Java Overview and z/OS Update

Theresa Tai

March 3, 2005 (Session: 8365)

zSeries New Technology Center, Poughkeepsie, New York

ttai@us.ibm.com

Trademarks

IBM, AIX, CICS, DB2, IMS, z/OS, OS/390, S/390, System/390, VisualAge, WebSphere Application Server, WebSphere Studio, z/VM - are trademarks or registered trademarks of the IBM Corporation

Sun, Sun Microsystems, JavaSoft, Java, JavaBeans, JDK, Java 2 Micro Edition, J2ME, Java 2 Standard Edition, J2SE, Java 2 Enterprise Edition, J2EE - are trademarks or registered trademarks of Sun Microsystems Inc.

Click on view and follow link to header & footer to enter Copyright and Author information

Content

Overview: Java on z/OS

IBM Investment in Java

- Market leader in delivering Java technology
- z/OS platform for mission critical workload

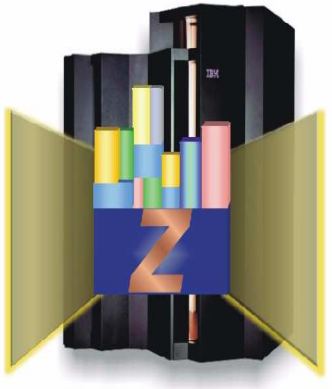
2005 focus for Java on z/OS

- IBM SDK for z/OS, Java 2 Technology Edition V1.4
- Supporting key IBM middleware & Operating Systems
- Introducing zAAP Engines for Java Workloads

Wrap-up and final Q&As

Click on view and follow link to header & footer to enter Copyright and Author information

SDK for z/OS and OS/390



- IBM Developer Kit for OS/390 , Java 2 Technology Edition
  - Currently at SDK1.3.1 Level
  - Product 5655-D35
  - Supported on OS/390 R8 and above, z/OS V1.1 and above
- IBM SDK for z/OS, Java 2 Technology Edition, Version 1.4
  - Currently at SDK1.4.2 level
  - Product 5655-I56
  - Supported on z/OS V1.2 and above
- IBM 64-bit SDK for z/OS, Java 2 Technology Edition, Version 1.4
  - Web available on September 24, 2004 at SDK1.4.2 level
  - Product 5655-M30
  - Supported on z/OS V1.6 and above
  - First 64-bit Java SDK on z/OS
- All products are delivered via the z/OS Java website in non-SMP/E format and via ShopIBM in SMP/E format
- No Charge Product but it is supported by the normal IBM support channels

NOTE: The EOC date for SDK 1.3.1 is March 2007

SDK for Linux on zSeries

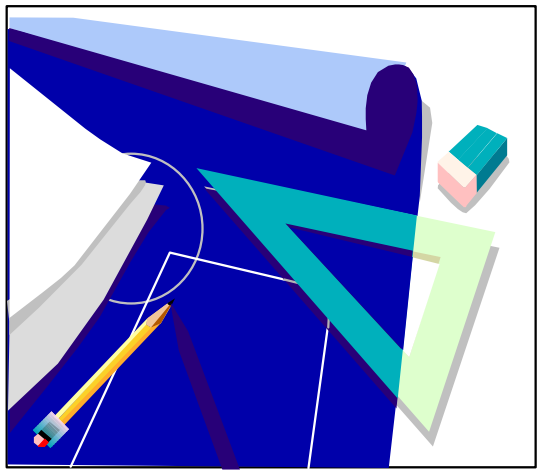
- IBM Developer Kit for Linux® , Java 2 Technology Edition
  - SDK1.3.1 Level
  - Suse SLES 7, Red Hat 7.2, SuSe SLES 8, Turbo
  - SuSe SLES 7 64-bit (in 31-bit compatibility mode)
- IBM 31-bit SDK for Linux® on zSeries, Java 2 Technology Edition, Version 1.4
  - SDK1.4.2 Level
  - Suse SLES 7, SuSe SLES 8 - 31 bit mode, RHEL3
- IBM 64-bit SDK for Linux® on zSeries, Java 2 Technology Edition, Version 1.4
  - SDK 1.4.2 Level
  - SuSe United LINUX (aka SLES 8), RHEL 3 (NPTL)
  - First 64-bit Java SDK on zSeries
- Delivery and Service
  - On DeveloperWorks at https://www6.software.ibm.com/dl/lxdk/lxdk-p
  - Also available from LINUX distributors
  - Level 1, 2 service by IGS contract

