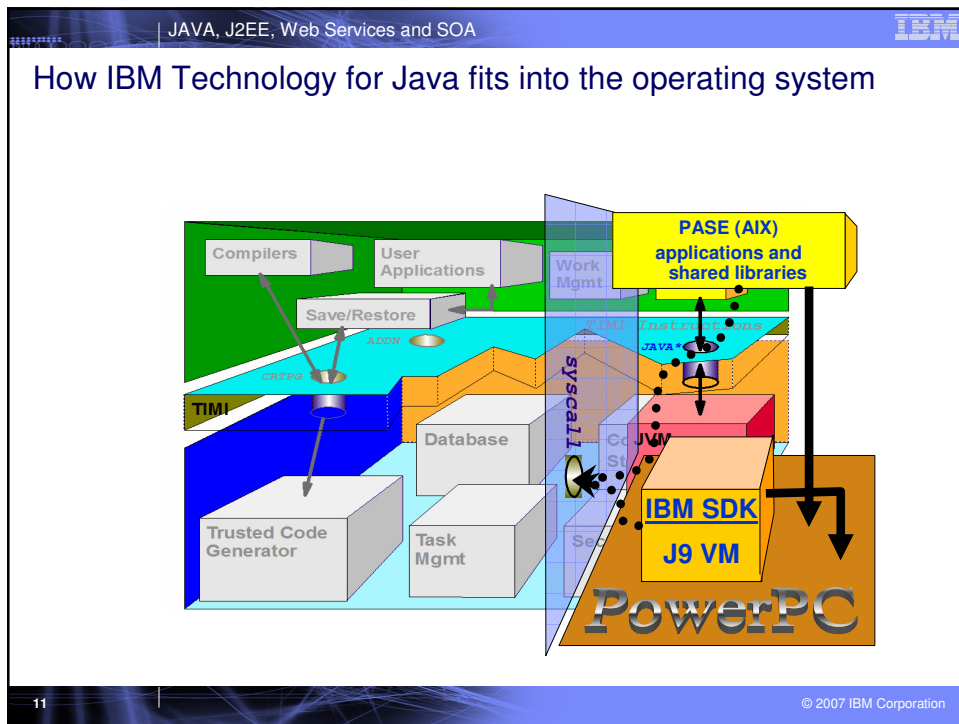


The Classic JVM

- When a new JDK is released, i5/OS creates an “interface” to the Classic JVM
 - Advantages to creating an interface
 - Fast to market
 - Modifications to the JVM are usually apparent in all JDKs
 - More stability early on the cycle
 - Disadvantages to creating an interface
 - Not a lot of innovation for each release of a JDK
 - New functions are shoehorned into existing JVM implementation.
 - i5/OS Classic JVM cannot take advantage of R&D of IBM distributed JVM.







JAVA, J2EE, Web Services and SOA

The i5/OS “IBM Technology for Java” (or “J9”) JVM - Overview

- Runs in a 32-bit address space
 - Object references require only 4 bytes of memory
 - rather than 8 bytes as required in the 64-bit Classic VM
 - Java applications running in the new VM may require significantly less memory
 - the maximum heap size is less than 3 GB.
- Same technology as IBM's JVMs for other platforms
 - Available JVM tuning parameters are now essentially the same as on other platforms.
- Runs in i5/OS PASE
 - Portable Application Solutions Environment

