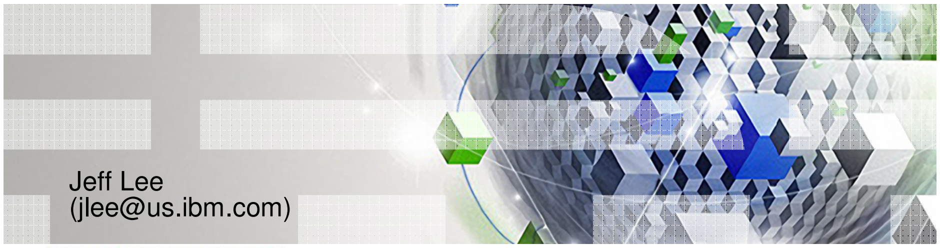
Session Number: 520161

V6R1: The Future of Java™ on IBM i™



Jeff Lee
(jlee@us.ibm.com)

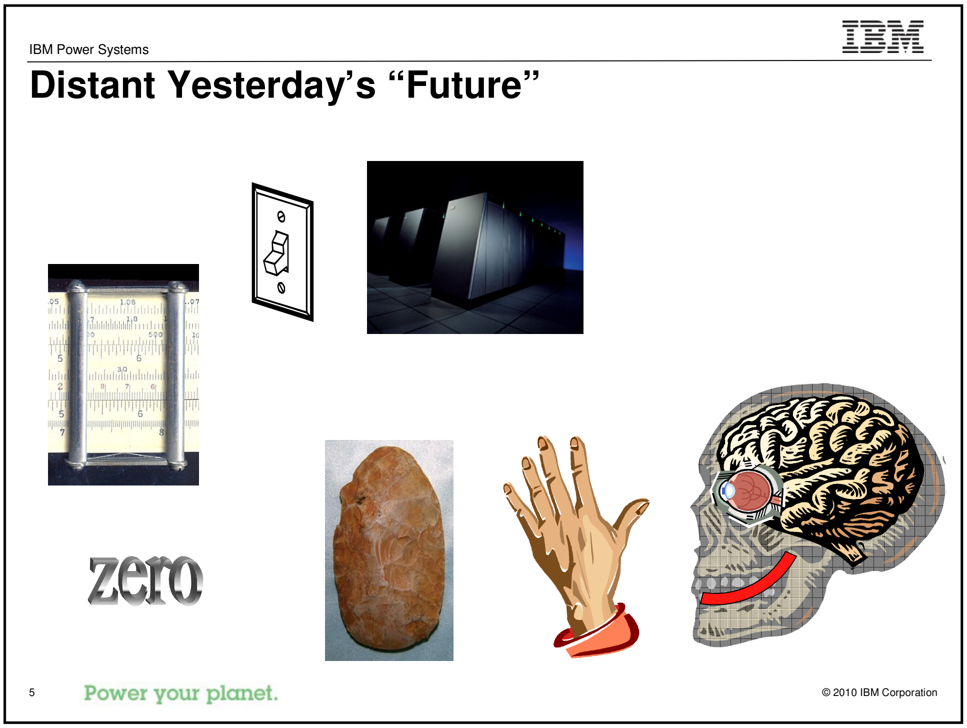
Abstract

- Support for Java™ technology in *IBM i* is in the midst of fundamental change, as IBM converges on a **single virtual machine implementation** across all its platforms. This session details the change embodied by the adoption of this new VM – its **content, timeline, rationale and impacts** – and charts IBM's progress so far.
- This new VM brings with it **known impacts** – good and bad – to compatibility, performance, and long-term migration. These impacts are outlined, along with IBM's plans for their **automatic detection and mitigation** in most cases.
- **Impacts that defy automatic mitigation** are expected to be rare; IBM i facilities for their identification will be described and demonstrated.
- By the end of this session, you will understand:
  - Java **support** in IBM i, and how it is **changing**
  - the **timeline** of the change, and where we are in it
  - the **impacts** of the change, both good and bad
  - how to determine whether to try the new VM **for yourself**

Outline

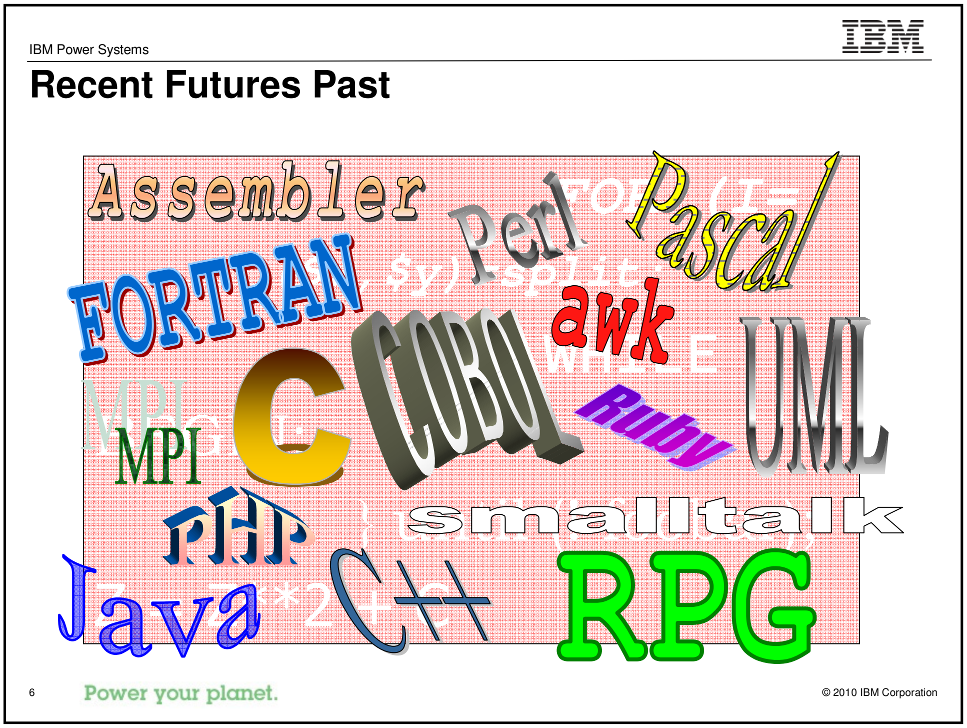
- Introduction
  - Putting "the future" in context
- The Change
  - **Content**: A new "IBM Technology" VM for Java™
  - **Timeline**: Two releases of transition
  - **Rationale**: Software convergence
  - **Impacts**: Compatibility, performance and migration
- Summary

"The Future…



Distant Yesterday's "Future"

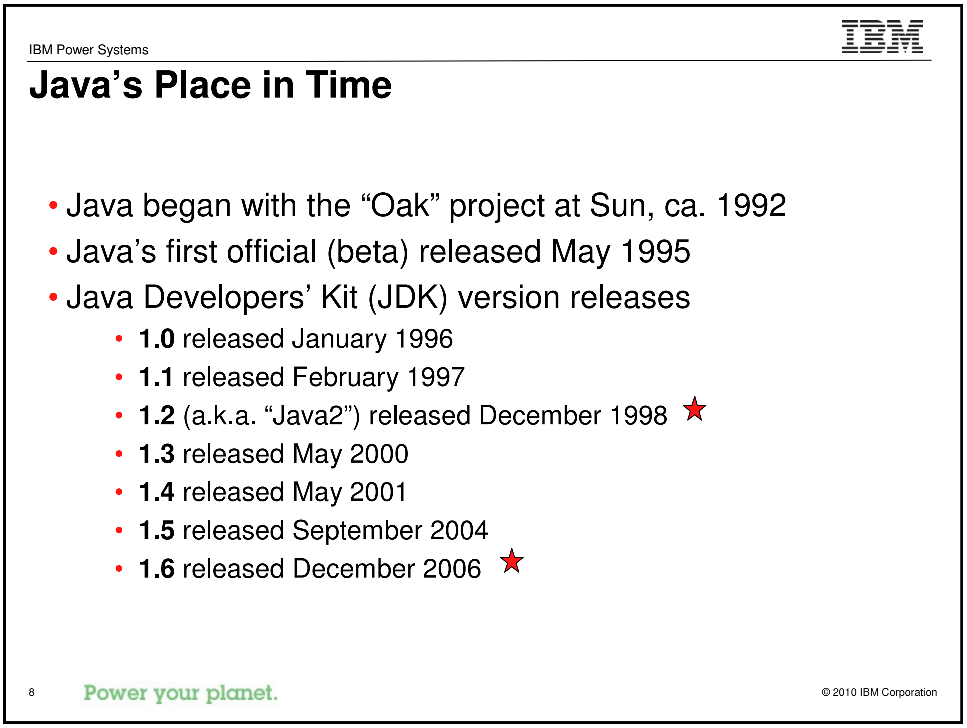
Power your planet.



Recent Futures Past

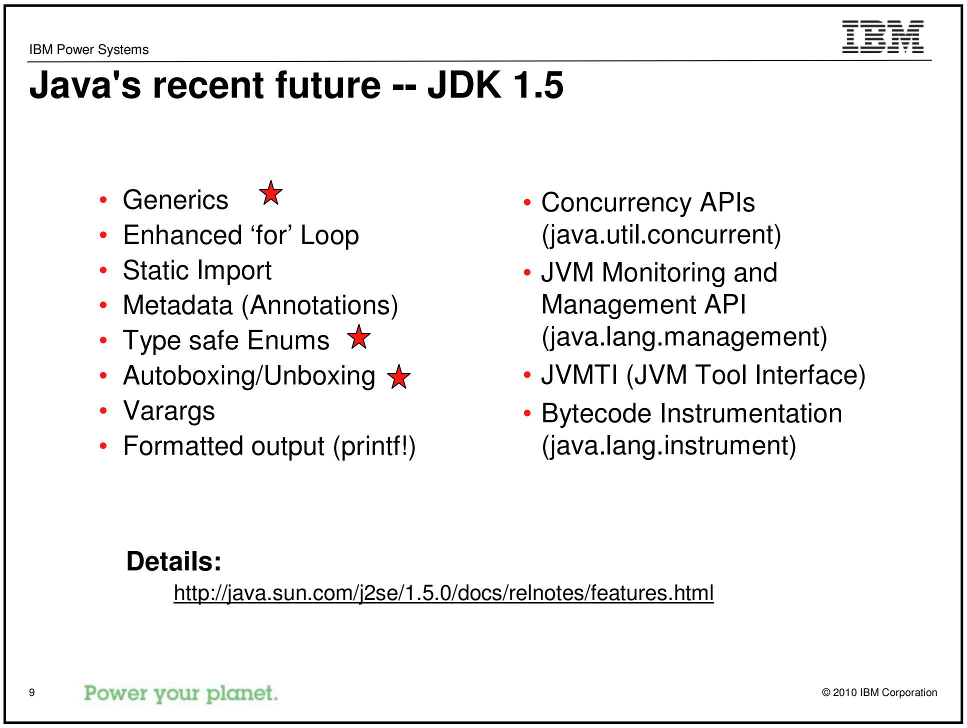
Power your planet.

and what about Java …



Java’s Place in Time

- Java began with the “Oak” project at Sun, ca. 1992
- Java’s first official (beta) released May 1995
- Java Developers’ Kit (JDK) version releases
  - **1.0** released January 1996
  - **1.1** released February 1997
  - **1.2** (a.k.a. “Java2”) released December 1998 ★
  - **1.3** released May 2000
  - **1.4** released May 2001
  - **1.5** released September 2004
  - **1.6** released December 2006 ★