IBM 64-Bit JVM

- IBM has announced the IBM 64-bit SDK for z/OS, Java ™ 2 Technology Edition Version 1.4
  - Announcement 204-180 August 10, 2004
  - GA December 2004
- Product PID Number
  - 5655-M30 (IBM 64-bit SDK for z/OS Java 2 Technology Edition, V1.4)
  - 5655-I48 (IBM 64-bit SDK for z/OS S&S)
- Key Requirements
  - HW: z990, z900, z890, z800
  - SW: z/OS V1.6 or z/OS.e V1.6 or later
- Compatibility
  - Most Java applications ran on prior version of SDK should run on 64-bit SDK
  - Applications using JNI need to ensure native code is XPLink enabled
  - Provides the same functions as 31-bit SDK with the exception of Persistent Reusable JVM technology
- SDK 1.4.2 is at UQ90449 PTF level

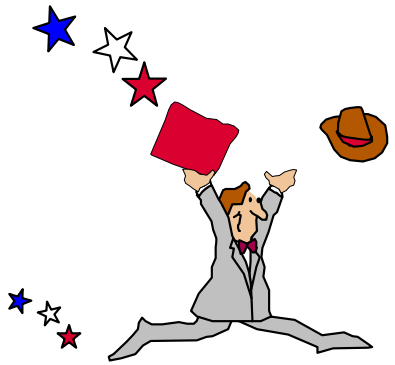
The Re-engineered IBM JVM

- The Re-engineered JVM common code base, plus Platform-specific code for
  - File handling
  - ASCII vs EBCDIC code page
  - JRIO (Java Record I/O)
  - RAS Characteristics
  - Profiling and Security APIs
  - RACF Integration
  - Hardware instruction set



- Integral Part of WebSphere Application Server platform
- Be the market leader in delivering Java technology

IBM Java™ 2 Objective - Value Add



- IBM Owned SDK Asset Base
- Improved Quality of IBM SDKs
  - Better development processes
  - More consistent functional implementation
  - Performance enhancements across platforms
  - IBM Technology Added Value (JVM and Classes)
- Reduced cost of porting to new platforms
- Leverage new technology in both IBM hardware and software
- Continually Improving Tools for Application Development and Deployment
- Improved Performance, Scalability, RAS and improved System Exploitation

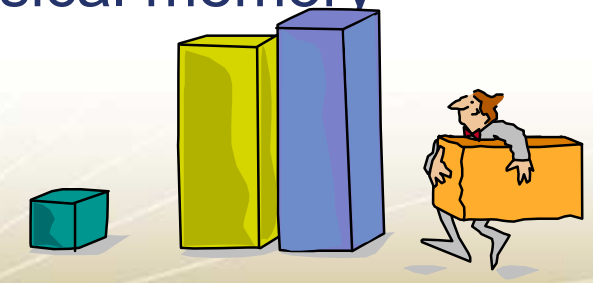
GOAL: Deliver Complete, Fully Compliant, Leadership SDKs

Java Strategy for zSeries Platform

- Establish Java as a "de facto" programming environment on z/OS
  - Deliver J2EE capability in synch with Java industry (J2EE Certified)
- Lead with z/OS Qualities of Service
  - Market leader in delivering Java technology
  - z/OS platform for mission critical workload
- Enable all "Application Execution Environments" to support Java2 based applications:
  - WebSphere Application Server
  - Transaction Servers, ie. CICS & IMS
  - DB2 data base (Stored Procedures)
  - Enable connectivity to middleware
  - Messaging queuing
  - Batch processing
- Continue to provide world class Development and Deployment Tooling