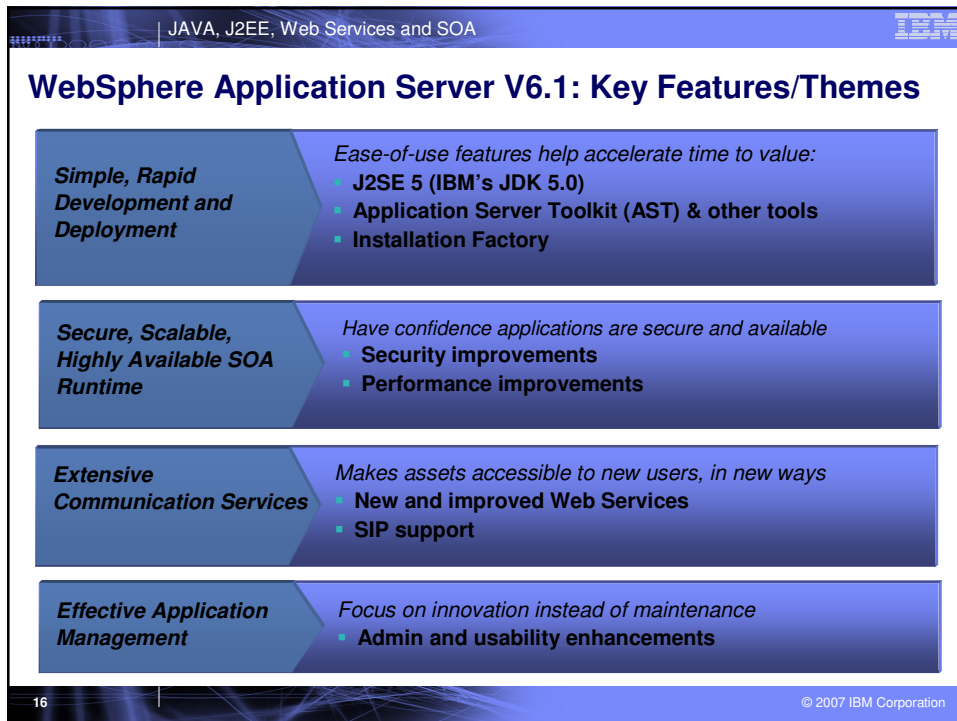
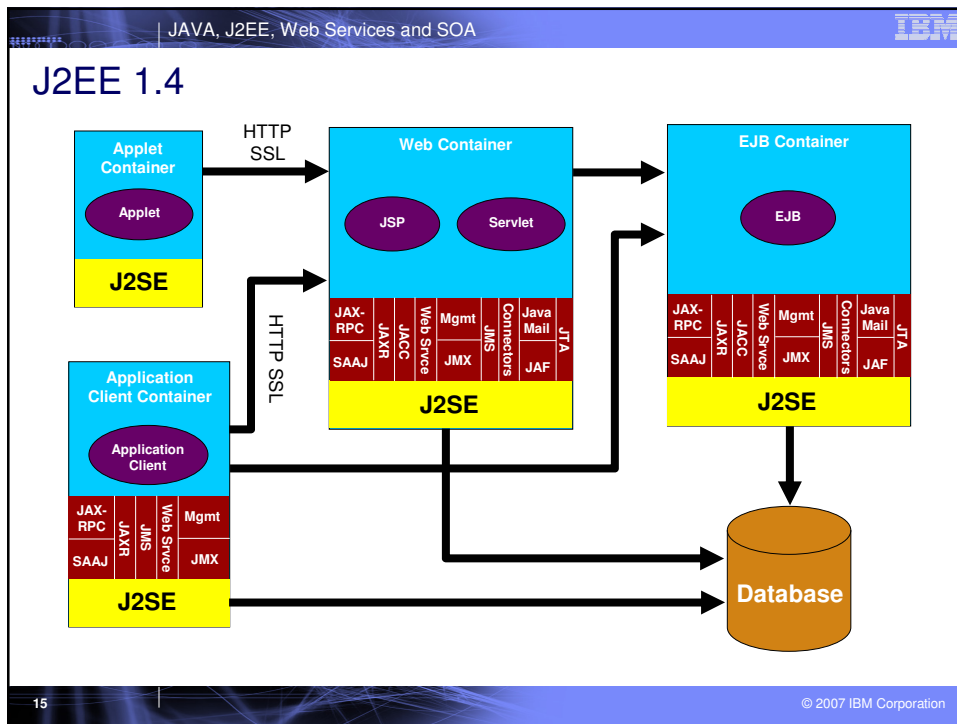
12

© 2007 IBM Corporation

Java on i5/OS Summary

- Within IBM, we pride ourselves on creating technology
 - Some technologies prove themselves, and some do not
 - We tend to converge on “proven” technologies
- Convergence is working for the IBM System line
 - Common CPU technology (PowerPC)
 - Common software technology (Linux, TCP/IP, etc.)
- Converging on a common JVM a logical step
 - IBM Technology for Java is the future JVM for all IBM platforms
- Not all applications can take advantage of the 32-bit JVM.
 - It is important to determine if you will be able to take advantage of the IBM Technology for Java

J2EE + WebSphere Application Server V6.1 Enhancements for i5/OS

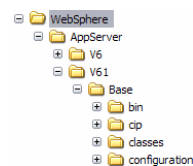


WebSphere Application Server 6.1 for i5/OS

- Base, Developers and ND editions
 - Available through Passport Advantage (Base, Developers and ND only)
- WAS-Express for i5/OS
 - Available free with V5R3M0 and higher
 - Product 5722WE2 (Web Enablement)
 - Delivers the physical media for the customer to install
 - Not available through Passport Advantage

