

Sistemas de Tiempo Real: Java

Luis Velásquez

Universidad de Costa Rica

Escuela de Ciencias de la Computación e Informática

Presentación Investigación Sistemas Operativos, 2013

Índice

- 1 Java
 - Introducción breve a Java

- 2 Real Time Java

Java RT

- 1 Java
 - Introducción breve a Java
- 2 Real Time Java

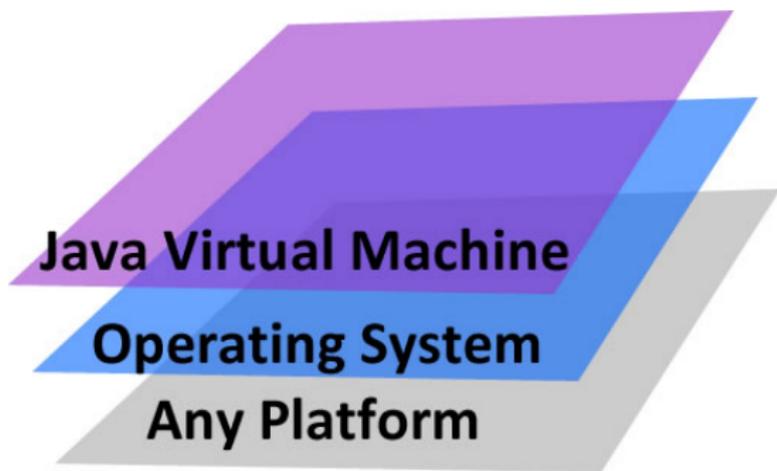
Breve recorrido de Java

- James Gosling, 1995
- Sun Microsystems
- ¿Copia de C/C++?
- Proposito general, concurrente, orientado a objetos/basado en clases



Máquina Virtual

- Compilar el código->Byte Codes
- JVM
- ¿Arquitectura que hay por debajo?



Garbage collector

- Manejo de memoria
- Referencia a objetos
- Java Runtime Environment



¿Por qué RT Java?

- 1997 se implementa la primer Real Time Library
- Problemas de la versión estándar: hilos/prioridades, recolección, sincronización

Sistemas de Tiempo Real

- Correctitud y precisión
- Soft - Hard
- Latencia

Servicios en TR de mayor utilidad para Java

- Un servicio es la acción ofrecida por parte de un sistema o usuario hacia otro de acuerdo a ciertos lineamientos establecidos por acuerdo mutuo.
- Distancia entre estos dos puede variar
- Varios hilos
- Sistemas de tiempo real suaves en software de nivel más alto, los sistemas embebidos son los que abarcan principalmente los sistemas de tiempo real duros.
- Tareas como interactividad, usabilidad, configuraciones y visualización

Real Time Java Specifications

- Interfaces o librerías
- Año 2000 se propone
- Establece protocolos para el manejo de memoria

El problema del Garbage Collector

- ¿Cuándo debe realizarse la recolección?
- 3 paradigmas:
 - JamaicaVM: cada hilo tiene su recolector
 - IBM Websphere RT: un hilo recolector tiene más prioridad que el resto
 - Sun Java RTS: un hilo recolector tiene menos prioridad

Algunos RTSJ

Implementación	Desarrollada por	Tipo
RTSJ-RI	Timesys	Referencia inicial
Mackinac	Sun Microsystems/Oracle	Commercial
JRockit	Oracle	Comercial
IBM Websphere	IBM	Comercial
Real-Time JRE	Apogee	Comercial
Aonix PERC	Atego	Comercial
Jamaica	Aicas	Comercial
aJ200	aJile	Comercial
J-Rate	Universidad de California	Open Source
OVM	Universidad de Purdue	Open Source
LJRT	Universidad de Lund	Open Source
JOP	Universidad de Viena	Open Source
Fiji	Universidad Purdue	Open Source

RTSJ Standards

- Planificación y despacho: Java mapea cada sistema operativo y núcleo
- Manejo de memoria dinámico: aún se investiga sobre como se debe manejar la memoria dinámica
- Sincronización y compartimiento de recursos: presenta algunos módulos basados en el problema de la inversión. También provee herramientas para trabajar con el model fork/join en su reciente versión JDK 7
- Manejo de eventos asincrónico: presenta una clase denominada Asynchronous Event handler (Aeh)
- Transferencia de control asincrónico: provee mecanismos seguros para abandonar los procesos de menor prioridad
- Terminación de eventos asincrónicos: similar al anterior
- Acceso a memoria física: no solamente se debe administrar y calcular los tiempos de entrada/salida

Algunos ejemplos de RTSJ con Java

- Manejo de redes
- Robótica
- Servidores web

Resumen

- Java es más eficiente a alto nivel
- El principal reto de los tiempos reales en Java es el manejo de memoria
- Java RT es una excelente herramienta con portabilidad y escalabilidad.

- Outlook
 - ¿Java sobre una máquina virtual?