Memory Management

- Java Heap & Garbage Collection
  - Do nothing, use defaults
    - 64MB
    - WebSphere default is 256MB for WAS V4 and V5
- Java heap size and GC frequency
  - Use small heap, frequent, fast GC
  - Use large heap, infrequent, slow GC
- Make sure you have enough physical memory to support the heap size

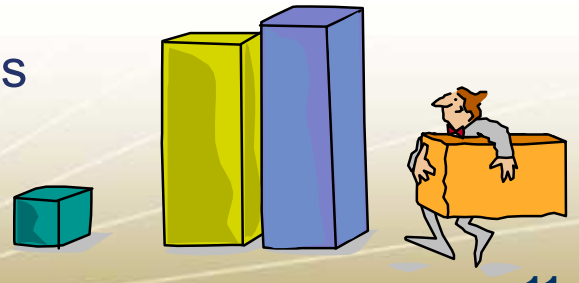


GC Evolution

- From Concurrent GC (SDK 1.3) to Mark & Sweep with optional Compaction (SDK 1.3.1)

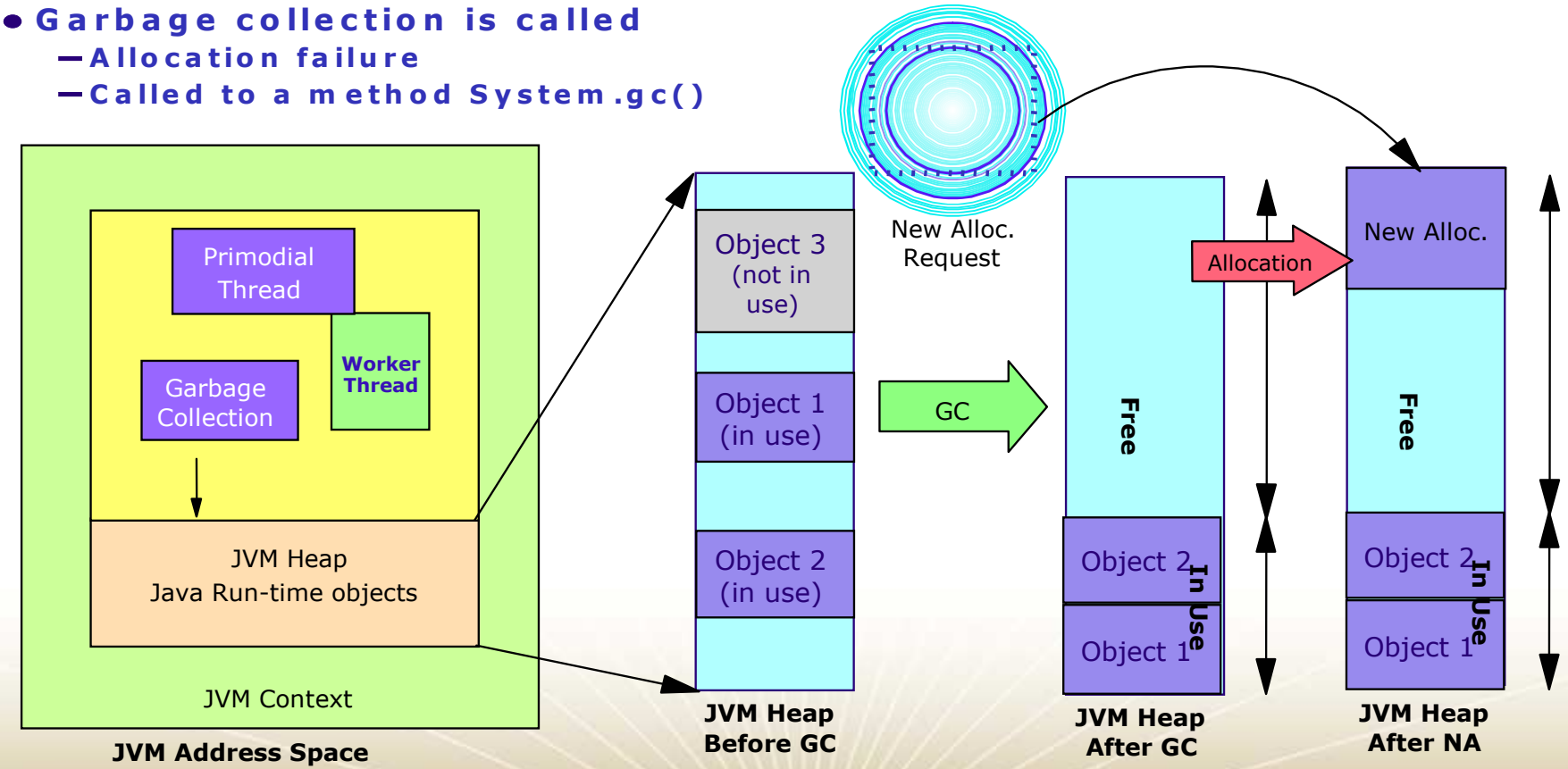
- Mark and Sweep with optional Compaction
  - Marking live objects
    - A live object is an object that is still reachable by program code; that is, it is not garbage
  - Sweeping up the garbage and coalescing it into large areas of free space
  - Compacting the remaining live objects to create a yet larger area of free space