What's New with Product Installation

- Multiple installations of WAS V6.1 are supported
 - "querywasinstalls" to query installed products
- Libraries:
 - QWAS61
 - Contains product definition, loads, subsystem description, job description, job queue, etc.
 - QWAS61x where x = A,B,C,D,...
 - Installer detects next available install specific library available and assigns it
 - Contains the *PGM/*SRVPGM object required for security, process start/stop/ping, server runtime, etc.
- Default installation locations in IFS
 - /QIBM/ProdData/WebSphere/AppServer/V61/<version>
 - /QIBM/UserData/WebSphere/AppServer/V61/<version>



Trial	No feedback (%)	Feedback (%)
1	85	85
2	82	88
3	80	90
4	78	92
5	76	94
6	75	95
7	75	95
8	75	95
9	75	95
10	75	95

- ## Create Application Server

Your system has more than one version of WebSphere Application Server installed.

V6.1 Base WebSphere Application Server V6.1 Base, allows you to add a wide range of complex business logic to your Web application with a full function J2EETM compliant application server.

- ☐ **V5.1 Base** WebSphere Application Server Version V6.1 Base can be installed in multiple locations on this system. Multiple application servers can be created from each installation. The product install path for this installation is `/QIBM/ProdData/WebSphere/AppServer/V61/Base`.

- Potential performance improvements over previous versions

Web Services and SOA

Terminology

... a service?

A **repeatable business task** – e.g., check customer credit; open new account

... service orientation?

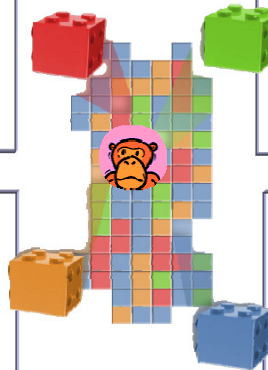
A way of integrating your **business as linked services** and the outcomes that they bring

... Service-Oriented Architecture (SOA)?

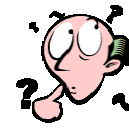
An IT **architectural style** that supports service orientation

... a composite application?

A set of **related & integrated** services that support a business process built on an SOA



Five Stages of Dealing with Web Services

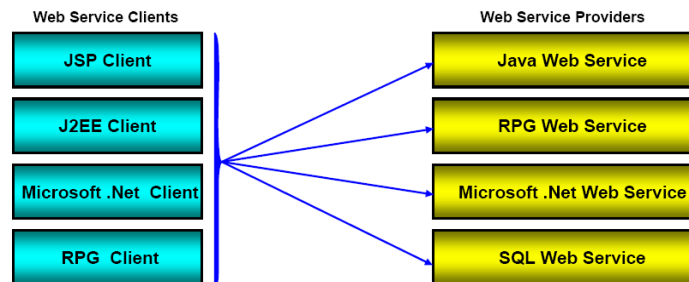


1. **Denial** - It's Simple Object Access Protocol, right?
2. **Over Involvement** - OK, I'll read the SOAP, WSDL, WS-I BP, JAX-RPC, SAAJ, JAX-P,... specs. next, I'll check the Wiki and finally follow an example showing service and client sides.
3. **Anger** - I can't believe those #\$\$%&*@s made it so difficult!
4. **Guilt** - Everyone is using Web Services, it must be me, I must be missing something.
5. **Acceptance** - It is what it is, Web Services aren't simple or easy.

Dave Podnar

Web Services Overview

- Web services are:
 - Applications that are published, found and invoked over the Web
 - Platform and language independent
 - An implementation of Service Oriented Architecture (SOA)
 - Interfaces to business logic, without a user interface
 - Self-contained
 - Self-describing
- Web services are a key technology for implementing SOA



System i developers roadmap/atlas and SOA

The diagram illustrates the evolution of System i architecture through four stages, connected by arrows indicating progression:

- Traditional:** A single box labeled "5250 DDS RPG/COBOL File IO" sits above a box labeled "PDM, SEU".
- Improved Developer Productivity:** A box labeled "5250 DDS RPG/COBOL File IO" sits above a stack of two boxes: "Remote System Explorer" and "WDS*c*".
- Modular Architecture:** A stack of four boxes: "5250 DDS" (green), three "ILE" boxes (blue), "DB2 and SQL" (blue), "IBM Toolbox for Java" (blue), "Remote System Explorer" (blue), and "WDS*c*" (blue).
- Application Integration:** A stack of four boxes: "WebServices" (blue), three "ILE" boxes (blue), "DB2 and SQL" (blue), and a stack of three boxes: "IBM Toolbox for Java" (blue), "Remote System Explorer" (blue), and "WDS*c*" (blue).

