



IBM and Next Generation Linux

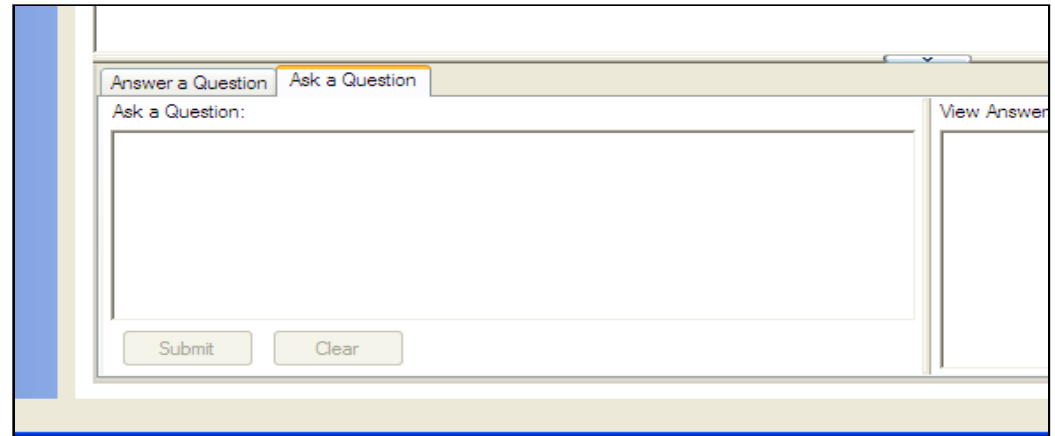
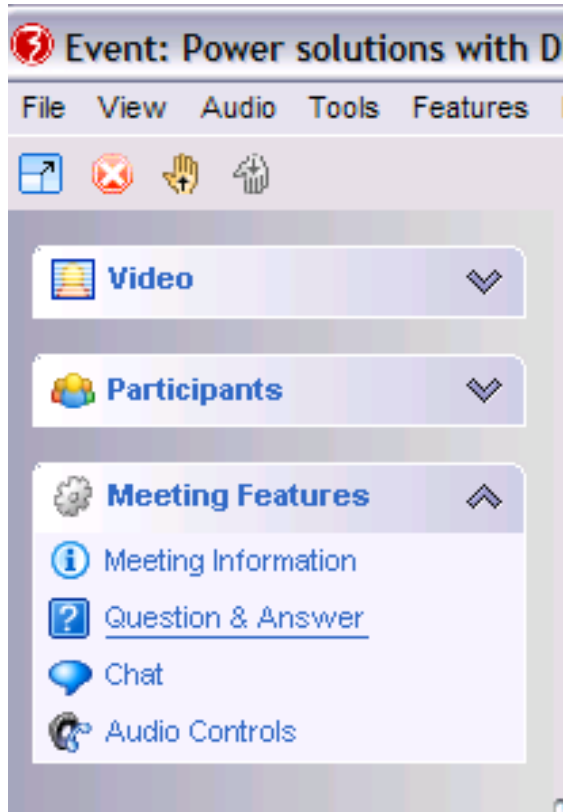
Implementing Real Time Solutions with Linux and IBM

Benjamin Vera-Tudela, Sr. Software Engineer

Audio dial-in: 1-800-528-2856
Conference ID: 1014729
International dial-in: 1-770-325-4737



How to submit questions





IBM and Next Generation Linux

Implementing Real Time Solutions with Linux and IBM

Benjamin Vera-Tudela, Sr. Software Engineer

Audio dial-in: 1-800-528-2856
Conference ID: 1014729
International dial-in: 1-770-325-4737





Inna Kuznetsova, Director, Cross-IBM Linux Strategy



Benjamin S. Veda-Tudela, IBM Software Group,
Strategy Linux Middleware

Software Pre-Sales, Linux Integration Center



Inna Kuznetsova, Director, Cross-IBM Linux Strategy



The growth and expansion of Linux as a mature, cost-effective alternative for business-critical workloads



Linux continues to enable new ways of doing business

Edge and Web Infrastructure

- Community Driven
- Internet Enabled
- Worldwide Volunteers

Application and Data Serving

- Open Industry Driven
- Open elements of IT industry join existing community
- Linux adoption in the enterprise accelerates

Business-Critical Enterprise Workloads

- Competition Driven
- Accepted as mature, open, lower-cost alternative for hosting DB, BI, ERP, CRM in business-critical environments
- Linux is a permanent presence in the datacenter

Typical Applications

- E-mail Servers
- Apache
- Lightweight database
- DHCP
- HPC

- e-Business Applications
- Application Servers
- Mission critical database
- Dynamic Business Models

- Next-generation workloads
- New business models
- Virtualization and dynamic architectures

1991 – 2004

2005 – 2006

2007 – 2009



Why Choose Linux and IBM for Real-Time Workloads?