- SDK 1.4.2 Incremental Compaction
  - Compacting a different section of each heap's GC cycle
  - Reduce pause time



What is Garbage Collection (GC) ?

- Garbage collecton is within the JVM that automatically detects objects that are no longer being used and frees them to reclaim the space
- Garbage collection is called
  - Allocation failure
  - Called to a method System.gc()



Java Heap Tuning & Performance Tips

- Collect verboseGC data under peak load
- Verify that your free space in the heap does not 'shrink' from GC to GC
  - Possible memory leak?
  - Look at the %Free data field
- Performance Tips
  - Make sure the JIT is enabled
  - Do not use debug version of Java
  - Make you only point to classes you need
  - Make your CLASSPATH efficient
    - Put frequently referenced classes first
  - Turn off JRAS debugging support
  - Set a reasonable Java heap size
  - Review GC overhead % on a regular basis

Tuning: The ***Right*** Java Heapsize

- Collecting -verbose:gc data (output below)
- The desirable GC overhead % is about 5%
  Reference Chapter 3 Tuning Performance Parameter Index of WAS V5.1 Information Center
  Document Title: Performance Monitoring and Tuning

$$\text{GC Overhead\%} = \frac{\text{elapsed time doing GC}}{\text{elapsed time from GC to GC}} * 100$$

- Example: 118/32225 * 100 = 0.366%

```
 . . .
<AF[21]: Allocation Failure. need 32784 bytes, 32225 ms since last AF>
<AF[21]: managing allocation failure, action=1 (84320/131004928) (3145728/3145728)>
<GC(21): GC cycle started Wed Aug 05 22:46:11 2003
<GC(21): freed 99587928 bytes, 76% free (102817976/134150656), in 118 ms>
 <GC(21): mark: 103 ms, sweep: 15 ms, compact: 0 ms>
 <GC(21): refs: soft 0 (age >= 32), weak 0, final 878, phantom 0>
<AF[21]: completed in 118 ms> . . . . . .
 . . .
```

The Persistent Reusable JVM (aka Resettable JVM)

- The Hi-performance transactional JVM
- Significant savings on the start up cost
- Allow single JVM to be used for multiple transactions
- The high throughput, short & repetitive transactions
- Single transaction per JVM at any time
- Optimized Garbage Collection - Split Heaps

- The Exploiters
  - CICS, IMS and DB2 (Stored Procedures)
- Shared Classes
  - Reduced memory footprint
  - Increased performance
- **Obsolete in SDK 1.4.2 with 64-bit JVM**