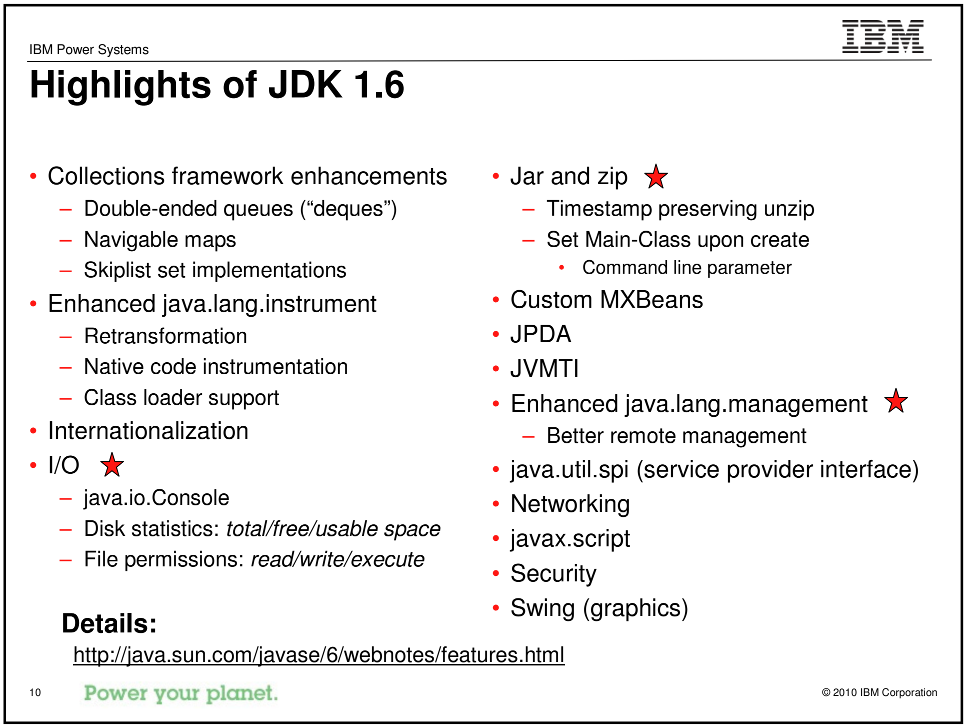
Java's recent future -- JDK 1.5

- Generics ★
- Enhanced 'for' Loop
- Static Import
- Metadata (Annotations)
- Type safe 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)

Details:

http://java.sun.com/j2se/1.5.0/docs/relnotes/features.html



Highlights of JDK 1.6

- Collections framework enhancements
  - Double-ended queues ("deques")
  - Navigable maps
  - Skiplist set implementations
- Enhanced java.lang.instrument
  - Retransformation
  - Native code instrumentation
  - Class loader support
- Internationalization
- I/O ★
  - java.io.Console
  - Disk statistics: *total/free/usable space*
  - File permissions: *read/write/execute*

- Jar and zip ★
  - Timestamp preserving unzip
  - Set Main-Class upon create
    - Command line parameter
- Custom MXBeans
- JPDA
- JVMTI
- Enhanced java.lang.management ★
  - Better remote management
- java.util.spi (service provider interface)
- Networking
- javax.script
- Security
- Swing (graphics)

Details:

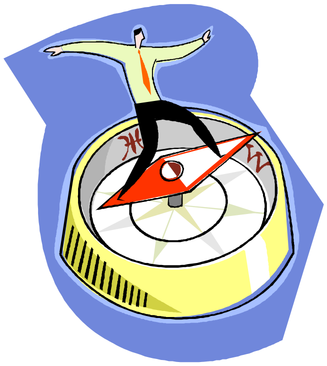
http://java.sun.com/javase/6/webnotes/features.html

…on IBM *i*

…in the midst of fundamental change…

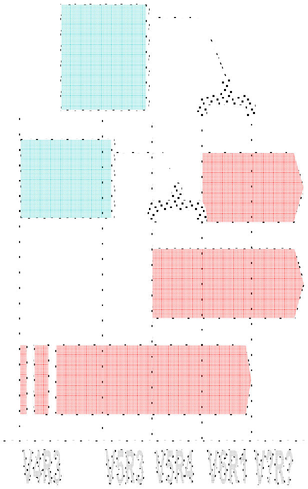
What's the Big Deal?

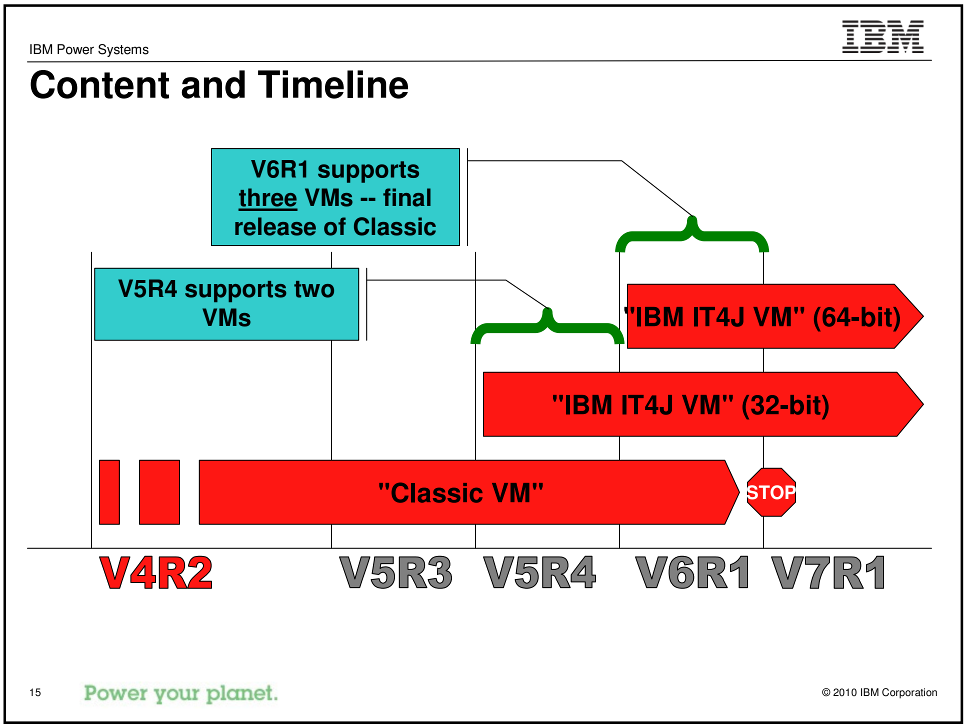
- Things are changing in IBM i
- IBM is **converging** on a **single** virtual machine (VM) for Java
  - Modular, high-performance, 4GL and 5GL codebase
  - Platform-specific code accommodations and adaptation
- IBM proprietary extensions
  - On IBM i for the first time



Content and Timeline

- The venerable 64-bit **"Classic VM"** remains through IBM i **6.1**
  - On the platform since V4R3
  - Has evolved to support each new JDK from Sun
  - Leverages **Sun's** licensed codebase
  - Supports JDK 1.3, 1.4, 1.5, 1.6
- **V5R4**: adds a new **"IBM Technology" (32-bit) VM**
  - New code – not just a port of prior IBM VMs
  - **Independent** of Sun's licensed code
  - Supports JDK 1.5 and 1.6
- **V6R1**: adds new **"IBM Technology" (64-bit) VM**
  - Based on same codebase as new (32-bit) version
  - Supports JDK 1.5 and 1.6





IBM Power Systems

Content and Timeline

V6R1 supports three VMs -- final release of Classic

V5R4 supports two VMs

"IBM IT4J VM" (64-bit)

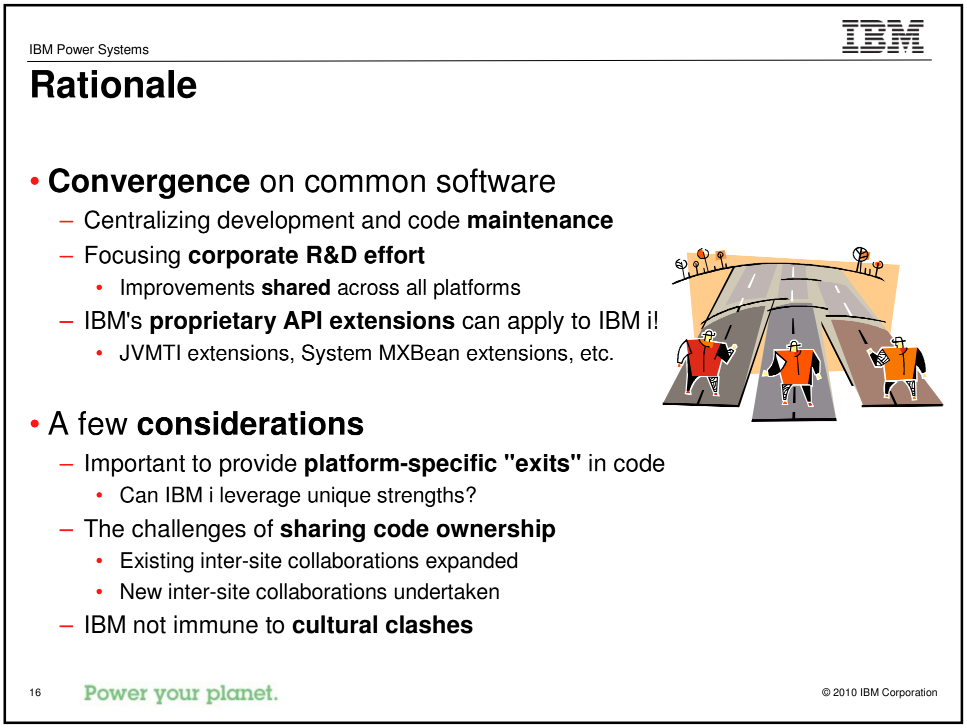
"IBM IT4J VM" (32-bit)

"Classic VM"

STOP

V4R2 V5R3 V5R4 V6R1 V7R1

15 Power your planet. © 2010 IBM Corporation

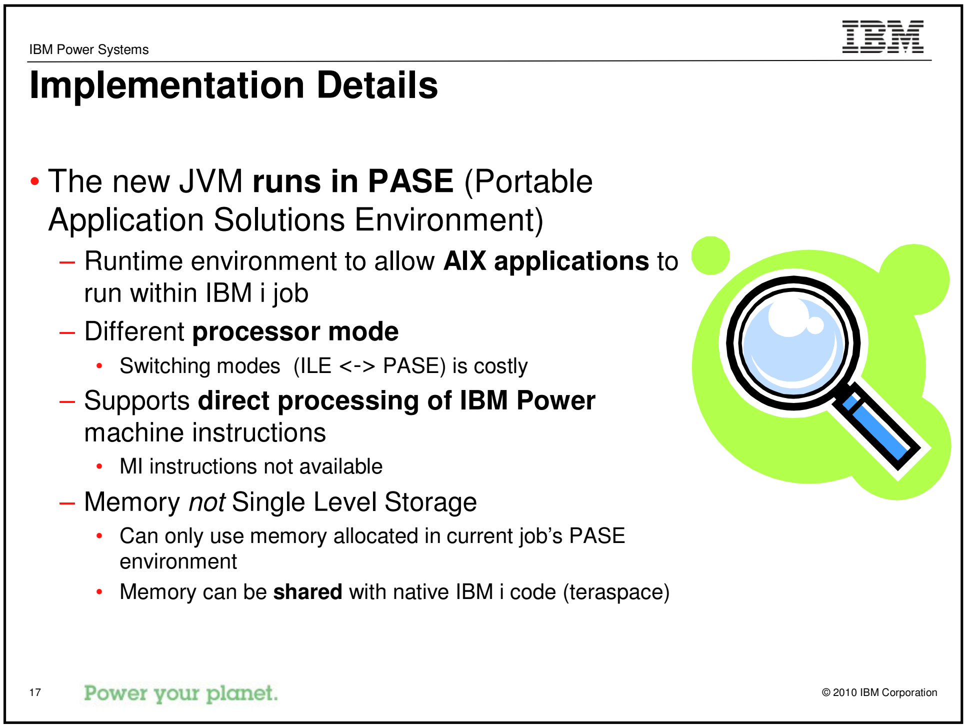


IBM Power Systems

Rationale

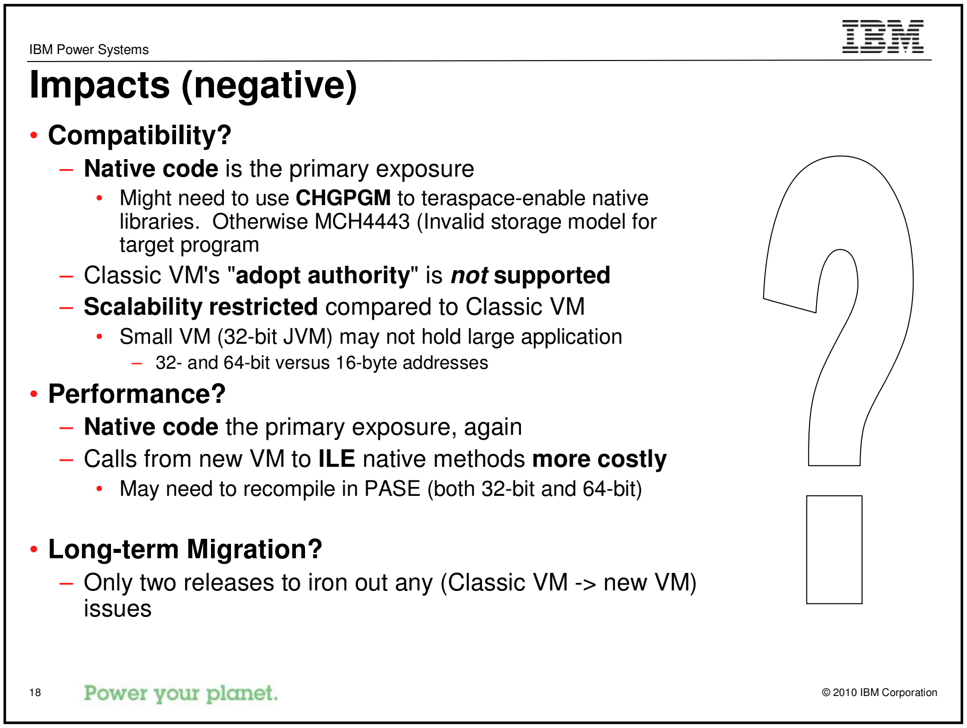
- **Convergence** on common software
  - Centralizing development and code **maintenance**
  - Focusing **corporate R&D effort**
    - Improvements **shared** across all platforms
  - IBM's **proprietary API extensions** can apply to IBM i!
    - JVMTI extensions, System MXBean extensions, etc.
- A few **considerations**
  - Important to provide **platform-specific "exits"** in code
    - Can IBM i leverage unique strengths?
  - The challenges of **sharing code ownership**
    - Existing inter-site collaborations expanded
    - New inter-site collaborations undertaken
  - IBM not immune to **cultural clashes**