- **Linux provides (at a lower cost)**
 - Business-critical stability
 - Enterprise-grade virtualization
 - Industry-leading performance
 - Government-ready security
- **Real-time Linux and Java solutions from IBM**
 - Lower cost, mature, open alternative to traditional proprietary real-time solutions
 - Utilize existing system administration and programming paradigms with minor variations
 - IBM collaborates closely with Novell and Red Hat to provide industry-leading real-time solutions
- **Linux is an investment in your future**
 - Fastest growing operating system in the industry drives need for skilled professionals
 - Broadly applicable skills increase your portability and relevance, while skills like real-time ensure differentiation

Raytheon





Benjamin S. Veda-Tudela, IBM Software Group, Strategy
Linux Middleware

Software Pre-Sales, Linux Integration Center



What is Real Time?

- A **“hard” real-time** application has tasks which have hard, firm deadlines for completion of their computation. Missed deadlines cause system failure (at best) or life-threatening catastrophe (at worst).
 - **Aero & Defense:** Reacting quickly and predictably to threatening situations can be a matter of life and death. Not only do weapons and tracking systems need to be fast, but also they must be completely reliable all the time, any time. IBM and its portfolio of ultra predictable real-time products provide the military and defense sectors the ability to implement reliable and predictable defense systems based on Linux and open standards.
 - **Manufacturing:** With the growing adoption of six-sigma processes, manufacturing and assembly plants must synchronize their lines to ensure that variation in their processes is significantly reduced. IBM and its portfolio of ultra predictable real-time products offer manufacturing firms the ability to automate their plants with reduced variability while taking full advantage of the benefits from solutions based on Linux and open standards.
- A **“soft” real-time** application can tolerate some missed deadlines and does not typically require hard scheduling guarantees from the OS.
 - **Finance:** News and markets change rapidly. Reacting quickly to such changes allows financial services firms to profit from the small windows of arbitrage opportunities. There are 3 areas can take advantage required increased reliability: 1) trade order management, 2) trading 3) analytics. with that arise throughout the day. IBM and its portfolio of real-time products provide financial services firms with the ability to build trade order management applications that take advantage of the flexibility of Linux and open standards.
 - **Telecom:** Companies offering communication solutions using Voice over IP (VoIP) must ensure that the exchange of messages between parties takes place at predictable intervals to prevent degradation in the quality of communications or that communications terminate abruptly. IBM and its portfolio of real-time products provide telecommunication firms with the opportunity to build applications offering predictable performance using Linux and open standards.

Where is Real-Time Important?

- Where **guaranteed** or **highly predictable** responses are needed
- Real-Time Response Times Have Broad Industry Appeal in:
 - Defense
 - Energy & Utilities
 - Financial Services
 - Gaming
 - Industrial Automation
 - Telecommunications



```
1729R U.S. FEBRUARY INDU
1728RH #DOM JONES INDUSTR
2487DH #DJIA TOPS 10000 P
INDU +42.18 VOLU 77,275
INDP 10000.95 UVOL 48,804
UTIL +.60 DVOL 20,289
TRAN -7.91 TRIN .49
```

The Power of Linux® and Java™ for Hard RT

WebSphere® Real-Time (WRT)

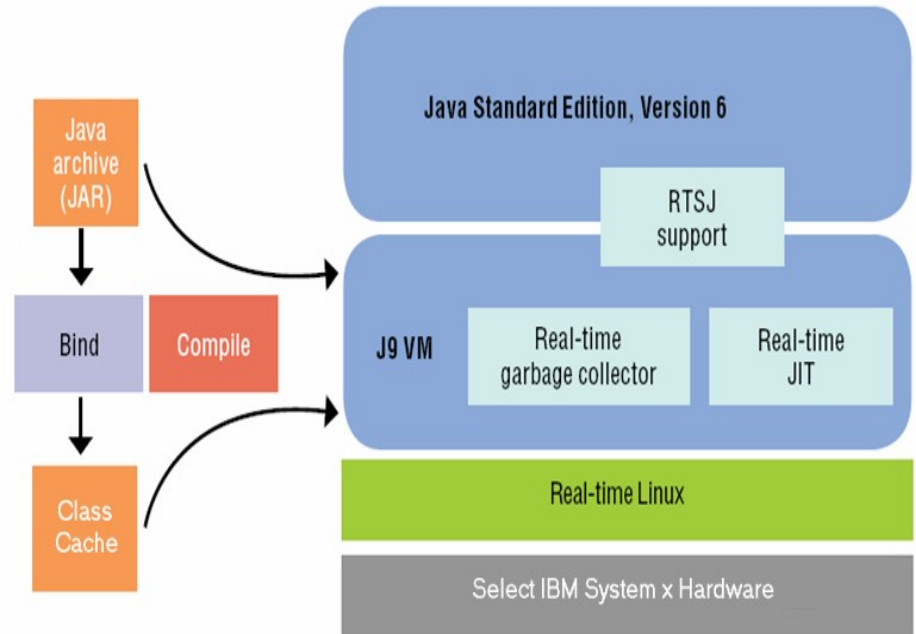
- Java 2 Standard Edition V6 with the Real-Time Specification for Java (JSR 1.0.2)
- Compilation Strategies for Real-Time (AOT, JIT)
- Metronome Garbage Collector (GC)

Real Time Linux®

- High resolution time and timers
- Fully pre-emptible kernel
- Threaded interrupt handlers
- Priority inheritance and fast user-space mutexes
- Symmetric Multiprocessing (SMP) RT scheduling