- Evolve existing host applications to a Model-View-Controller architecture
 - Separate business logic, data access and navigation
 - Event driven, stateless interfaces
 - Modularized business logic, ILE
- Evolve data access
 - SQL, stored procedures
 - Triggers, referential integrity, constraints

© 2007 IBM Corporation

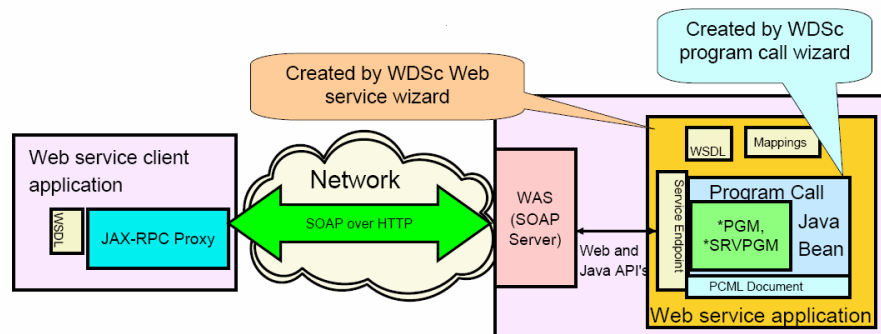
- Evolve existing host applications to a Model-View-Controller architecture
 - Separate business logic, data access and navigation
 - Event driven, stateless interfaces
 - Modularized business logic, ILE
- Evolve data access
 - SQL, stored procedures
 - Triggers, referential integrity, constraints

Web service enabling i5/OS program resources

- WDS Sc Web Service wizard
 - Builds JavaBean interfaces to i5/OS program resources from program source or PCML
- Builds the Web Service artifacts

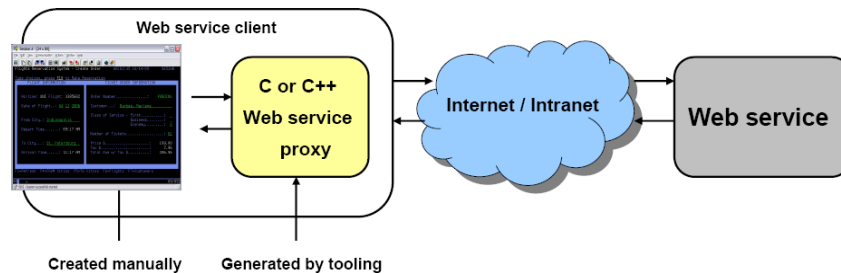
The diagram illustrates the architecture for enabling i5/OS program resources as web services. On the left, a 'Web service client application' (purple box) contains a 'WSDL' icon and a 'JAX-RPC Proxy' (blue box). In the center, a yellow cloud labeled 'Network' contains a large green double-headed arrow labeled 'SOAP over HTTP'. To the right of the network is a 'WAS (SOAP Server)' (pink box). Further right is a 'Web service application' (yellow box). Inside this application, there is a 'Service Endpoint' (green box) and a 'Program Call' (blue box). The 'Program Call' contains a green box with '*PGM, *SRVPGM' and a blue box with 'Java Bean'. Below the 'Program Call' is a 'PCML Document' (white box). Above the 'Program Call' are 'WSDL' and 'Mappings' (white boxes). A callout bubble from the 'WSDL' and 'Mappings' boxes points to the 'WDS Sc Web Service wizard' (orange box) and says 'Created by WDS Sc program call wizard'. Another callout bubble from the 'Program Call' box points to the 'WDS Sc program call wizard' and says 'Created by WDS Sc program call wizard'. A double-headed arrow connects the 'WAS (SOAP Server)' and the 'Service Endpoint'.