16 Power your planet. © 2010 IBM Corporation



Implementation Details

- The new JVM **runs in PASE** (Portable Application Solutions Environment)
  - Runtime environment to allow **AIX applications** to run within IBM i job
  - Different **processor mode**
    - Switching modes (ILE <-> PASE) is costly
  - Supports **direct processing of IBM Power machine instructions**
    - MI instructions not available
  - Memory *not* Single Level Storage
    - Can only use memory allocated in current job's PASE environment
    - Memory can be **shared** with native IBM i code (teraspace)

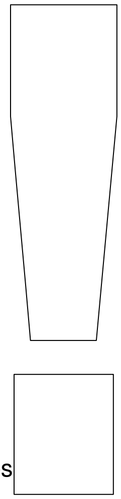


Impacts (negative)

- **Compatibility?**
  - **Native code** is the primary exposure
    - Might need to use **CHGPGM** to teraspace-enable native libraries. Otherwise MCH4443 (Invalid storage model for target program
  - Classic VM's "**adopt authority**" is ***not* supported**
  - **Scalability restricted** compared to Classic VM
    - Small VM (32-bit JVM) may not hold large application
      - 32- and 64-bit versus 16-byte addresses
- **Performance?**
  - **Native code** the primary exposure, again
  - Calls from new VM to **ILE** native methods **more costly**
    - May need to recompile in PASE (both 32-bit and 64-bit)
- **Long-term Migration?**
  - Only two releases to iron out any (Classic VM -> new VM) issues

Impacts (positive)

- **Compatibility**
  - "*Pure Java*" code **inherently compatible**
    - Object code requires **no changes** because of *VM*
  - **JNI** preserves value of ILE native methods
- **Performance**
  - New VM *markedly* **faster than Classic** in most cases
    - Record-setting performance
  - Much **smaller storage footprint** reduces paging effects
    - Heap requirements reduced 30%-40%
  - Optimized for Power hardware
    - Hardware benefits map through to PASE
- **Long-term Migration**
  - Two full releases of **overlap** (Classic VM -> new VM) issues
  - Automatic detection (and mitigation) of adopted authority



Automatic impact identification/mitigation

- Adopted authority
  - In **V5R4**, Classic VM silently **detects and logs**
  - In **V6R1**, adopted authority **not permitted**
    - Enabled with (PRPQ) 5799-AAJ
  - V5R3/V5R4 **tool to detect/report** adopted authority
    - V5R3 PTF SI27769 (Java Group PTF SF99269, level 15 and in JDK 1.5)
    - V5R4 PTF SI27772

    Usage: /QSYS.LIB/QJAVA.LIB/**QJVAADPTL**.PGM *[option]...*

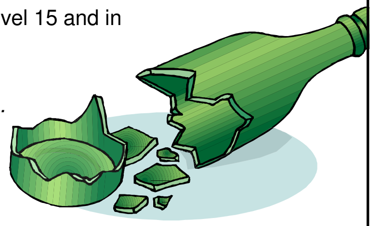
    Valid **options** include

    -h : Show this usage statement.

    -o <file> : Write output to the specified file.

    -d <directory> : Scan only the specified directory tree.

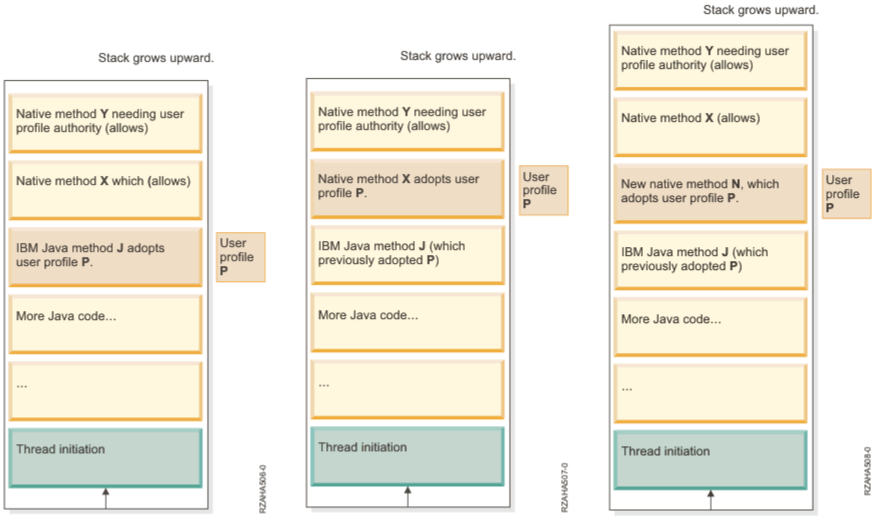
    -noscan : Do not scan system. Report only logged uses.



Example

```
$ /qsys.lib/qjava.lib/qjvaadptl.pgm  -d /home/eberhard

Java Programs with adopted authority that have been
 logged by the JVM:
   /Qibm/Proddata/OS400/jt400/lib/jt400Native.jar
   /QIBM/ProdData/HTTPA/java/lib/wasadmin.jar
   /home/eberhard//FileExists
3 files found.

The following Java Programs were created with the option
 to adopt authority from the owner of the Java Program:
  /home/eberhard/FileExists.class
1 files found.
```

Options for handling adopted authority



Stack grows upward.

Native method **Y** needing user profile authority (allows)

Native method **X** which (allows)

IBM Java method **J** adopts user profile **P**.

More Java code...

...

Thread initiation

User profile **P**

RZAHA506-0

Stack grows upward.

Native method **Y** needing user profile authority (allows)

Native method **X** adopts user profile **P**.

IBM Java method **J** (which previously adopted **P**)

More Java code...

...

Thread initiation

User profile **P**

RZAHA507-0

Stack grows upward.

Native method **Y** needing user profile authority (allows)

Native method **X** (allows)

New native method **N**, which adopts user profile **P**.

IBM Java method **J** (which previously adopted **P**)

More Java code...

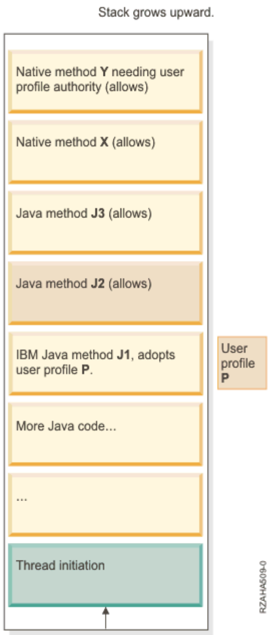
...

Thread initiation

User profile **P**

RZAHA508-0

Options for handling adopted authority *(continued)*


Stack grows upward.

Native method **Y** needing user profile authority (allows)

Native method **X** (allows)

Java method **J3** (allows)

Java method **J2** (allows)

IBM Java method **J1**, adopts user profile **P**.

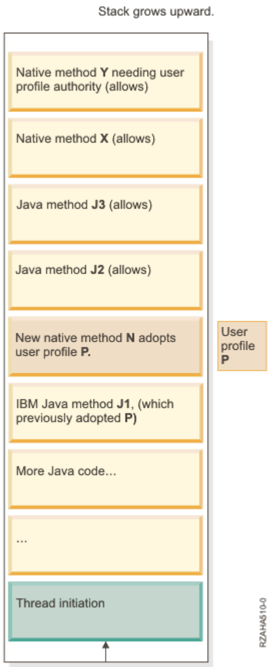
More Java code...

...

Thread initiation

User profile **P**

RZAHA509-0


Stack grows upward.

Native method **Y** needing user profile authority (allows)

Native method **X** (allows)

Java method **J3** (allows)

Java method **J2** (allows)

New native method **N** adopts user profile **P.**

IBM Java method **J1**, (which previously adopted **P**)

More Java code...

...

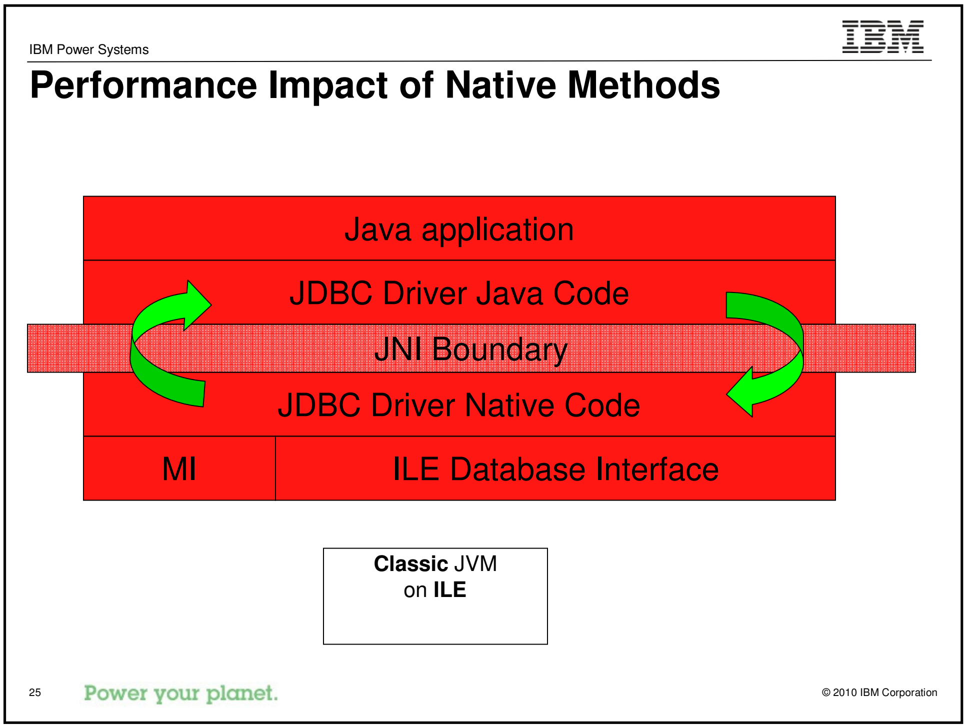
Thread initiation

User profile **P**

RZAHA510-0

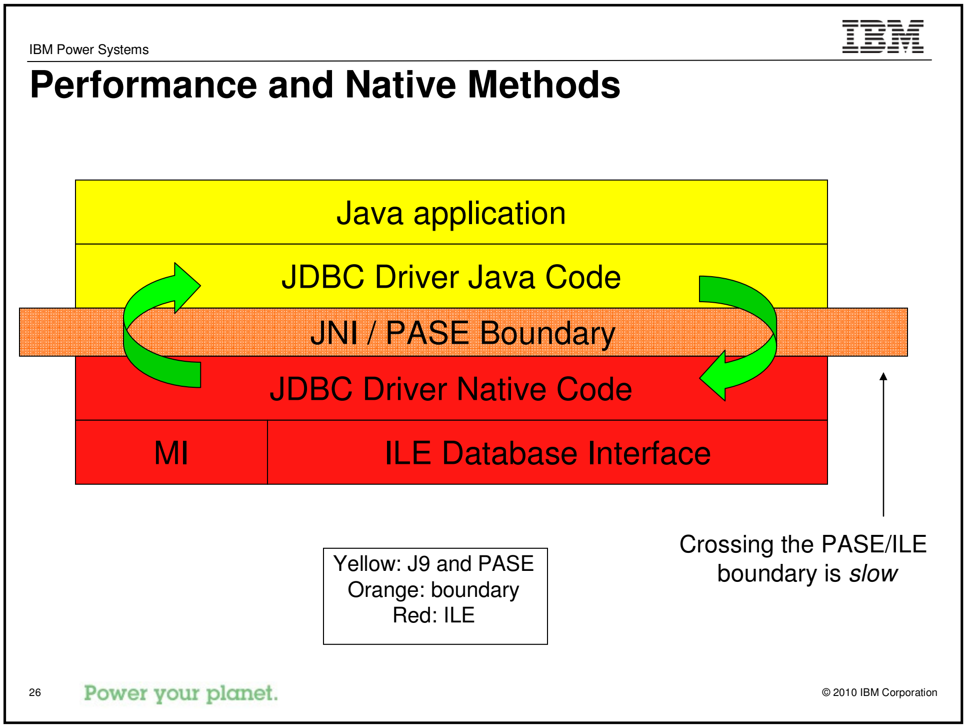
Other Impacts

- For calling 64-bit PASE native methods
  - In **V5R4**, the Classic VM is the only option
  - In **V6R1**, with new 64-bit VM supports them unchanged