Select IBM System x® Hardware

- Enhancements to optimize real-time workloads



Real-time feature

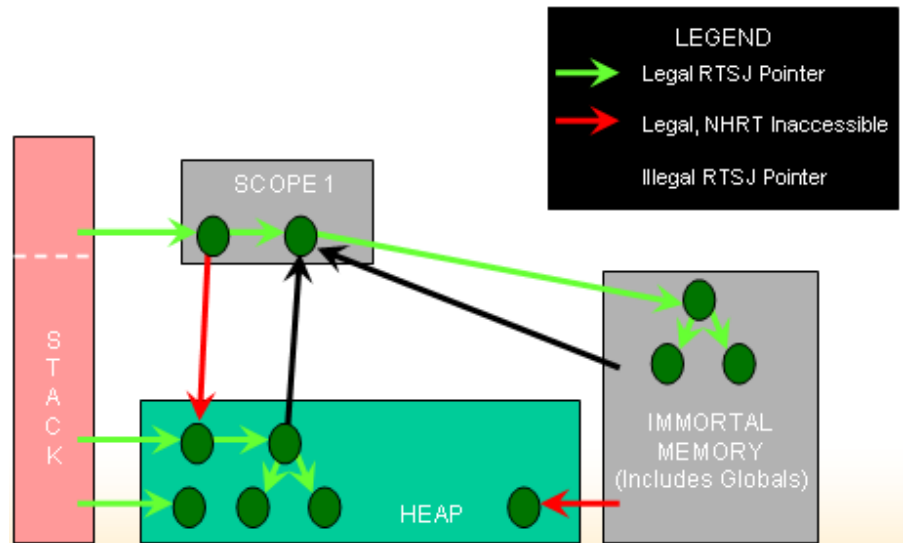
Standard J9 component



Real Time Specification for Java™ - RTSJ

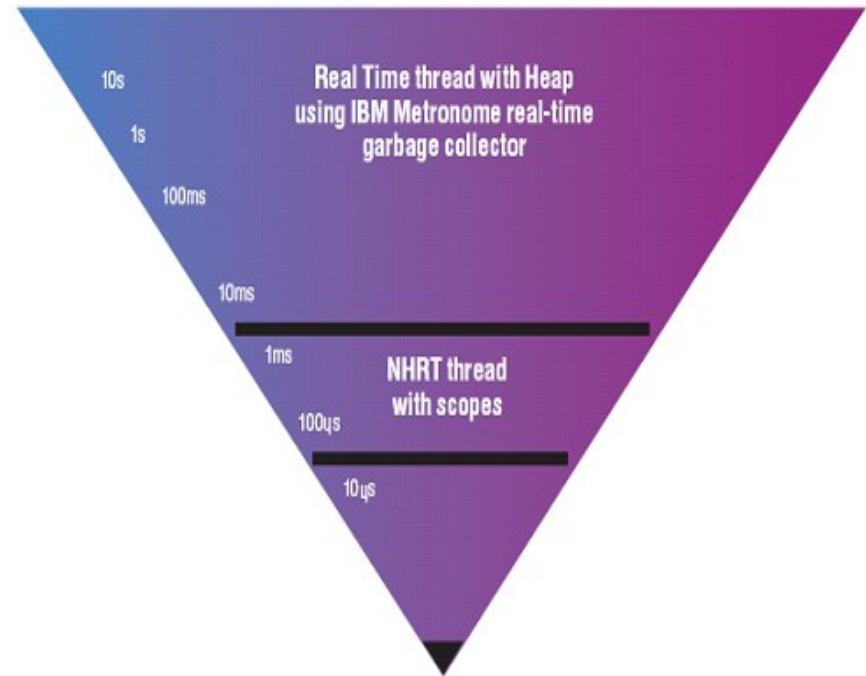
Memory Management

- Partitions memory into non-garbage collected areas that allow special threads to run at priorities higher than that of the garbage collector
- Immortal memory areas allow allocated objects to be accessible to all threads and never be collected
- Scoped memory areas can be created and destroyed under programmer control with a specified maximum size intended for allocation.



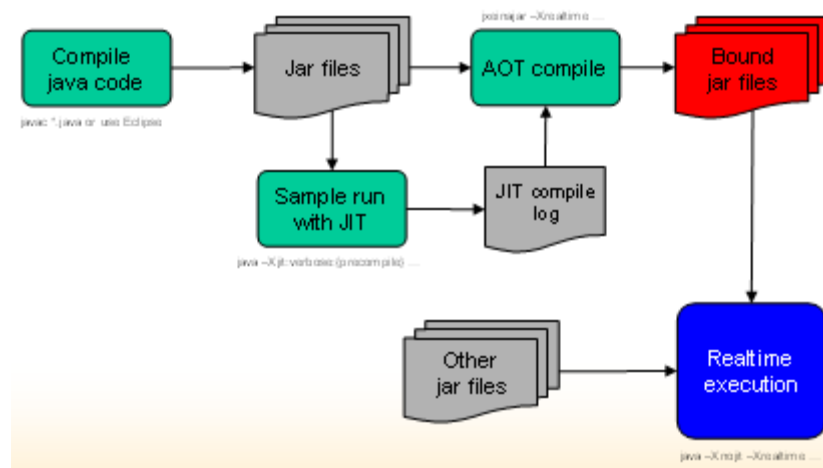
Real Time Specification for Java™ – RTSJ (cont'd)