Java Record I/O

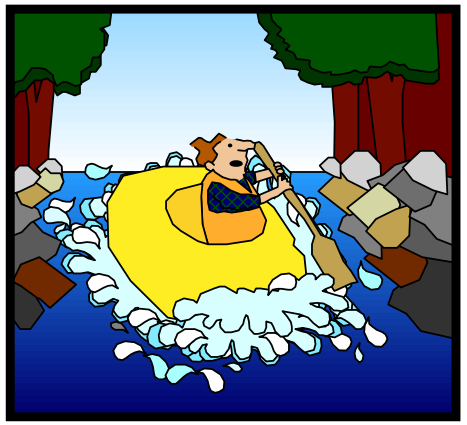
- JRIO is a class library, similar to java.io
- JRIO provides record-oriented access on z/OS
  - Virtual Sequential Access Method (VSAM) data sets (KSDS only)
  - Non-VSAM record-oriented data sets
  - The System Catalog
  - Partitioned data set (PDS) directory
  - DDName and GDG support
- How do I .... and sample programs
  www-1.ibm.com/servers/eserver/zseries/software/java/jrio/jriofaq.html#usage

Java Debugger (JDB)

- JDB uses the Java Platform Debugger Architecture (JPDA)
- Included in the SDK for z/OS
- Invoked by the JDB command
  - Attach the debugger to the JVM
    - Jdb –attach <port number>
- Start JVM with
  - Java –Xdebug –Xrunjdwp:transport=dt_socket,server=y, address=<port number> MyApplication <myapplication>
- JDB options
  - Jdb -help

IBM RAS for Java

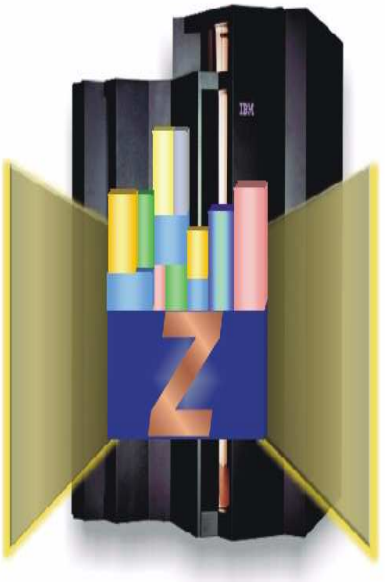
- JVM initiated System Dumps
  - Environment variable JAVA_DUMP_OPTS available
- Improved tracing and problem determination
  - JVM Exception handler improvements
  - Method Level Trace
  - Application Trace
    - Call Java trace from Java Programs
  - JVM Monitoring Interface JVMMI
    - Lightweight version of JVMPI
  - Robust termination
  - GTF trace for a JITed Java method
  - Hook to interface with MVS dynamic slip trap to collect dumps/traces
  - Rewritten Dump Formatter (IPCS JVMDATA)



- **IBM SDK 1.3.1 and SDK 1.4 Diagnostics Guide**

Key Prerequisites For SDK 1.4

- Hardware:
  - z800, z900, z890, z990 or comparable server
  - S/390 Parallel Enterprise Servers
  - G5 and G6 or comparable server
- Software:
  - IBM SDK for z/OS, Java 2 Technology Edition, V1.4.x
    - z/OS V1 R2 or above
    - z/OS.e V1 R3 or above
  - IBM 64-Bit SDK for z/OS, Java 2 Technology Edition, V1.4.2
    - z/OS V1 R6 or above
    - z/OS.e V1 R6 or above



SDK 1.4 Highlights

- The primary focus of J2EE 1.4 is support for Web Services
  - JAX-RPC (Java API for XML-Remote Procedure Call)
  - SAAJ APIs (SOAP with Attachments API for Java)
  - EJB specifications are extended to support WS using stateless session beans
  - JAXR APIs supports access to registries and repositories
- The Connector API now supports integration with asynchronous messages systems, plug in JMS providers
- XPLINK - Use z/OS XPLINK linkage mechanisms
- The IBMJCE4758 provider
- Compliant with Sun Java SDK 1.4 APIs
  - http://java.sun.com/j2se/1.4/docs/
- We are certified – Sun's CTS (visit Sun's website)
- JSR 0059 defines content of SDK 1.4
  http://www.jcp.org/aboutJava/communityprocess/first/jsr059/index.html