Performance Impact of Native Methods

Classic JVM
on **ILE**

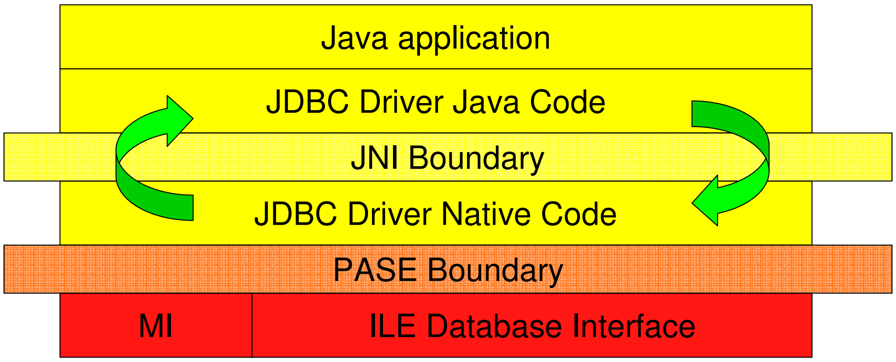


Performance and Native Methods

Crossing the PASE/ILE
boundary is *slow*

Yellow: J9 and PASE
Orange: boundary
Red: ILE

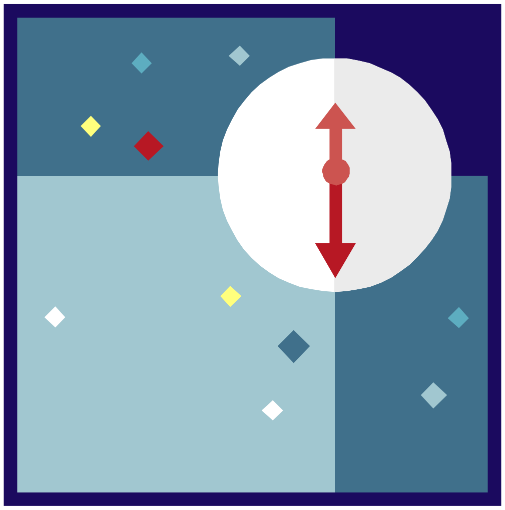
Performance and Native Methods



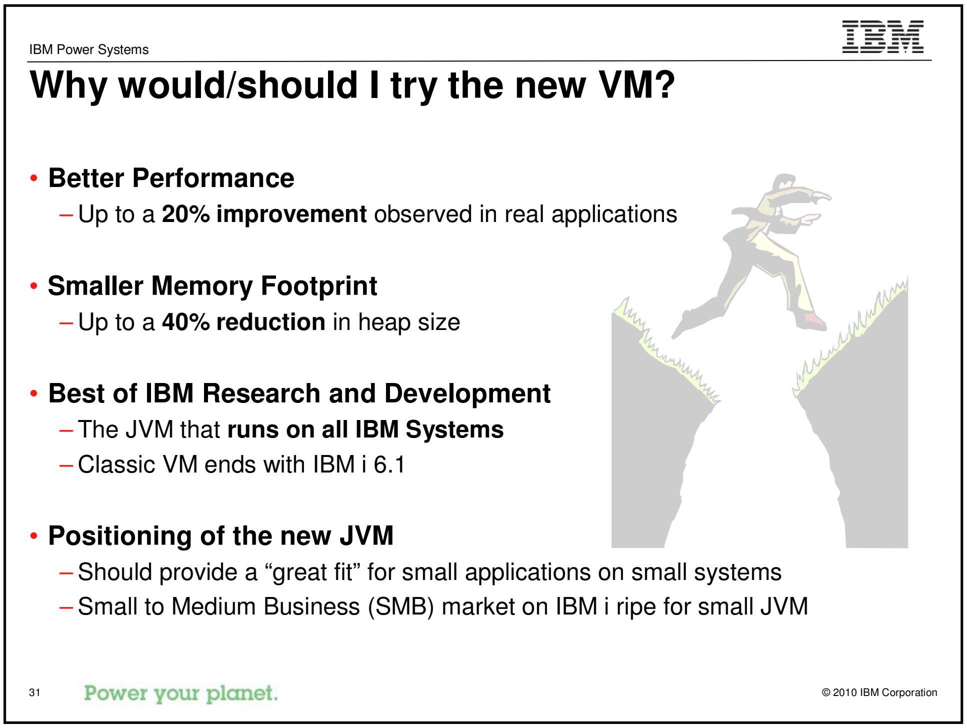
Yellow: J9 and PASE
Orange: boundary
Red: ILE

Native Method Impacts

- **Gotchas** when porting native code (ILE) to PASE
  - **Memset** newly allocated memory that will hold **pointers**
  - Limit size of **automatic storage** (on-stack variables)
    - char buffer[4096];
    - PASE stack size is only 40K!
  - PASE stacks grow *downward*
    - Writing beyond the declared bounds of a variable will corrupt your caller's stack

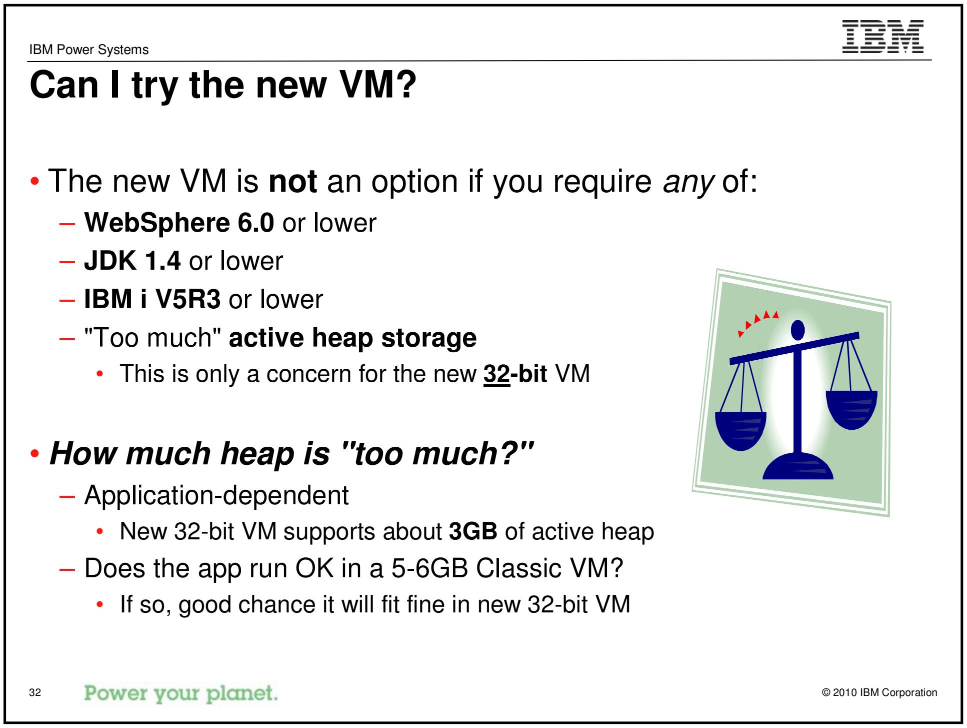


Trying times...



Why would/should I try the new VM?

- **Better Performance**
  - Up to a **20% improvement** observed in real applications
- **Smaller Memory Footprint**
  - Up to a **40% reduction** in heap size
- **Best of IBM Research and Development**
  - The JVM that **runs on all IBM Systems**
  - Classic VM ends with IBM i 6.1
- **Positioning of the new JVM**
  - Should provide a “great fit” for small applications on small systems
  - Small to Medium Business (SMB) market on IBM i ripe for small JVM



Can I try the new VM?

- The new VM is **not** an option if you require *any* of:
  - **WebSphere 6.0** or lower
  - **JDK 1.4** or lower
  - **IBM i V5R3** or lower
  - "Too much" **active heap storage**
    - This is only a concern for the new **32-bit** VM
- ***How much heap is "too much?"***
  - Application-dependent
    - New 32-bit VM supports about **3GB** of active heap
  - Does the app run OK in a 5-6GB Classic VM?
    - If so, good chance it will fit fine in new 32-bit VM

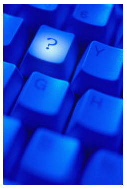
How do I try the new VM?

- Set your job's **JAVA_HOME environment variable** before calling Java
  - From **QShell or QP2TERM**:

```
export JAVA_HOME=/QOpenSys/QIBM/ProdData/JavaVM/jdk50/32bit
```

  - From i5/OS **Command Entry**:

```
ADDENVVAR +
  ENVVAR(JAVA_HOME) +
  VALUE('/QOpenSys/QIBM/ProdData/JavaVM/jdk50/32bit')
```



- The new VM is available on V5R4 in JDK **1.5** and JDK **1.6** versions
  - *Ignores* `java.version=` specs, which only apply to Classic VM
- The **V5R4 default** is **Classic VM** version JDK **1.4**
  - Used whenever `JAVA_HOME` is unset
    - Unless called via `Runtime.exec("java...")` from the new VM!
  - To invoke Classic VM from new VM using `Runtime.exec()` specify environment variable in exec parameters:

```
JAVA_HOME=/QIBM/ProdData/Java400/jdk14
```

- The **V6R1 default** is the **new VM** version: JDK **1.5 32-bit**

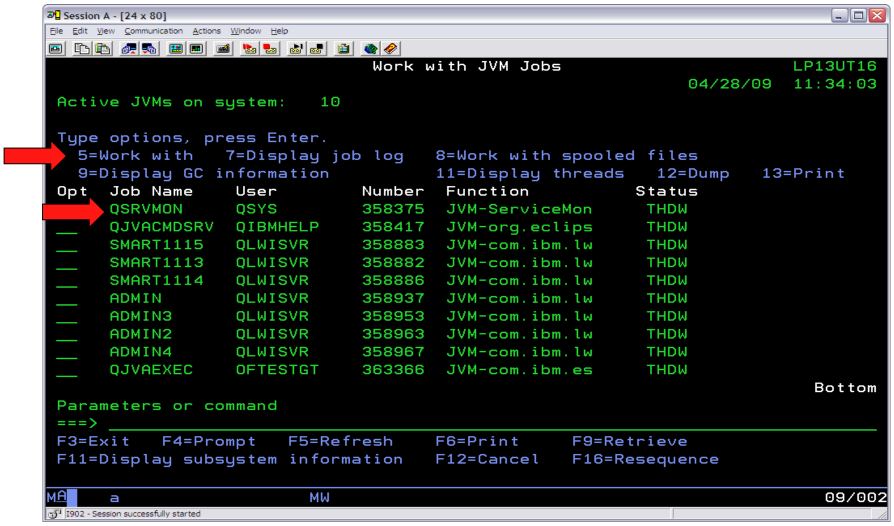
Supported JVM options for IBM i 6.1

5761-JV1 options	JAVA_HOME env var	java.version
Option 6 - **Classic 1.4**	/QIBM/ProdData/Java400/**jdk14**/	1.4
Option 7 - **Classic 5.0**	/QIBM/ProdData/Java400/**jdk15**/	1.5
Option 8 – IBM Technology for Java **5.0 32-bit**	/QOpenSys/QIBM/ProdData/JavaVM/**jdk50/32bit**	1.5
Option 9 – IBM Technology for Java **5.0 64-bit**	/QOpenSys/QIBM/ProdData/JavaVM/**jdk50/64bit**	1.5
Option 10 - **Classic 6**	/QIBM/ProdData/Java400/**jdk6**	1.6
Option 11 – IBM Technology for Java **6 32-bit**	/QOpenSys/QIBM/ProdData/JavaVM/**jdk60/32bit**	1.6
Option 12 – IBM Technology for Java **6 64-bit**	/QOpenSys/QIBM/ProdData/JavaVM/**jdk60/64bit**	1.6