- **Thread Management**
 - Predictable scheduler allows threads to execute based on their strict enforced priorities
 - Realtime Threads may use the heap for memory allocations and are thus subject to garbage collection interrupts
 - No-Heap Realtime Threads are not allowed to use the heap and so are uninterrupted by garbage collection cycles
 - Lowest latencies can be achieved using standard scoped memory areas with No-Heap Realtime Threads



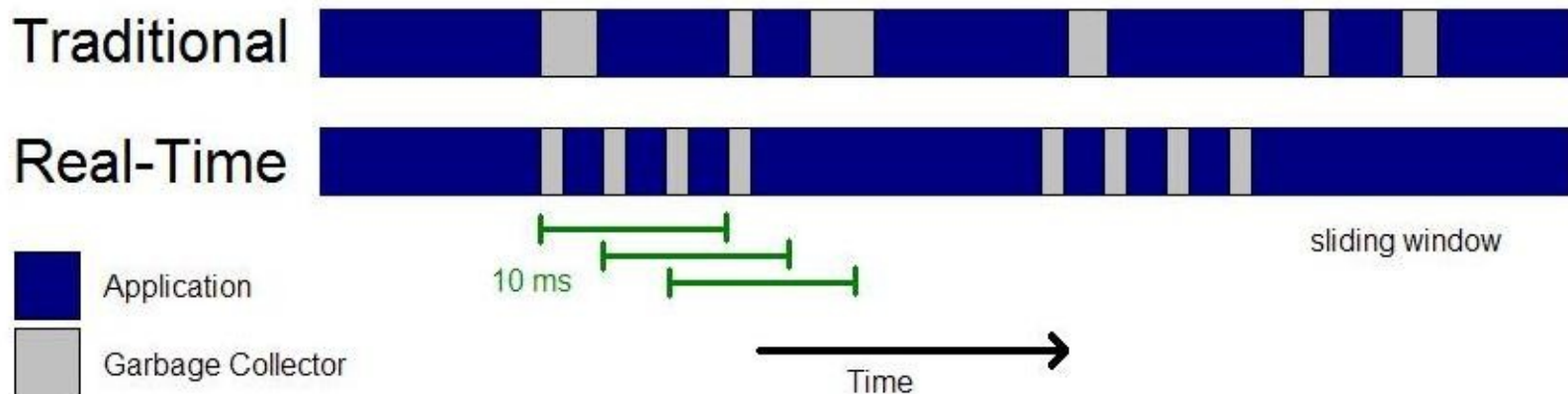
Compilation Strategies for Real Time

- Compilation in J9 is dynamic by default**
 - Code may be interpreted or compiled selectively through JIT, but JIT can run at unpredictable times and have unpredictable (although almost always positive) effect on performance
- For real-time, there are multiple choices**
 - Ahead-of-time (AOT) compilation (much better than interpreted performance)
 - User-controlled JIT (better than AOT performance, controlled via API)
 - JIT-at-low-priority (best achievable performance, runs on low priority thread)



Real Time Garbage Collection - Metronome

- **Unique technology from IBM T.J. Watson Research**
 - Garbage collection is scheduled as just another periodic real-time task
- **Enables the use of off-the-shelf Java code**
 - No need for specialized allocation schemes outside the Java heap
 - Existing Java applications can experience latencies of up to 1 ms without changes



Real Time Linux Extensions

- **High resolution time and timers**
 - High-resolution timers provide single nanosecond granularity
 - Timers can now expire within a few microseconds of each other
- **Fully pre-emptible kernel**
 - Kernel spin-locks replaced with mutexes for higher levels of pre-emption
- **Threaded interrupt handlers**
 - Allows real-time processes to be prioritized against interrupt handler processes
- **Priority inheritance and fast user-space mutexes**
 - Kernel avoids priority inversion by raising priority of locking process to that of the highest priority process waiting for the lock
 - Fast-user space mutexes (futexes) help reduce regular mutex overhead
- **SMP real-time scheduling**
 - Allows real-time processes to be scheduled against all processor run queues

Select System x IBM Hardware

- **Engineered for reliability**
 - Dual power connections
 - Thermal/cooling redundancy
 - Dual blade connections for all I/O
 - Dual switch modules
 - Dual paths through backplane
 - Dual Management Modules
- **Engineered for availability**
 - Automated failover capability via Management Module
 - Management Module monitors health of chassis components
 - Comprehensive Predictive Failure Analysis® proactively identifies many potential issues before they cause failures
 - First Failure Data Capture helps provide integrity of error reporting
 - Light Path Diagnostics for easy trouble shooting