- **WDS Sc Web Service wizard**
 - Builds JavaBean interfaces to i5/OS program resources from program source or PCML
- **Builds the Web Service artifacts**



Web services clients: System i applications

- I5/OS supports Apache Axis Version 1.5 C++ based Web service client
 - Included with i5/OS V5R4
 - Included with XML Toolkit for i5/OS (5733-XT1 and 5733-XT2)
- RPG, COBOL or C Web service client
 1. Generate Web service proxy based on a WSDL file
 2. Add code to call Web service proxy to an RPG, COBOL or C program



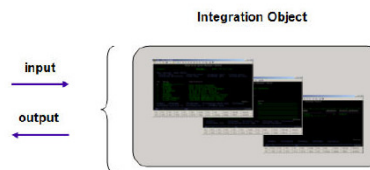
27

© 2007 IBM Corporation

HATS Web Service Creation

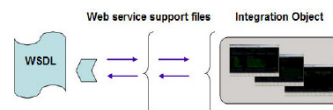
■ Create an Integration Object

- Defines the programming interface to the macro

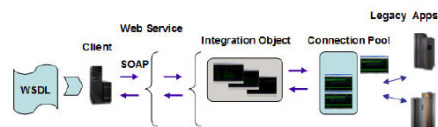


■ Create Web Service

- Web Service files
- Web Service Definition Language (WSDL) file



- Create Web Service client for test
- From WSDL file

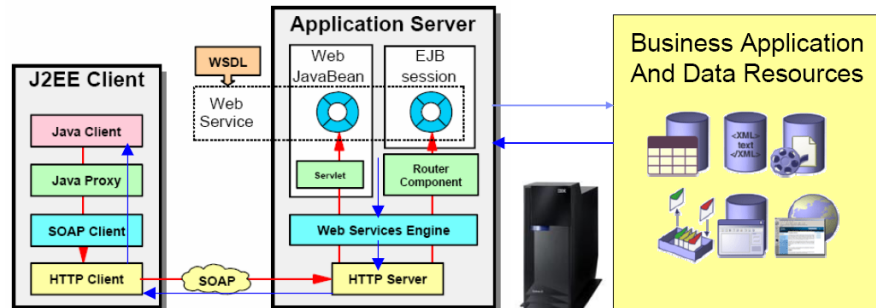


28

© 2007 IBM Corporation

Web services runtime environment

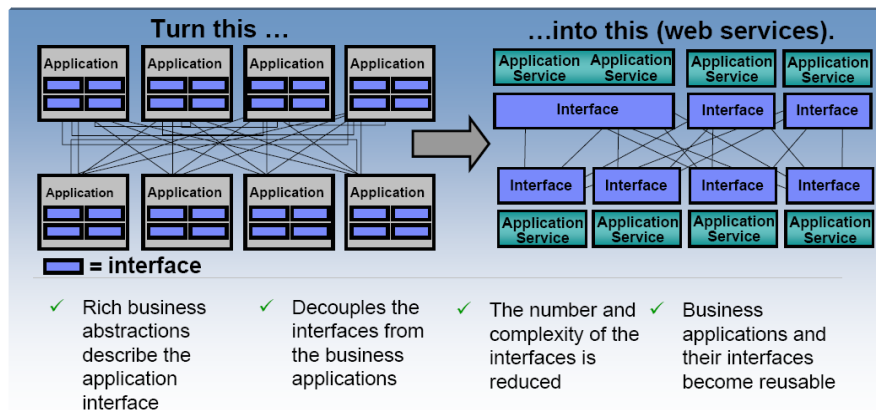
- Open standards, platform & programming language independence
- HTTP and application server foundation
- Web services runtime engine
- Connectors to enterprise resources
- Business process orchestration



29

© 2007 IBM Corporation

Web Services are a good start...