Working with the new VM

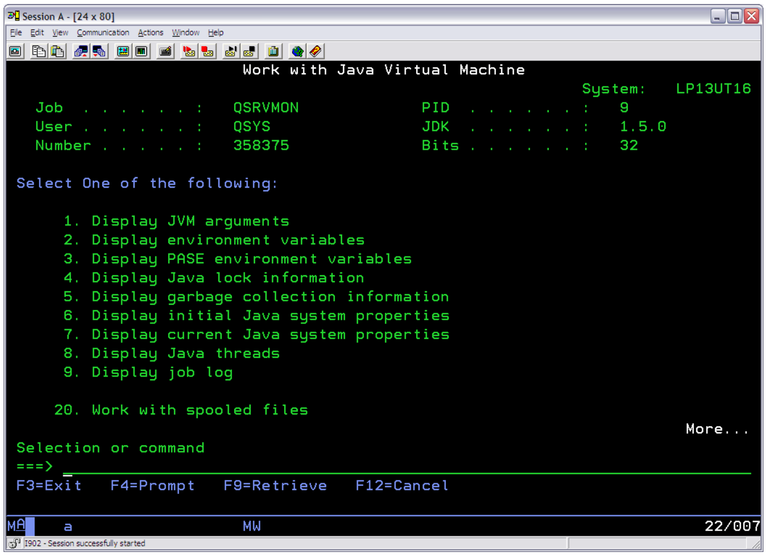
- New **CL Commands** in IBM i **6.1**
  - **WRK**JVMJOB
  - **PRT**JVMJOB
  - **GEN**JVMDMP
- Diagnostic files
  - Java crash information stored in IFS
    - `core.*.dmp, javacore.*.txt, Snap*.trc`

WRKJVMJOB



```
                              Work with JVM Jobs                      LP13UT16
                                                             04/28/09  11:34:03
 Active JVMs on system:    10

 Type options, press Enter.
   5=Work with    7=Display job log    8=Work with spooled files
   9=Display GC information            11=Display threads    12=Dump    13=Print

 Opt  Job Name     User         Number   Function             Status
      QSRVMON      QSYS         358375   JVM-ServiceMon        THDW
 __   QJVACMDSRV   QIBMHELP     358417   JVM-org.eclips        THDW
 __   SMART1115    QLWISVR      358883   JVM-com.ibm.lw        THDW
 __   SMART1113    QLWISVR      358882   JVM-com.ibm.lw        THDW
 __   SMART1114    QLWISVR      358886   JVM-com.ibm.lw        THDW
 __   ADMIN        QLWISVR      358937   JVM-com.ibm.lw        THDW
 __   ADMIN3       QLWISVR      358953   JVM-com.ibm.lw        THDW
 __   ADMIN2       QLWISVR      358963   JVM-com.ibm.lw        THDW
 __   ADMIN4       QLWISVR      358967   JVM-com.ibm.lw        THDW
 __   QJVAEXEC     OFTESTGT     363366   JVM-com.ibm.es        THDW
                                                                        Bottom
 Parameters or command
 ===> ____________________________________________________________________
 F3=Exit   F4=Prompt   F5=Refresh    F6=Print     F9=Retrieve
 F11=Display subsystem information   F12=Cancel   F16=Resequence

MA    a                  MW                                            09/002
I902 - Session successfully started
```

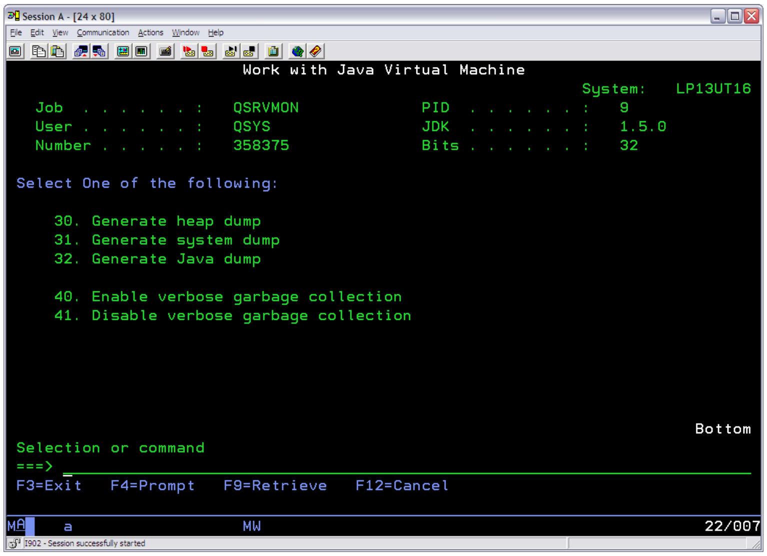
WRKJVMJOB Option 5



```
                        Work with Java Virtual Machine
                                                        System:   LP13UT16
   Job  . . . . . . . :   QSRVMON              PID  . . . . . . :   9
   User . . . . . . . :   QSYS                 JDK  . . . . . . :   1.5.0
   Number . . . . . . :   358375               Bits . . . . . . :   32

 Select One of the following:

      1. Display JVM arguments
      2. Display environment variables
      3. Display PASE environment variables
      4. Display Java lock information
      5. Display garbage collection information
      6. Display initial Java system properties
      7. Display current Java system properties
      8. Display Java threads
      9. Display job log

     20. Work with spooled files
                                                                       More...
 Selection or command
 ===> _
 F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Session A - [24 x 80]
File Edit View Communication Actions Window Help
MA a MW 22/007
I902 - Session successfully started

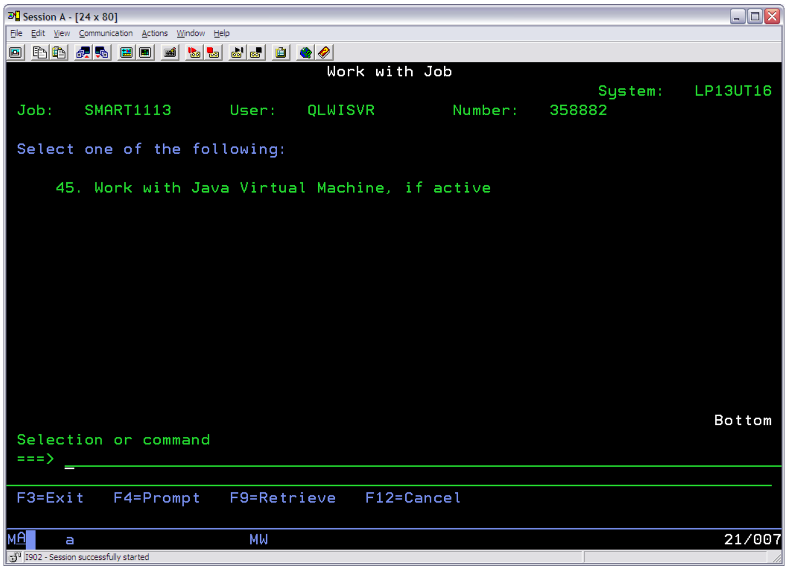
WRKJVMJOB Option 5



```
                        Work with Java Virtual Machine
                                                        System:   LP13UT16
   Job  . . . . . . . :   QSRVMON              PID  . . . . . . :   9
   User . . . . . . . :   QSYS                 JDK  . . . . . . :   1.5.0
   Number . . . . . . :   358375               Bits . . . . . . :   32

 Select One of the following:

     30. Generate heap dump
     31. Generate system dump
     32. Generate Java dump

     40. Enable verbose garbage collection
     41. Disable verbose garbage collection








                                                                        Bottom
 Selection or command
 ===> _
 F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

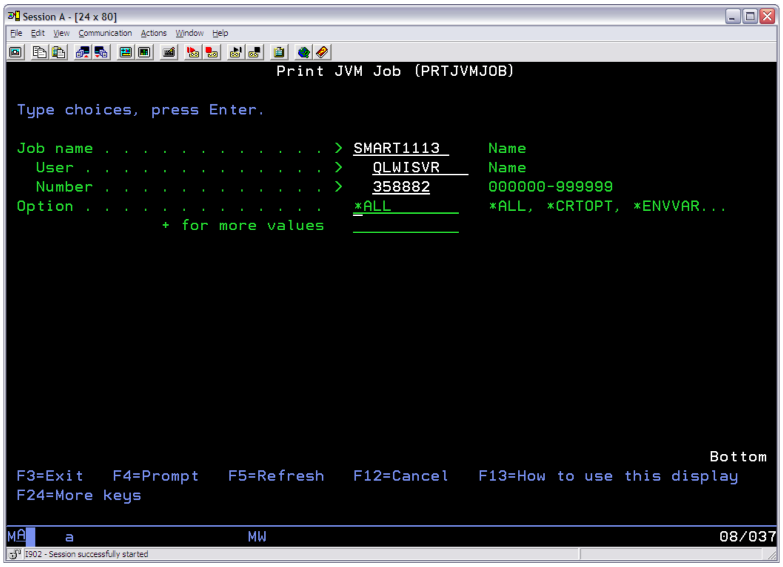
Session A - [24 x 80]
File Edit View Communication Actions Window Help
MA a MW 22/007
I902 - Session successfully started

JVM information from WRKJOB



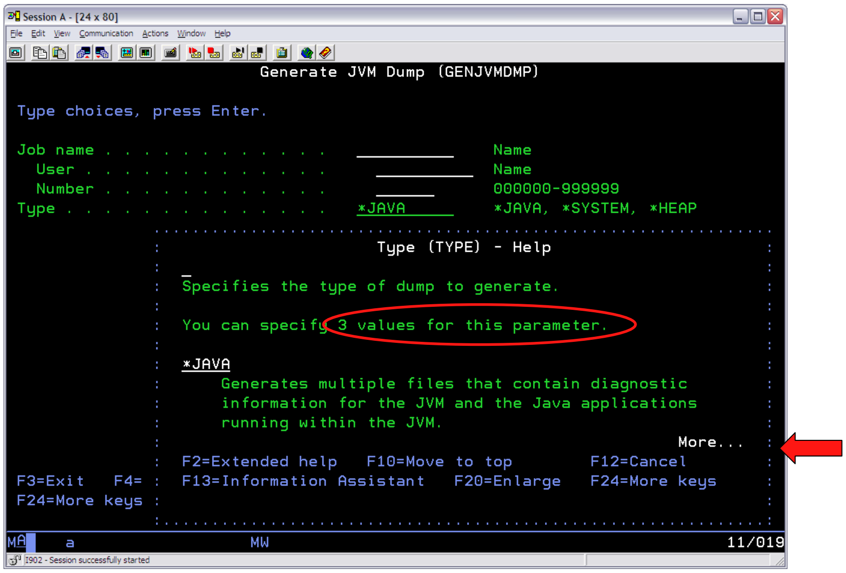
```
                                Work with Job
                                                        System:   LP13UT16
Job:   SMART1113      User:   QLWISVR        Number:   358882

Select one of the following:

    45. Work with Java Virtual Machine, if active










                                                                      Bottom
Selection or command
===> _

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel

MA     a                  MW                                         21/007
I902 - Session successfully started
```

PRTJVMJOB



```
                           Print JVM Job (PRTJVMJOB)

Type choices, press Enter.

Job name . . . . . . . . . . . . > SMART1113     Name
  User . . . . . . . . . . . . . >   QLWISVR     Name
  Number . . . . . . . . . . . . >   358882      000000-999999
Option . . . . . . . . . . . . .   *ALL          *ALL, *CRTOPT, *ENVVAR...
               + for more values











                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys

MA     a                  MW                                         08/037
I902 - Session successfully started
```