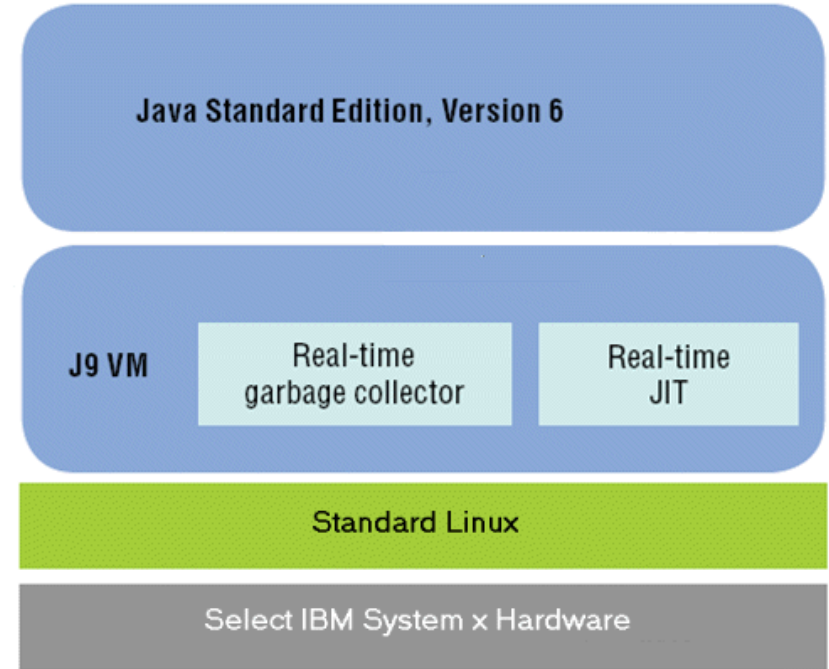


Select System x IBM Hardware – cont'd

- **System Management Interrupts (SMIs) are used to perform a variety of tasks at the CPU level**
 - Report fatal and non-fatal hardware errors
 - Perform power management (thermal throttling, power capping)
- **SMIs introduce latency which are hard to detect**
- **LS21, LS22, and HS21 XM Blades have been optimized to support real-time workloads through SMI remediation:**
 - Moves non-fatal event handling to the OS
 - BIOS handles only fatal events
 - BMC no longer requests to throttle the CPU
 - New OS service “ibm-prtm” manages entering and exiting SMI-free state

The Power of Linux® and Java™ for Soft RT

- **WebSphere® Real-Time (WRT)**
 - Java 2 Standard Edition V6
 - Compilation Strategies for Real-Time (AOT, JIT)
 - Metronome Garbage Collector (GC)
- **Standard Linux®**
 - Symmetric Multiprocessing (SMP) RT scheduling
- **IBM System x® Hardware**
 - Enhancements to optimize real-time workloads



Real-time feature Standard J9 component

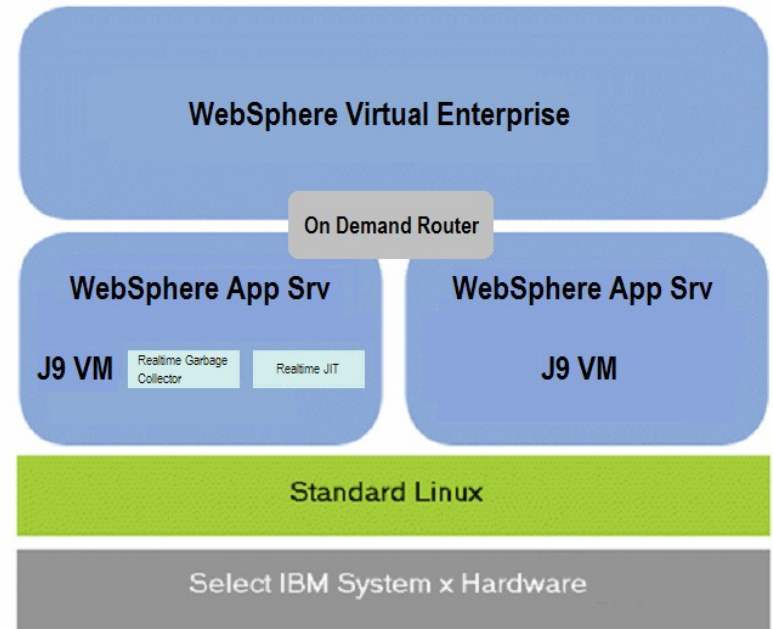


Combining WebSphere®, Linux®, and Java™

- **WebSphere® Virtual Enterprise with Soft Real-Time Java™ (SRTJ)**
 - Java 2 Enterprise Edition V6
 - Compilation Strategies for Real-Time (AOT, JIT)
 - Metronome Garbage Collector (GC)
 - On-Demand Router (ODR) component used to route prioritized transactions for improved QoS

- **Standard Linux®**
 - Symmetric Multiprocessing (SMP) scheduling

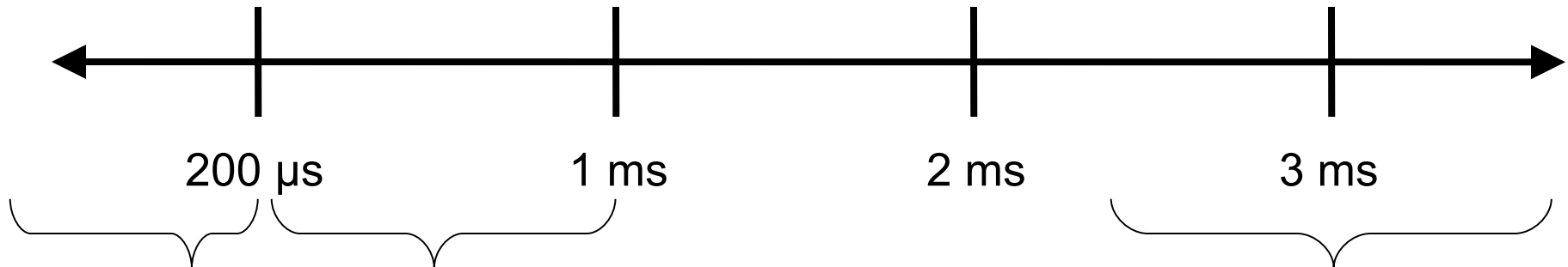
- **IBM System x® Hardware***
 - Enhancements to optimize real-time workloads