Migration Considerations

- Coexistence and Interoperability
  - SDK 1.4.x shall coexist with **PREVIOUS** release of the IBM SDK installed on the same platform
- Back up all the configuration files and security policy files before upgrade
- There is no guarantee that SDK 1.4.2 compiled classes work on pre 1.4.0 SDK releases
- The 64-bit z/OS SDK installation is designed to install to a different directory tree from any currently installed 31-bit SDK (they co-exist)
  - /usr/lpp/java/J1.4_64/bin (default)

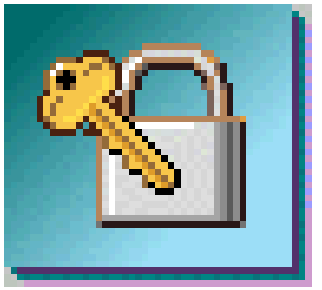
J2EE 1.4 at a Glance

- **Web Services and XML support**
  - **Standards / Portability** - XML Schema definitions for all deployment descriptors
  - **JAX-P 1.2** - New properties for XML parsers
  - **JAX-R** - XML registry API
  - **JAX-RPC** - APIs for representing WSDL-based services as RPCs in Java (and vice-versa)
  - **JSR 109** - Web services programming and deployment model
  - **SAAJ 1.1** - SOAP Attachments API for Java
- **Messaging**
  - **EJB 2.1**
    - Timer service Web service end-point support
    - Typed message beans (used for any inbound JCA including pluggable JMS provider)
  - **JMS 1.1**
    - Unification of point-to-point and pub-sub interfaces
- **ISV Enablement**
  - **JMX 1.2 / JSR-077 (J2EE Management)**
    - Notification emitters, and standard patterns
    - Information model representing J2EE application server concepts
  - **JSR-088 (J2EE Deployment)**
    - XML-based deployment interfaces for J2EE
  - **JACC 1.0**
    - Java Authorization Contract with Containers
    - APIs for registering J2EE component authorization policies
- **Other**
  - **Servlet 2.4**
    - Extensible deployment descriptors
    - Request/response listeners
  - **JSP 2.0**
    - Expression Language
    - Simple Tag Extension
  - **EJB 2.1**
    - Timer Service
  - **JDBC 3.0**
    - Meta data and cursor support
  - **JavaMail 1.3** updates
  - **J2CA 1.5**
    - In-bound connections
    - RA lifecycle support
    - Work manager (threads for resource adapters)

zAAP: New Java Workload Engine on z/OS

- zAAP (zSeries Application Assist Processor)
- New processor type on z890 and z990 hardware supporting z/OS
- Order using Feature Code 6520
  - You can order up to one zAAP per configured or unassigned standard CP on the processor
- A specialized z/OS and z/OS.e Java execution environment for Java-based applications
  - WAS V5.1
  - CICS/TS V2.3
  - DB2 V7 and V8
  - IMS V8
  - WebSphere WBI for z/OS
- Newly instrumented IBM JVM
  - With no anticipated modifications to Java application
- Require z/OS 1.6 and SDK 1.4 with PTF UQ88783

What is Java Security?