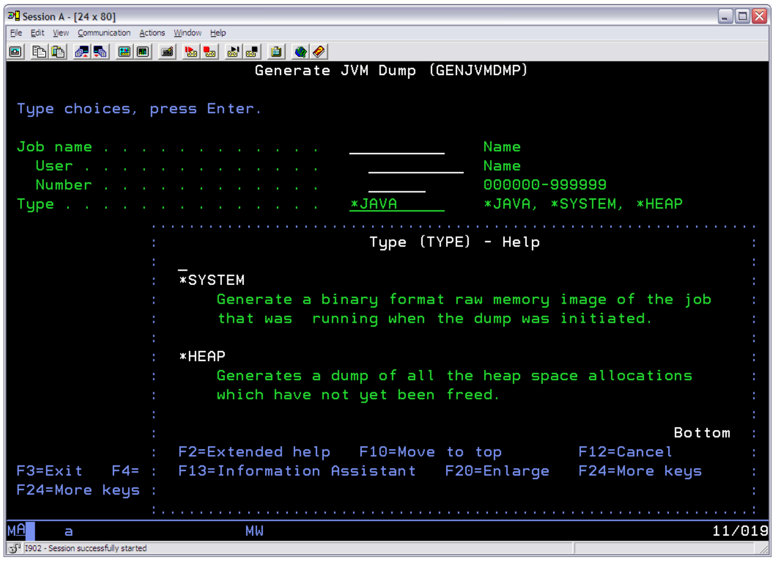
GENJVMDMP



```
                             Generate JVM Dump (GENJVMDMP)

 Type choices, press Enter.

 Job name . . . . . . . . . . . .   __________   Name
   User . . . . . . . . . . . . .     __________   Name
   Number . . . . . . . . . . . .     ______       000000-999999
 Type . . . . . . . . . . . . . .   *JAVA          *JAVA, *SYSTEM, *HEAP
               ...............................................................
               :                      Type (TYPE) - Help                      :
               :                                                              :
               : Specifies the type of dump to generate.                     :
               :                                                              :
               : You can specify 3 values for this parameter.                 :
               :                                                              :
               : *JAVA                                                        :
               :     Generates multiple files that contain diagnostic         :
               :     information for the JVM and the Java applications        :
               :     running within the JVM.                                  :
               :                                                     More...  :
               : F2=Extended help   F10=Move to top       F12=Cancel          :
 F3=Exit   F4= : F13=Information Assistant   F20=Enlarge   F24=More keys      :
 F24=More keys :                                                              :
               :..............................................................:
 MA     a                    MW                                          11/019
 I902 - Session successfully started
```

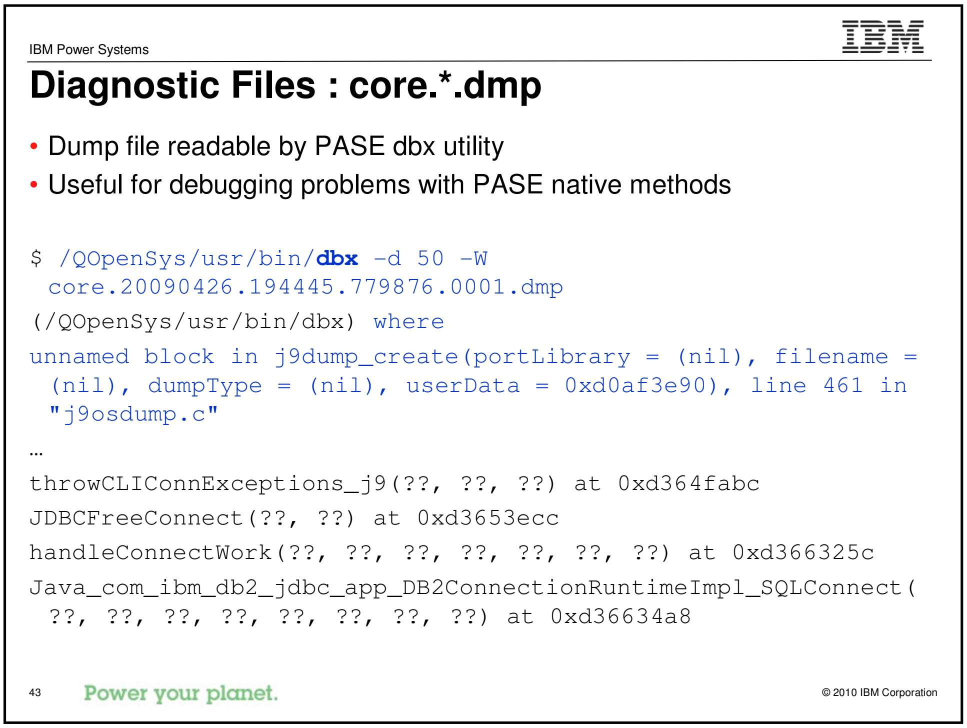
Power your planet.

GENJVMDMP



```
                             Generate JVM Dump (GENJVMDMP)

 Type choices, press Enter.

 Job name . . . . . . . . . . . .   __________   Name
   User . . . . . . . . . . . . .     __________   Name
   Number . . . . . . . . . . . .     ______       000000-999999
 Type . . . . . . . . . . . . . .   *JAVA          *JAVA, *SYSTEM, *HEAP
               ...............................................................
               :                      Type (TYPE) - Help                      :
               :                                                              :
               : *SYSTEM                                                      :
               :     Generate a binary format raw memory image of the job     :
               :     that was  running when the dump was initiated.           :
               :                                                              :
               : *HEAP                                                        :
               :     Generates a dump of all the heap space allocations       :
               :     which have not yet been freed.                           :
               :                                                              :
               :                                                      Bottom  :
               : F2=Extended help   F10=Move to top       F12=Cancel          :
 F3=Exit   F4= : F13=Information Assistant   F20=Enlarge   F24=More keys      :
 F24=More keys :                                                              :
               :..............................................................:
 MA     a                    MW                                          11/019
 I902 - Session successfully started
```

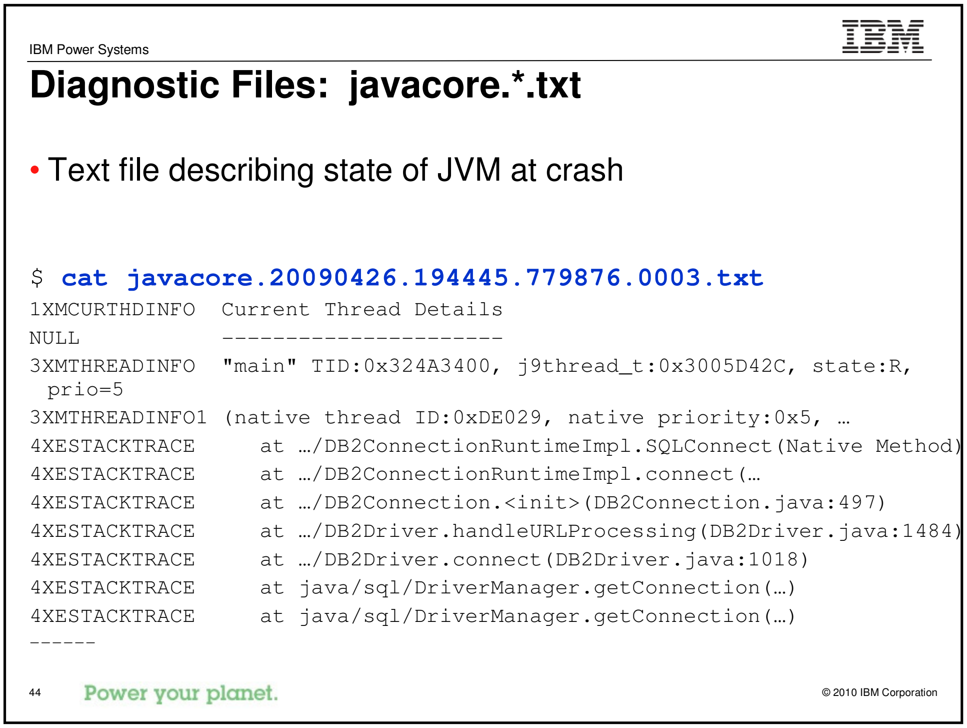
Power your planet.



Diagnostic Files : core.*.dmp

- Dump file readable by PASE dbx utility
- Useful for debugging problems with PASE native methods

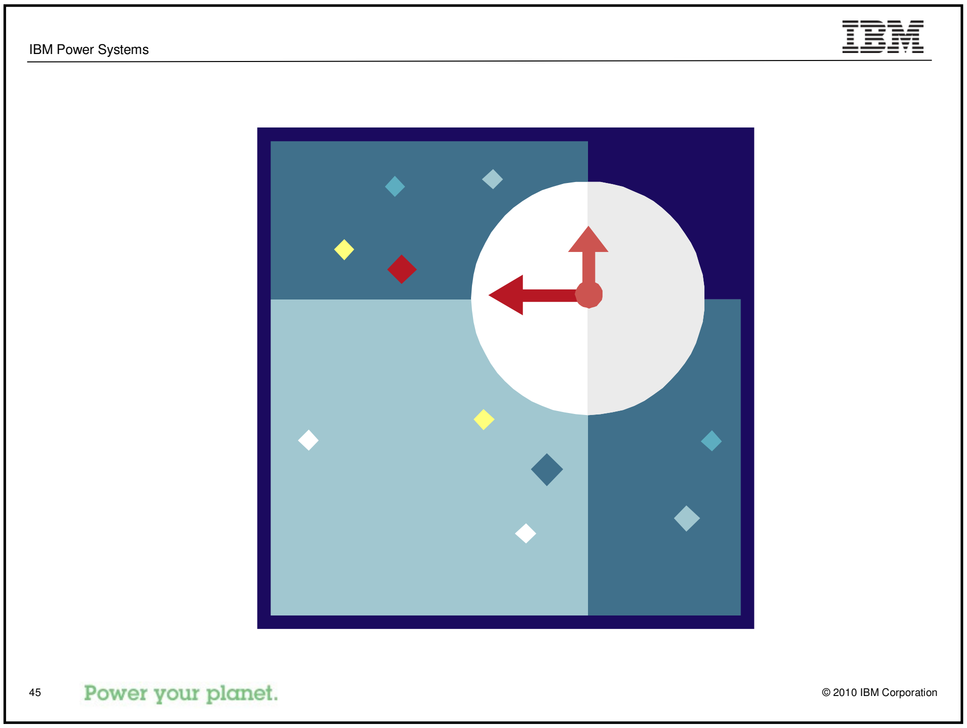
```
$ /QOpenSys/usr/bin/dbx -d 50 -W
 core.20090426.194445.779876.0001.dmp
(/QOpenSys/usr/bin/dbx) where
unnamed block in j9dump_create(portLibrary = (nil), filename =
  (nil), dumpType = (nil), userData = 0xd0af3e90), line 461 in
 "j9osdump.c"
…
throwCLIConnExceptions_j9(??, ??, ??) at 0xd364fabc
JDBCFreeConnect(??, ??) at 0xd3653ecc
handleConnectWork(??, ??, ??, ??, ??, ??, ??) at 0xd366325c
Java_com_ibm_db2_jdbc_app_DB2ConnectionRuntimeImpl_SQLConnect(
 ??, ??, ??, ??, ??, ??, ??, ??) at 0xd36634a8
```

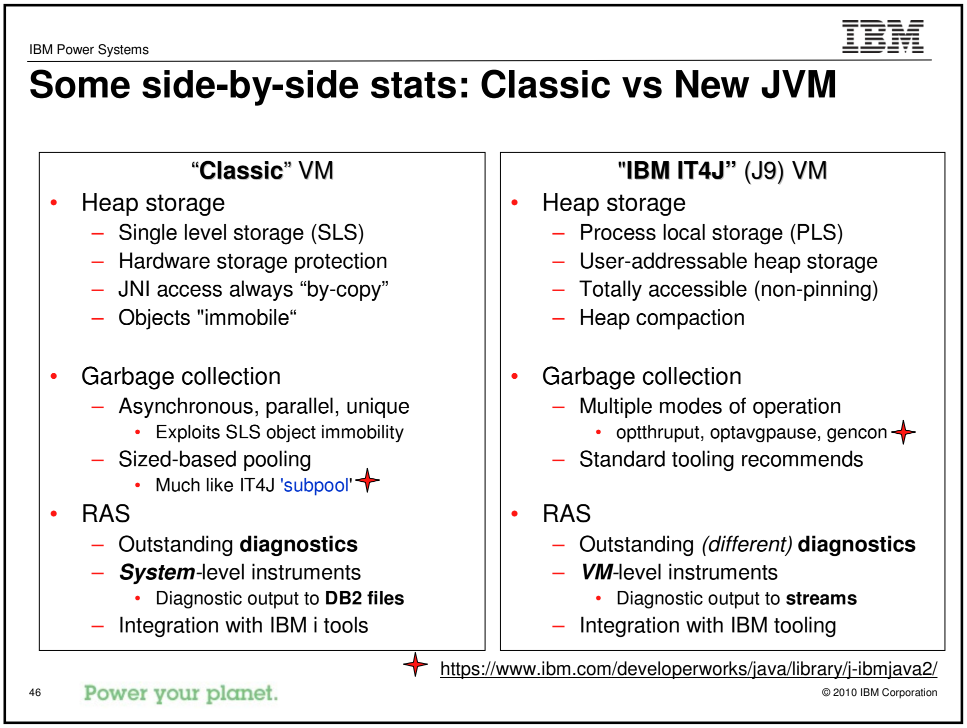


Diagnostic Files: javacore.*.txt

- Text file describing state of JVM at crash

```
$ cat javacore.20090426.194445.779876.0003.txt
1XMCURTHDINFO  Current Thread Details
NULL           ----------------------
3XMTHREADINFO  "main" TID:0x324A3400, j9thread_t:0x3005D42C, state:R,
 prio=5
3XMTHREADINFO1 (native thread ID:0xDE029, native priority:0x5, …
4XESTACKTRACE     at …/DB2ConnectionRuntimeImpl.SQLConnect(Native Method)
4XESTACKTRACE     at …/DB2ConnectionRuntimeImpl.connect(…
4XESTACKTRACE     at …/DB2Connection.<init>(DB2Connection.java:497)
4XESTACKTRACE     at …/DB2Driver.handleURLProcessing(DB2Driver.java:1484)
4XESTACKTRACE     at …/DB2Driver.connect(DB2Driver.java:1018)
4XESTACKTRACE     at java/sql/DriverManager.getConnection(…)
4XESTACKTRACE     at java/sql/DriverManager.getConnection(…)
------
```