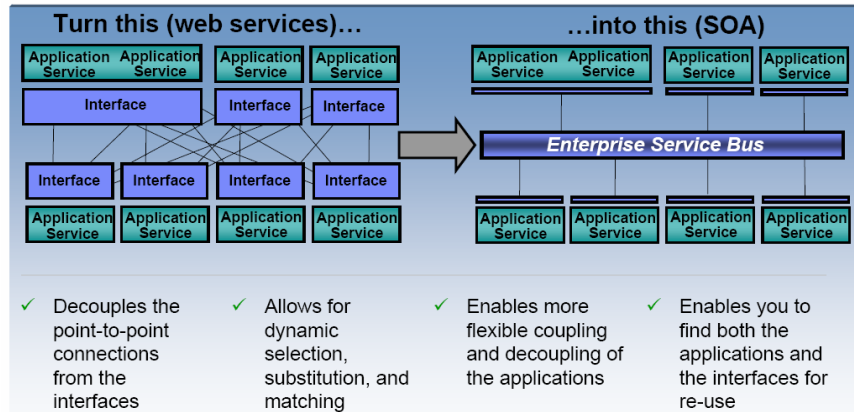


But separate connection points still leaving bloated interfaces

30

© 2007 IBM Corporation

The Enterprise Service Bus shrinks those interfaces further



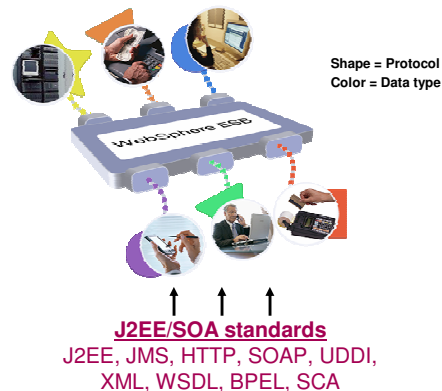
RESULT → Greater Business Responsiveness

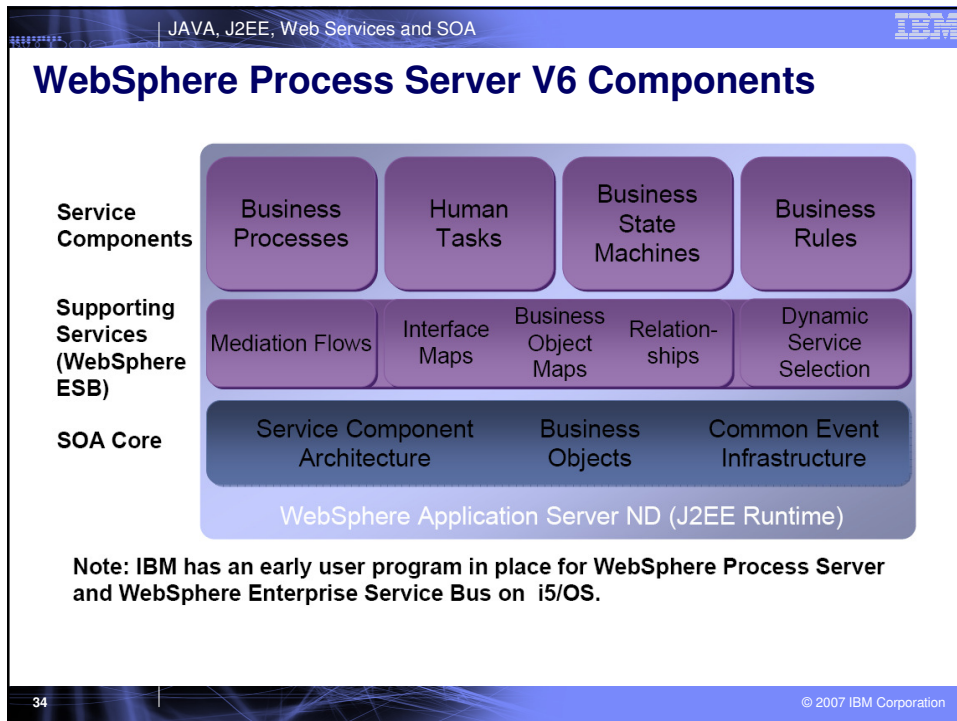
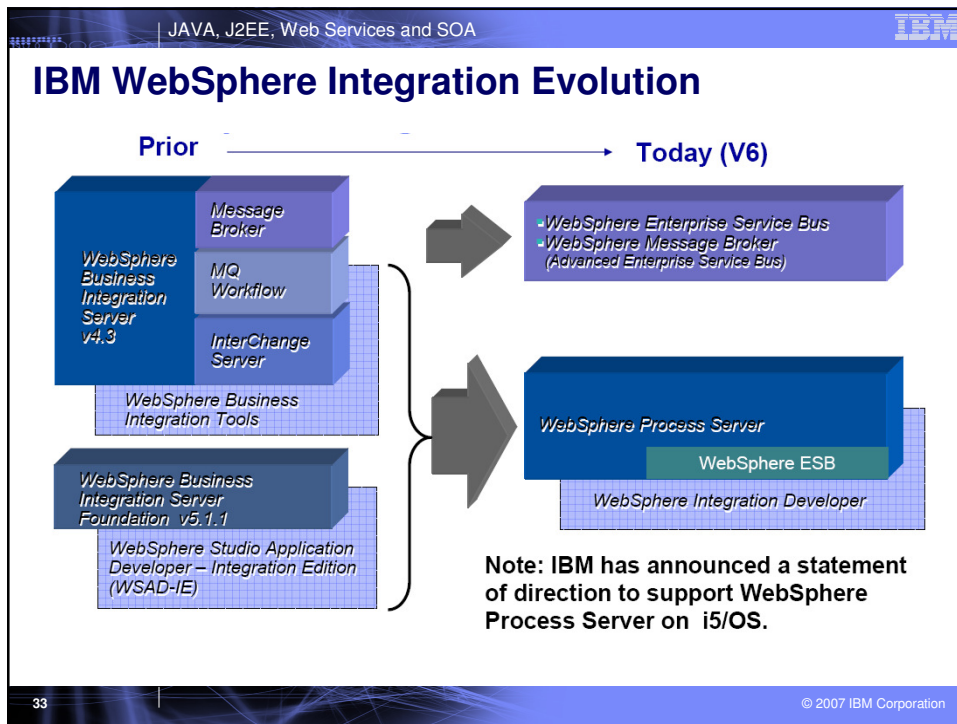
Enterprise Service Bus