Real-time feature Standard J9 component



Deciding Between Hard and Soft Real Time



Hard Real Time

- If your goal is to **achieve** extreme predictability **all** the time
- **Latency (worst case):**
 - ≤ 1 ms using Real Time Java Threads
 - ≤ 200 μ s using NHRTs (equivalent to C/C++)

Soft Real Time

- If your goal is to **improve** predictability **most** of the time
- **Latency:** ≤ 3 ms 99.999% of the times

Competitive Real Time Offerings

Feature / Vendor	IBM	SUN	BEA
Hard Real Time	Yes	Yes	No
Soft Real Time	Yes	No	Yes
OS Supported*	Red Hat v5, SLES v10 SP2, Red Hat Enterprise MRG v1.0.1, SLERT 10 SP2	Solaris 10 Updates 4 or 5, SLERT v10 SP2, Red Hat Enterprise MRG v1.0.1	Windows 32-bit XP and Server 2003, Solaris 64-bit 9 and 10, Red Hat v3 and v4, SuSE v9

* IBM x86 architecture support available only on select IBM hardware for hard real-time workloads. non-IBM hardware support available for soft real-time workloads on select systems.

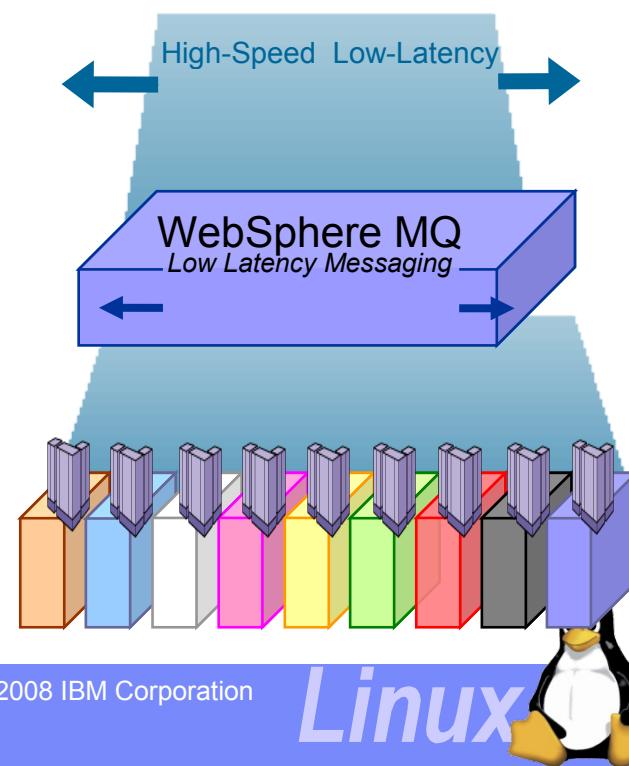
Comparing Real Time Garbage Collectors

- **Sun: RTGC runs at **lower priority** than RealtimeThreads**
 - **Pros:**
 - Provides times with higher resolution
 - Can claim sub-millisecond response times with RealtimeThreads
 - **Cons:**
 - Forces developer to avoid heap allocations from RealtimeThreads
 - Can lead to **OutOfMemory** conditions which are difficult to debug!
- **IBM: RTGC runs at **one-plus** from highest priority thread**
 - No need to worry about memory management from RealTimeThreads which improves development effort
 - Can still claim sub-millisecond response times using NoHeapRealtimeThreads