Some side-by-side stats: Classic vs New JVM

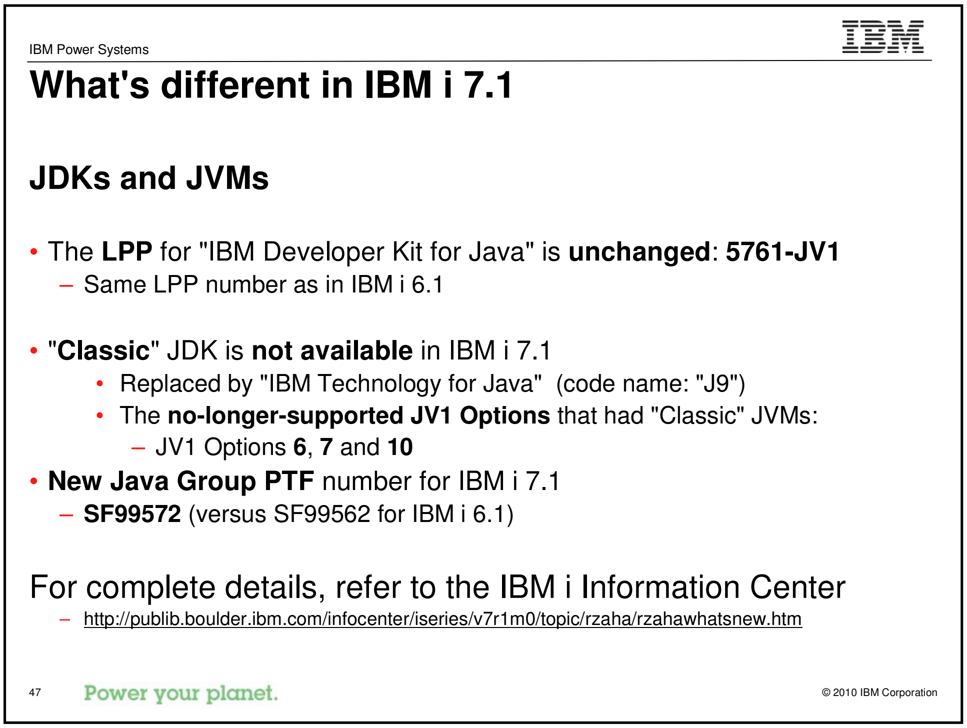
“**Classic**” VM

- Heap storage
  - Single level storage (SLS)
  - Hardware storage protection
  - JNI access always “by-copy”
  - Objects "immobile“
- Garbage collection
  - Asynchronous, parallel, unique
    - Exploits SLS object immobility
  - Sized-based pooling
    - Much like IT4J 'subpool'✦
- RAS
  - Outstanding **diagnostics**
  - ***System***-level instruments
    - Diagnostic output to **DB2 files**
  - Integration with IBM i tools

"**IBM IT4J**” (J9) VM

- Heap storage
  - Process local storage (PLS)
  - User-addressable heap storage
  - Totally accessible (non-pinning)
  - Heap compaction
- Garbage collection
  - Multiple modes of operation
    - optthruput, optavgpause, gencon✦
  - Standard tooling recommends
- RAS
  - Outstanding *(different)* **diagnostics**
  - ***VM***-level instruments
    - Diagnostic output to **streams**
  - Integration with IBM tooling

✦ https://www.ibm.com/developerworks/java/library/j-ibmjava2/



IBM Power Systems

What's different in IBM i 7.1

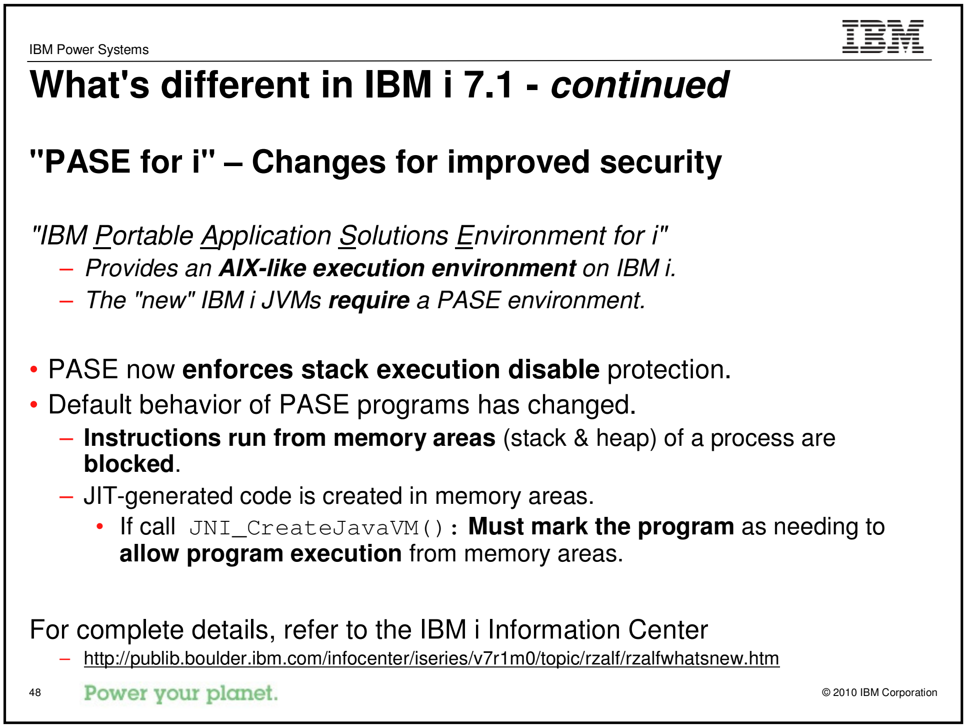
JDKs and JVMs

- The **LPP** for "IBM Developer Kit for Java" is **unchanged**: **5761-JV1**
  - Same LPP number as in IBM i 6.1
- "**Classic**" JDK is **not available** in IBM i 7.1
  - Replaced by "IBM Technology for Java" (code name: "J9")
  - The **no-longer-supported JV1 Options** that had "Classic" JVMs:
    - JV1 Options **6**, **7** and **10**
- **New Java Group PTF** number for IBM i 7.1
  - **SF99572** (versus SF99562 for IBM i 6.1)

For complete details, refer to the IBM i Information Center

- http://publib.boulder.ibm.com/infocenter/iseries/v7r1m0/topic/rzaha/rzahawhatsnew.htm

47 Power your planet.



IBM Power Systems

What's different in IBM i 7.1 - *continued*

"PASE for i" – Changes for improved security

"IBM Portable Application Solutions Environment for i"

- *Provides an **AIX-like execution environment** on IBM i.*
- *The "new" IBM i JVMs **require** a PASE environment.*

- PASE now **enforces stack execution disable** protection.
- Default behavior of PASE programs has changed.
  - **Instructions run from memory areas** (stack & heap) of a process are **blocked**.
  - JIT-generated code is created in memory areas.
    - If call `JNI_CreateJavaVM()`: **Must mark the program** as needing to **allow program execution** from memory areas.

For complete details, refer to the IBM i Information Center

- http://publib.boulder.ibm.com/infocenter/iseries/v7r1m0/topic/rzalf/rzalfwhatsnew.htm

48 Power your planet.

Summary and review

- Content and timeline:

  In its recent releases, *IBM i* is embracing IBM's **new VM** for Java -- first in a **32**-bit form and then in a 64-bit form -- to ultimately **replace** our 64-bit Classic VM.

- Rationale:

  This move represents IBM's ongoing efforts to **converge** on the very **best** of our own **corporate technology**.

- Impacts:

  '**Pure**' Java applications should benefit directly; impacts to applications using **i-specific code** vary less-predictably.

- For more information...

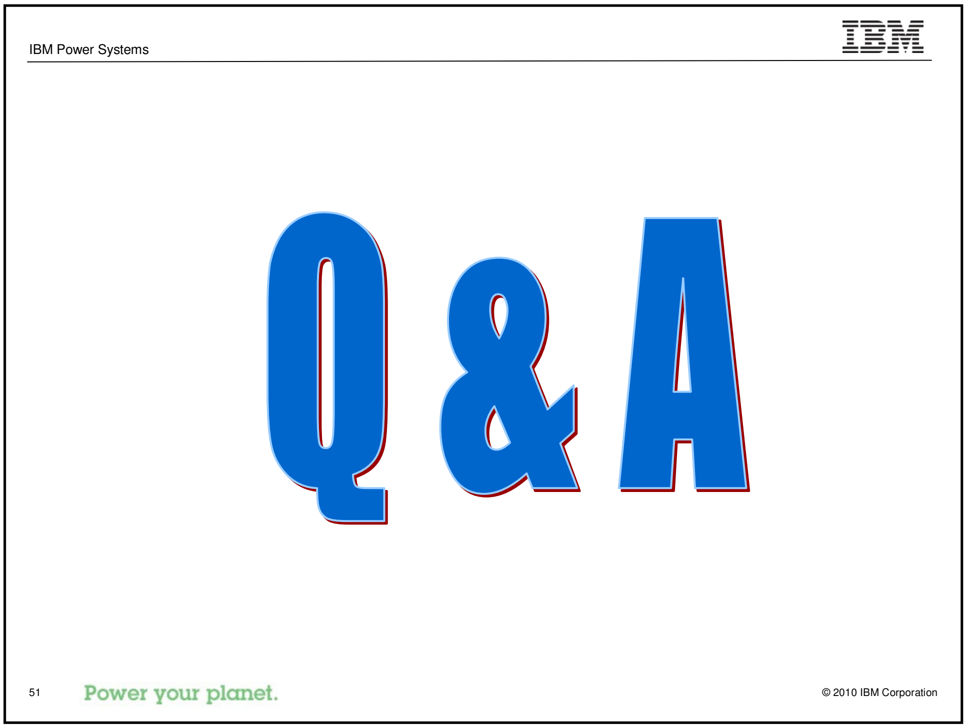
Resources

- IBM Technology for Java Virtual Machine in IBM i5/OS
  - ***Published February 2007***
  - www.redbooks.ibm.com/redbooks/pdfs/sg247353.pdf (6.5MB)

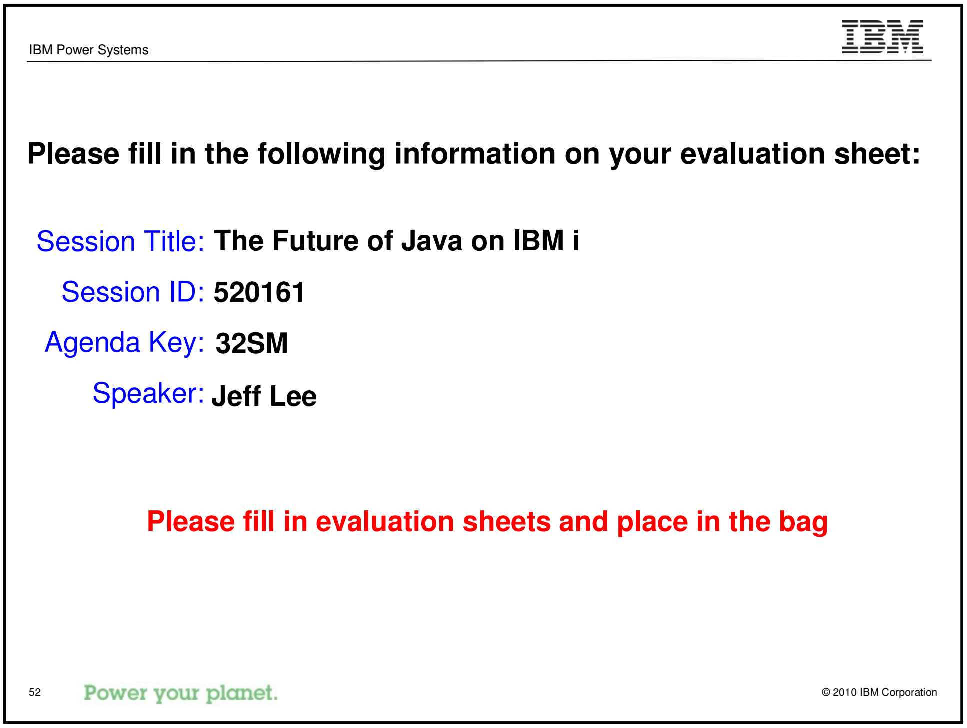
- IBM® Developer Kit and Runtime Environment, Java™ 2 Technology Edition, Version 5.0 -- Diagnostics Guide
  - http://download.boulder.ibm.com/ibmdl/pub/software/dw/jdk/diagnosis/diag50.pdf (4.5MB)