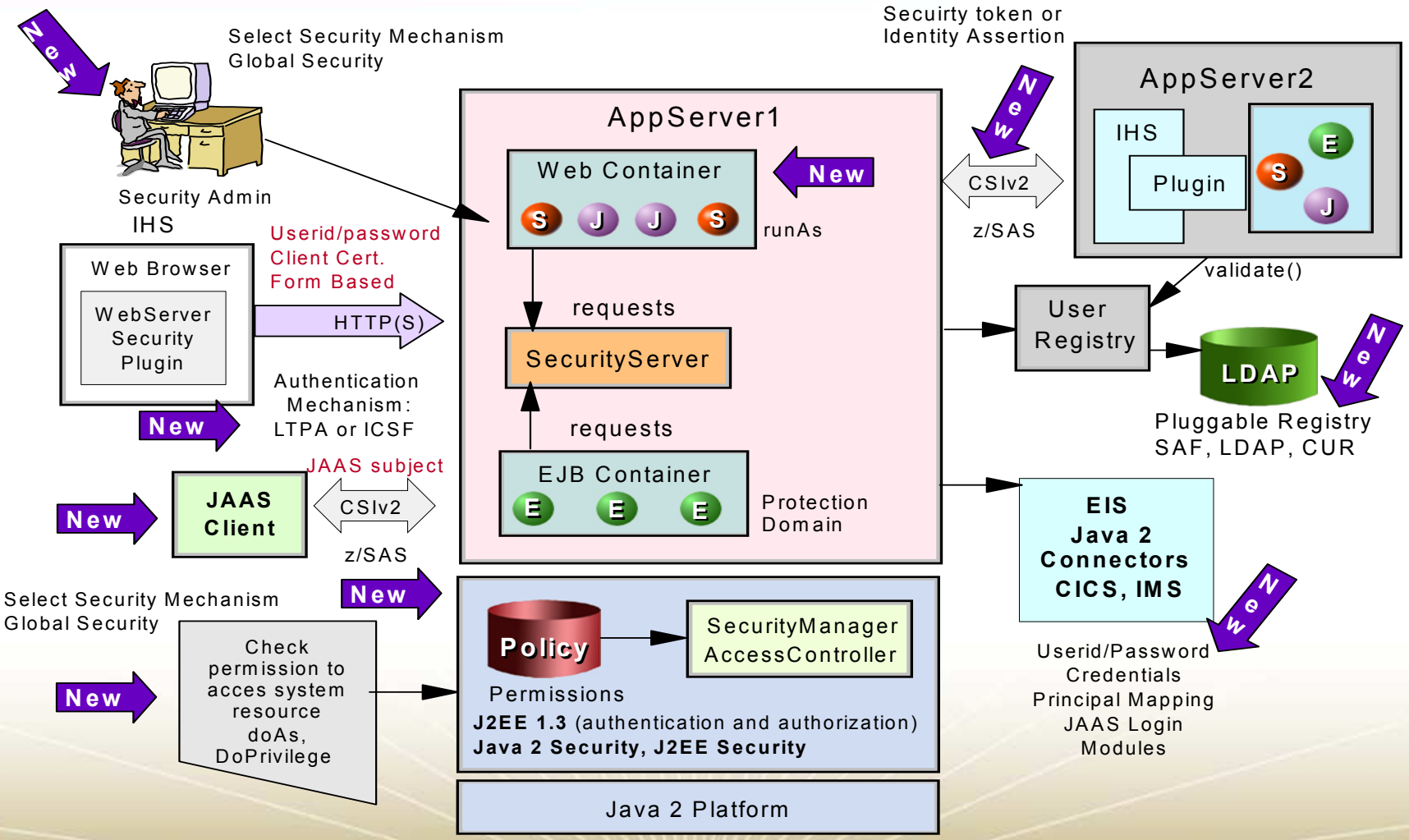


- Part of Java 2 framework
  - A set of common cross platform programming API's
- Java security extensions
  - A set of common API's to extend Java 2 to add security capabilities
- Allow Java applications easy access to complex security capabilities within Java framework
- Java security extensions added to base Java 2 (J2SE) framework with SDK 1.4

Java Security Components

Components	Function/Features in **Bold**
JAAS	**Java Authentication** and **Authorization Service**
JCE	**Java Cryptographic Extension** using CCA (Common Cryptographic Architecture)
JCE4758	**IBM Hardware Cryptographic Device for zSeries**
JSSE	**Java Secure Socket Extension** support SSL and TLS
SAF Interface	**z/OS Security Services in Java**, implement a SAF interface, JNI to call SAF/RACF

J2EE and Java 2 Security



Click on view and follow link to header & footer to enter Copyright and Author information

