



# Experiences with deploying J2SE class Java technology in large, embedded real-time systems.

**Aonix**  
Greg Thain



## Speaker's Qualifications

- Greg Thain is a Lead Software Engineer at Aonix, Inc.
- Greg Thain has worked on embedded Java VM internals for last seven years.



## One Slide Summary of Entire Talk

Some embedded systems

need a J2SE class VM



## One Slide Summary of Entire Talk

(revised)

Some *real-time* systems

need a J2SE class VM

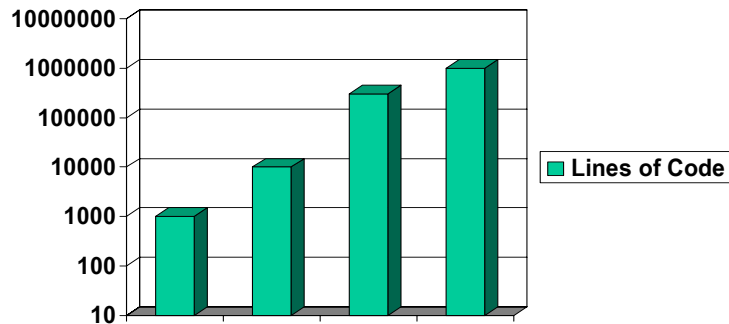


## Overview

- Embedded Systems and their problems
- J2SE Java for some embedded Systems
- Case study of J2SE Java in an Optical Switch
- Case study of J2SE Java in Industrial Control



## Some kinds of embedded Systems



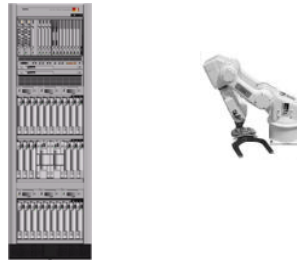


## J2ME vs J2SE

- J2ME fine for



- But what about?



## Characteristics of Big Embedded Systems

- Millions of lines of code
- Atop Real Time Operating System (RTOS)
- RTOS implements **no memory protection!**
  - Like programming for DOS
- Integration with legacy C code vital
- Hardware often not available
- Often buggy even when h/w is available



## Embedded, but not J2ME

- Need full power of the 1.3 libraries
  - Collections, java.net, java.text, RMI, JDBC
- Need full power of 1.3 tools and interfaces
  - Compiled code, JNI, JDWP, etc.
- Need platform support for RTOS and CPU
- PERC provides 1.3 environment for embedded systems



## WORA in the Embedded World

- Java code runs on three targets, unchanged
  - Desktop development platform
  - Solaris™ RTOS simulation environment
  - Target hardware

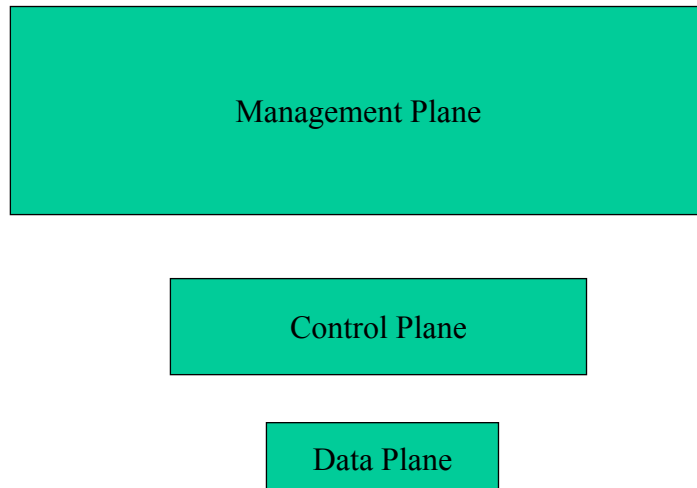


## More WORA goodness

- Few C/C++ based embedded components
- WORA allows desktop components into embedded machines
  - Poet's OO database
  - AdventNet's SNMP stack
  - Vertel's eORB CORBA broker



## Typical Network Element architecture





## Challenges of Using Java

- Platform support
- Deterministic GC issues
- Tool support
- Integration with legacy C code



## Nortel Experience Replacing S/DMS Transport

- Boeing 747 of the network world
- First to market with OC-192 (10 Gig)
- 750 KLOC of C
- Built with pSOS
- 4 years and 100s of engineers





## Goals for the Next Generation

- Protect their software investment from future RTOS and CPU changes
- Develop an architecture with the flexibility to "develop once, use anywhere"
- Eliminate software development bottleneck
- Develop a scalable, extensible, and maintainable architecture that can adapt to carrier's needs



**NORTEL**  
NETWORKS

Making Ideas a Reality

© Aonix 2004

15



## Optical Switch Hardware



- Modular, scalable, redundant architecture
- Dozens of CPUs
- Each has hundreds of Mb of RAM
- Hundreds of Mb of Flash ROM
- Terabits per second aggregate bandwidth

Making Ideas a Reality

© Aonix 2004

16



## Utilization and Development Stages

### Using PERC for:

- Embedded User Interface
- Database
- Internal Messaging
- Distributed Messaging
- Internal Events
- Remote Messaging
- Embedded web/file server/clients
- XML-based TL1 engine



**NORTEL**  
NETWORKS

Making Ideas a Reality

© Aonix 2004

17



## Product Lifecycle Benefits Realized

- Incorporation of “develop once, use anywhere”
  - Focused on control and management software within the switch
- Expected benefits
  - Software component development and integration
  - Physical and logical constraints limited previous generation
- Unexpected benefits
  - Shifting the line between hard and soft real-time functions
  - Increased use of “higher level functionality” lower in the device
  - Improvements in product release cycles: Management functionality is decoupled from “embedded part” of device



**NORTEL**  
NETWORKS

Making Ideas a Reality

© Aonix 2004

18



## Industrial Automation and Process Control

- Control Applications for factory flow
- Distributed computing  
Key:
  - Within a controller
  - Between Controllers
- Real time Important
  - 20-60 Hz cycle times



## Industrial Automation and Process Control

- RMI not a good fit
  - Creates coupling problems
- JMS works well here
- CORBA sometimes important too





## Industrial Automation and Process Control

- How Real Time?
- HotSpot's GC not good enough:
  - Still see “hiccupps”
- PERC's GC, < 1ms pauses



## What about RTSJ?

- Implementation Immature
- These applications need J2SE for library support!
  - JMS
  - CORBA
  - SNMP, etc.
- Want to use “Real Java”
- Few tools support RTSJ





## Remember!

Some embedded systems

need a J2SE class VM



## Remember!

(revised)

Some *real-time* systems

need a J2SE class VM



## Questions?

Thank you!