- Porting UNIX Applications Using AS/400 PASE (Published 2000)
  - www.redbooks.ibm.com/redbooks/pdfs/sg245970.pdf

- What's New in Java, in IBM i 7.1
  - http://publib.boulder.ibm.com/infocenter/iseries/v7r1m0/topic/rzaha/rzahawhatsnew.htm



Q & A



Please fill in the following information on your evaluation sheet:

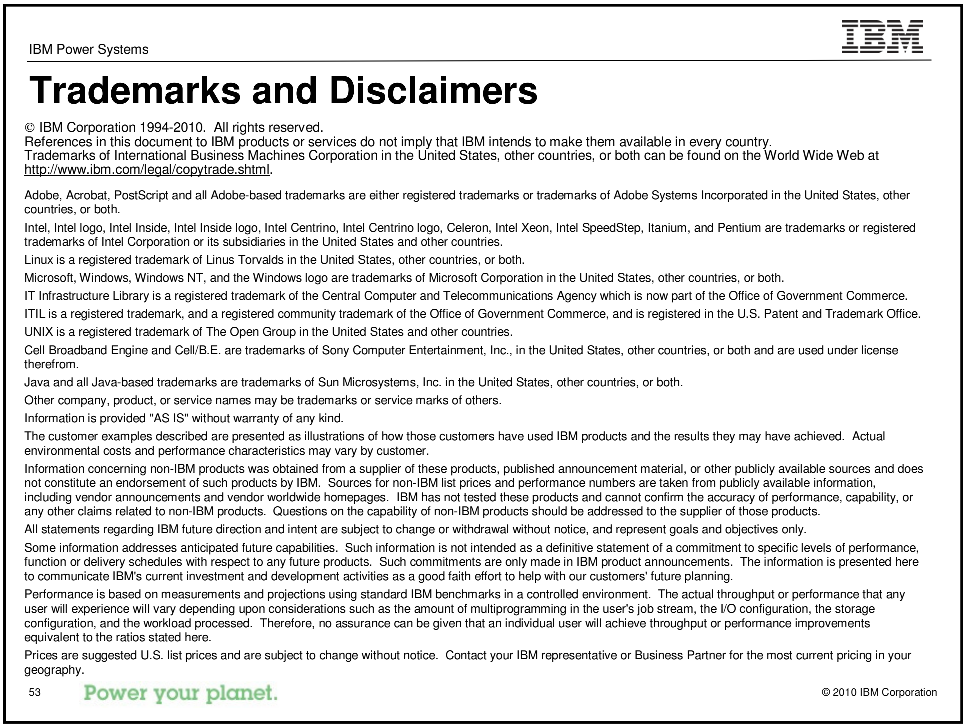
Session Title: **The Future of Java on IBM i**

Session ID: **520161**

Agenda Key: **32SM**

Speaker: **Jeff Lee**

Please fill in evaluation sheets and place in the bag



IBM Power Systems

Trademarks and Disclaimers

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.

Adobe, Acrobat, PostScript and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated 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 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.

Cell Broadband Engine and Cell/B.E. are trademarks of Sony Computer Entertainment, Inc., in the United States, other countries, or both and are used under license therefrom.

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.

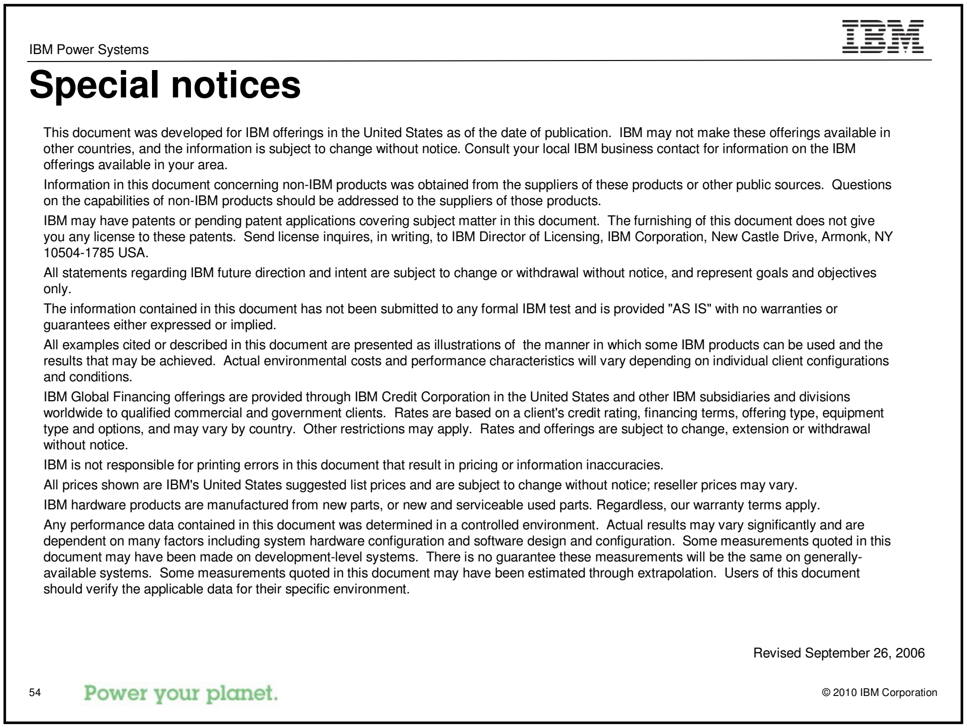
All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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. Contact your IBM representative or Business Partner for the most current pricing in your geography.

53 Power your planet. © 2010 IBM Corporation



IBM Power Systems

Special notices

This document was developed for IBM offerings in the United States as of the date of publication. IBM may not make these offerings available in other countries, and the information is subject to change without notice. Consult your local IBM business contact for information on the IBM offerings available in your area.

Information in this document concerning non-IBM products was obtained from the suppliers of these products or other public sources. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquires, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

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

The information contained in this document has not been submitted to any formal IBM test and is provided "AS IS" with no warranties or guarantees either expressed or implied.

All examples cited or described in this document are presented as illustrations of the manner in which some IBM products can be used and the results that may be achieved. Actual environmental costs and performance characteristics will vary depending on individual client configurations and conditions.

IBM Global Financing offerings are provided through IBM Credit Corporation in the United States and other IBM subsidiaries and divisions worldwide to qualified commercial and government clients. Rates are based on a client's credit rating, financing terms, offering type, equipment type and options, and may vary by country. Other restrictions may apply. Rates and offerings are subject to change, extension or withdrawal without notice.

IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies.

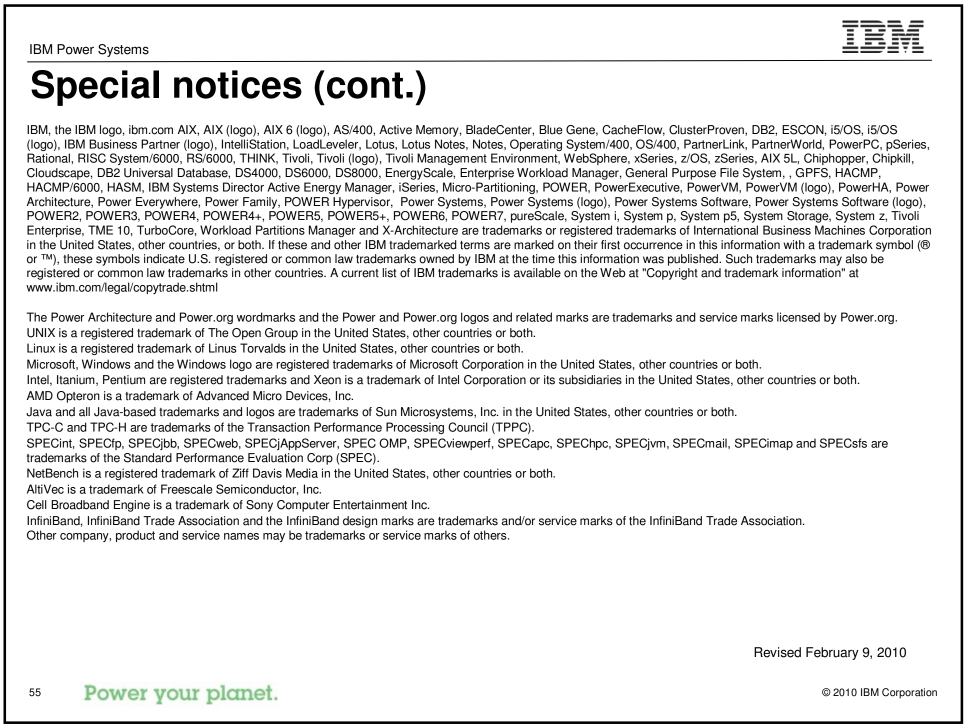
All prices shown are IBM's United States suggested list prices and are subject to change without notice; reseller prices may vary.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Users of this document should verify the applicable data for their specific environment.

Revised September 26, 2006

54 Power your planet. © 2010 IBM Corporation



Special notices (cont.)

IBM, the IBM logo, ibm.com AIX, AIX (logo), AIX 6 (logo), AS/400, Active Memory, BladeCenter, Blue Gene, CacheFlow, ClusterProven, DB2, ESCON, i5/OS, i5/OS (logo), IBM Business Partner (logo), IntelliStation, LoadLeveler, Lotus, Lotus Notes, Notes, Operating System/400, OS/400, PartnerLink, PartnerWorld, PowerPC, pSeries, Rational, RISC System/6000, RS/6000, THINK, Tivoli, Tivoli (logo), Tivoli Management Environment, WebSphere, xSeries, z/OS, zSeries, AIX 5L, Chiphopper, Chipkill, Cloudscape, DB2 Universal Database, DS4000, DS6000, DS8000, EnergyScale, Enterprise Workload Manager, General Purpose File System, , GPFS, HACMP, HACMP/6000, HASM, IBM Systems Director Active Energy Manager, iSeries, Micro-Partitioning, POWER, PowerExecutive, PowerVM, PowerVM (logo), PowerHA, Power Architecture, Power Everywhere, Power Family, POWER Hypervisor, Power Systems, Power Systems (logo), Power Systems Software, Power Systems Software (logo), POWER2, POWER3, POWER4, POWER4+, POWER5, POWER5+, POWER6, POWER7, pureScale, System i, System p, System p5, System Storage, System z, Tivoli Enterprise, TME 10, TurboCore, Workload Partitions Manager and X-Architecture are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml

The Power Architecture and Power.org wordmarks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.
UNIX is a registered trademark of The Open Group in the United States, other countries or both.
Linux is a registered trademark of Linus Torvalds in the United States, other countries or both.
Microsoft, Windows and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries or both.
Intel, Itanium, Pentium are registered trademarks and Xeon is a trademark of Intel Corporation or its subsidiaries in the United States, other countries or both.
AMD Opteron is a trademark of Advanced Micro Devices, Inc.
Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries or both.
TPC-C and TPC-H are trademarks of the Transaction Performance Processing Council (TPPC).
SPECint, SPECfp, SPECjbb, SPECweb, SPECjAppServer, SPEC OMP, SPECviewperf, SPECapc, SPEChpc, SPECjvm, SPECmail, SPECimap and SPECsfs are trademarks of the Standard Performance Evaluation Corp (SPEC).
NetBench is a registered trademark of Ziff Davis Media in the United States, other countries or both.
AltiVec is a trademark of Freescale Semiconductor, Inc.
Cell Broadband Engine is a trademark of Sony Computer Entertainment Inc.
InfiniBand, InfiniBand Trade Association and the InfiniBand design marks are trademarks and/or service marks of the InfiniBand Trade Association.
Other company, product and service names may be trademarks or service marks of others.

Revised February 9, 2010