Java Security Feature Comparison

Security Feature	Purpose	Details
Java 2 Security	Code based access control	Enforce access control based on the location of the code and who signed it; not based on the principal. Java 2 security is defined in policy files and enforced at runtime.
J2EE Security	Identification, Authorization and Identity propagation.	Role-based security for JSPs, Servlets and EJBs are supported. The identity can be changed based on method delegation attributes. J2EE security is defined in configuration settings or within application code and enforced by runtime or programmatically.
JAAS Security	Authentication and authorization	Enforce access control based on the current Principal/Subject. JAAS security is defined in application code and enforced programmatically.

IBM Rational Application Developer (RAD)

- RAD is the new development tool for applications
  - Rational Software Development Platform (Eclipse 3.0)
  - RAD - Rational Application Developer (V6.0)
  - RWD – Rational Web Developer (V6.0)
  - A follow-on products for
    - WSAD (WebSphere Studio Application Developer)
    - WSSD (WebSphere Studio Site Developer)
- Maintain standards and middleware currency
  - Full support for building J2EE 1.4 applications
    - SDO, JSF, Portal, Modeling, Deploying, Profiling and Testing
  - Support deployment to WAS V5, V5.1 and V6
  - Supports J2EE 1.2 and 1.3 applications migrated to 1.4
- Extends on the features of WSAD

Profiling Tools

- RAD Profiler
  - Memory analysis features to locate memory leaks in heap dumps
  - New ProbeKit
    - Easy profiling at key points in the application using byte-code instrumentation
    - Don't have to recompiling the classes to be profiled
    - Summary report on % on amount of code is executed
- RAD Application Analysis
  - Applications can be reviewed for coding practices
    - For GC, performance, scalability, portability, serialization and thread usage
  - Allow developers to create their own rules based templates
- WebSphere PMI (Performance Monitoring Infrastructure)
  - JMX MBeans, byte-code implementation
  - It collects statistics on activities in the server
  - Requires the TPV (Tivoli Performance Viewer)

XML Support in Java SDK 1.4.1

- SDK 1.4.2
  - XML4J 4.3 (comparable to Xerces-J 2.6)
    - Finalized XML Schema API support
    - Other standards support: DOM Level 3 "Core" and "Load and Save", XML 1.1, ...
    - Performance Improvements
  - XSLT4J 2.6 (comparable to XalanJ 2.6)
    - Configurable high performance underlying parse

http://www.ibm.com/zseries/software/xml
http://www-.ibm.com/servers/eserver/zseries/software/xml/perform/

Summary

- Continue to provide z/OS, OS/390 SDK technology base for
  - WebSphere
  - Linux
  - Middleware
  - ISVs
- Continued rollout of Java2 including new IBM architectures to allow better platform integration, function, tailoring and performance
- Recommendation: Stay Current by visiting our web site Frequently
  http://www.ibm.com/servers/eserver/zseries/software/java
- Reporting a problem
  http://www-1.ibm.com/servers/eserver/zseries/software/java/services.html

Reference Summary

- SDK 1.4 pre-reqs
  http://www-1.ibm.com/servers/eserver/zseries/software/java/prereqs14.html
- SDK 1.4 Restrictions and Considerations
  http://www-1.ibm.com/servers/eserver/zseries/software/java/restrict14.html
- SDK 1.4.2 APIs
  http://java.sun.com/j2se/1.4.2/docs/api/
- JDB and JPDA
  - http://java.sun.com/products/jpda/
  - http://java.sun.com/j2se/1.4.2/docs/guide/jpda
- SDK 1.3 and 1.4 compatibility
  http://java.sun.com/j2se/1.4/compatibility.html
- SDK Diagnostic Guide
  http://www-106.ibm.com/developerworks/java/jdk/diagnosis/
- WebSphere Studio Application Developer (WSAD)
  http://www-306.ibm.com/software/awdtools/studioappdev/
- Rational Application Developer (RAD)
  http://www-306.ibm.com/software/awdtools/developer/application/features/index.html