Note: Similar comparison currently not available for BEA's RTGC

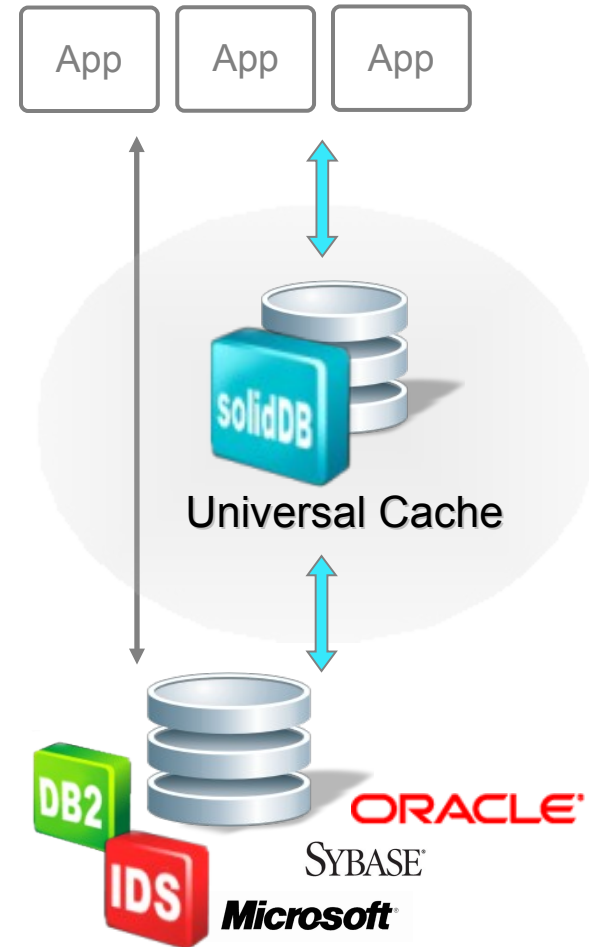
WebSphere® MQ Low Latency Messaging

- Provides a transport layer for high throughput, low latency, reliable messaging
 - **High Throughput**
 - Over 1 million messages per second
 - Fill up a 1GB Ethernet
 - **Low Latency**
 - Below 1ms end to end (faster is better)
 - **Reliability**
 - The ability to tolerate transient network disruption or application failure without compromising the integrity of the data stream.
 - Handle out of order packets in the network or kernel
 - Handle lost packets in the network or kernel
 - Handle temporary network disconnects



IBM SolidDB®

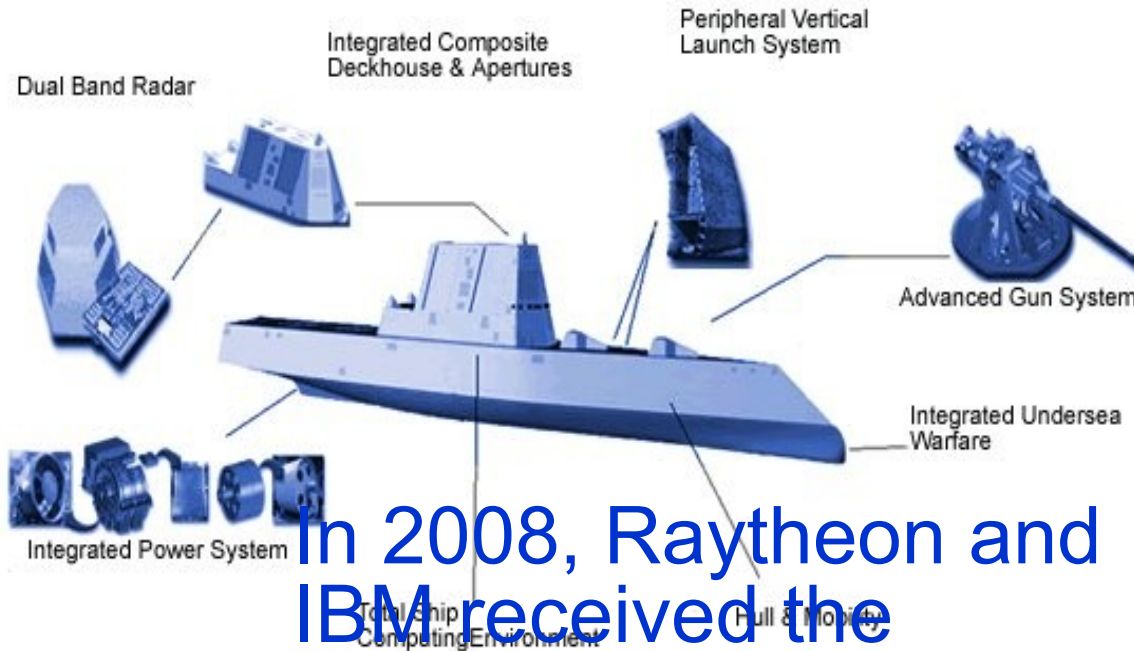
- Relational, in-memory caching technology that accelerates databases ***up to 10 times***
 - **Universal Caching**
 - Accelerates IBM DB2 for z/OS, DB2 for LUW, and IDS, Microsoft, Oracle, and Sybase
 - **Extreme Speed**
 - Tens of thousands of transactions per second
 - Response times measured in microseconds
 - Brings performance-critical data closer to the application
 - **Adaptability**
 - Adapts to different application and deployment needs
 - Provides powerful schema mapping and data transformation
 - Scales vertically and horizontally
 - **Robustness**
 - Preserves data durability
 - Delivers high availability and instant failover



Case Study

Case Study – US Navy DDG 1000

- America's next generation, multi-mission, naval destroyer, the Zumwalt systems' open architecture design enables commonality across the U.S. Navy's entire family of ships, which will pay dividends for years to come.