WebSphere Enterprise Service Bus is an integral part of WebSphere Process Server and provides a connectivity infrastructure for integrating applications and services.

An ESB performs the following between requestor and service

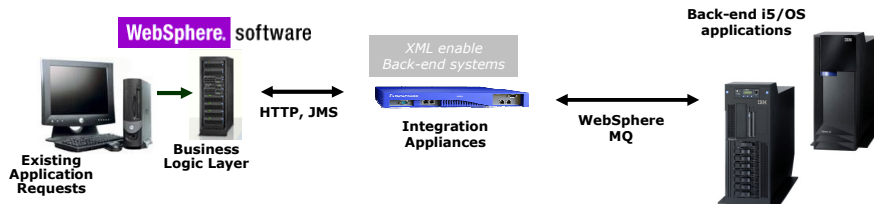
- **ROUTING** messages between services
- **CONVERTING** transport protocols between requestor and service
- **TRANSFORMING** message formats between requestor and service
- **HANDLING** business events from disparate sources





Create an ESB with an Integration Appliance

SOA enable back-end systems without deep Web Services technical skills.



- IBM Datapower XI50 can deliver traditional RPG/COBOL apps as web services leveraging WebSphere MQ
- RPG/COBOL programmers can easily code to the WMQ API's

Redbook

- SOA: Understanding the big picture
 - SOA overview
 - SOA application design
 - Web services technology stack
 - Sample scenario
- Implementing the service provider
 - Program Call (RPG, Cobol) Web service
 - DB2 UDB Web service
 - HATS Web service
 - PHP Web service
- Implementing the Service Consumer
 - IBM Web Services Client for ILE (RPG, C, C++, COBOL)
 - JSF Web service client
 - PHP Web service client



<http://www.redbooks.ibm.com/abstracts/sg247284.html>

Resources

- Redbooks, available at <http://www.redbooks.ibm.com>
 - Building SOA solutions for System i, SG24-7284
 - WebSphere Version 6 Web Services Handbook Development and Deployment, SG24-6461-00
 - Rational Application Developer V6 Programming Guide , SG24-6449-00
- WebSphere Application Server for System i Information Center -
 - <http://publib.boulder.ibm.com/infocenter/wsdoc400/index.jsp>
- IBM developerWorks
 - <http://www.ibm.com/developerworks/webservices>
 - <http://www-128.ibm.com/developerworks/db2/zones/webservices/>
 - <http://www.ibm.com/developerworks/rational/products/egl/>
 - <http://www-128.ibm.com/developerworks/lotus/library/nd7-webservices/>
 - <http://www.ibm.com/developerworks/opensource/top-projects/php.html>

Trademarks and Disclaimers

© IBM Corporation 1994-2006. 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.

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

AS/400	e-business on demand	i5/OS
AS/400e	IBM	OS/400
eServer	IBM (logo)	System i5
@server	iSeries	

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, other countries, or both.
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 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.
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.

All 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. Contact your local IBM office or IBM authorized reseller for the full text of the specific Statement of Direction.

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.