IBM Real Time in a Nutshell

- **J9 (WRT) ***
 - IBM-authored virtual machine used in all IBM products and on all platforms
 - Leadership performance, scalability and reliability
- **Metronome (WRT) ***
 - Real-time garbage collection with 1ms worst case pause time and providing assured minimum application CPU utilization
- **Optimizing compilation (WRT) ***
 - Static (or ahead-of-time) compilation for predictable performance
 - Dynamic (or just-in-time) compilation for best performance (predictable under controlled conditions)
- **RTSJ (WRT)**
 - **Fully compliant with the latest version of JSR 1 (1.0.2)**
 - **Includes fixed priority scheduling, priority inheritance, asynchronous event handling, scoped and immortal memory management**
- **Linux**
 - **RedHat or Novell**
 - **Updated (open source) kernel for real-time operation**
- **Select IBM xSeries Hardware**
 - **With modifications to eliminate firmware induced delays**



In Summary

- **Linux provides (at a lower cost)**
 - Business-critical stability
 - Enterprise-grade virtualization
 - Industry-leading performance
 - Government-ready security

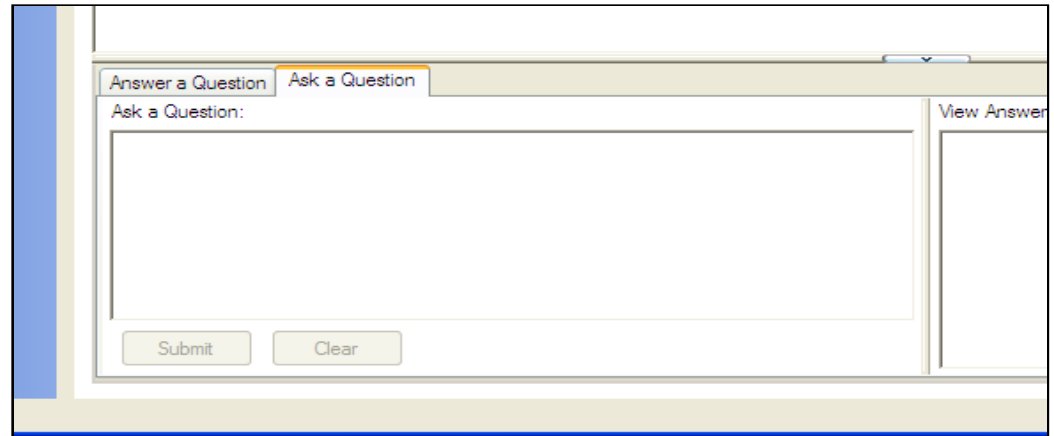
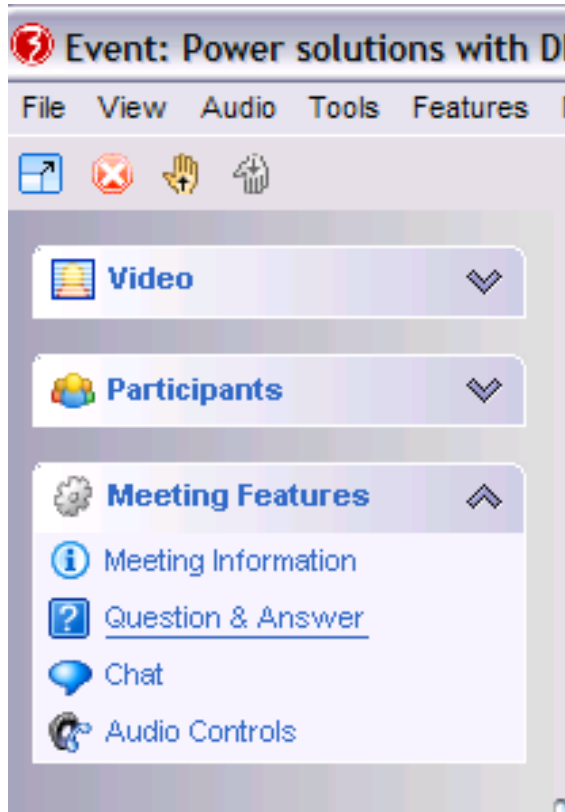
- **Real-time Linux and Java solutions from IBM**
 - Lower cost, mature, open alternative to traditional proprietary real-time solutions
 - Utilize existing system administration and programming paradigms with minor variations
 - IBM collaborates closely with Novell and Red Hat to provide industry-leading real-time solutions

- **Linux is an investment in your future**
 - Fastest growing operating system in the industry drives need for skilled professionals
 - Broadly applicable skills increase your portability and relevance, while skills like real-time ensure differentiation

Resources

- **For questions and assistance please contact:**
 - Benjamin Vera-Tudela, Linux WW Lead – bveratudela@us.ibm.com
- **For access to WebSphere Real Time, please visit:**
 - <https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-wsrt>
- **Technical documentation is available through developerWorks:**
 - Part 1: Using the Java language for real-time systems
 - Part 2: Comparing compilation techniques
 - Part 3: Threading and synchronization
 - Part 4: Real-time garbage collection
 - Part 5: Writing and deploying real-time Java applications
 - Part 6: Simplifying real-time Java development
 - Sample applications including source code
 - URL: http://www.ibm.com/developerworks/views/java/libraryview.jsp?search_by=Real+time+Java+Part
- **IBM Systems Journal:**
 - <http://www.research.ibm.com/journal/sj47-2.html>

How to submit questions



Thank You!



Trademarks and Registered Trademarks

- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- RedHat and the RedHat logo are trademarks or registered trademarks of RedHat in the United States, other countries, or both.
- SUSE and the SUSE logo are trademarks or registered trademarks of Novell in the United States, other countries, or both.
- Other company, product, or service names may be trademarks or service marks of others.