

Programación de Dispositivos Móviles



“Conceptos de J2ME”

Rogelio Ferreira Escutia



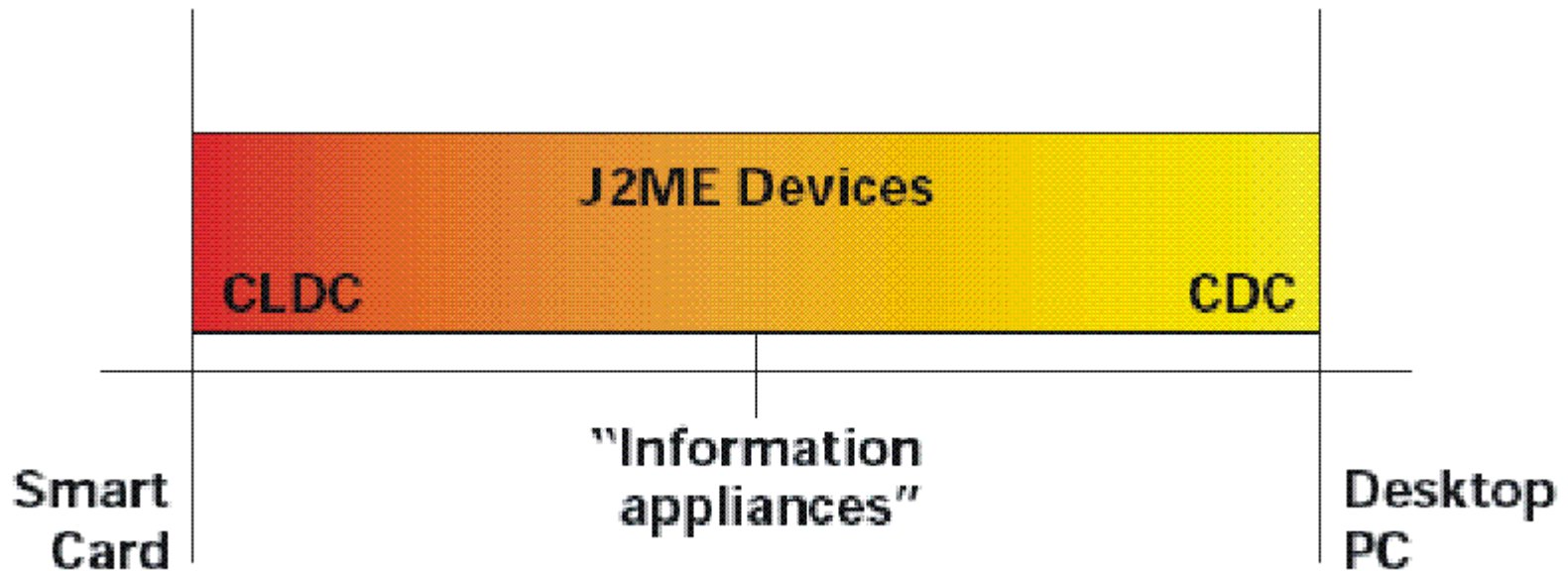
Entorno J2ME

- **Un entorno de ejecución de J2ME se compone de:**
 - a) Máquina virtual.**
 - b) Configuración.**
 - c) Perfil.**
 - d) Paquetes Opcionales.**

- Las diferentes tecnologías Java comparten un conjunto más o menos amplio de APIs básicas del lenguaje, agrupadas principalmente en los paquetes `java.lang` y `java.io`.
- J2ME contiene una mínima parte de las APIs de Java.
- J2ME usa 37 clases de la plataforma J2SE provenientes de los paquetes `java.lang`, `java.io`, `java.util`.
- J2ME a diferencia de J2SE utiliza una máquina virtual distinta de la clásica JVM denominada KVM.
- Esta KVM tiene unas restricciones que hacen que no posea todas las capacidades incluidas en la JVM. J2ME es un subconjunto de J2SE (excepto por el paquete `javax.microedition`).



Dispositivos Soportados



KVM

- **KVM es la Máquina Virtual más pequeña desarrollada por Sun.**
- **Su nombre KVM proviene de Kilobyte (haciendo referencia a la baja ocupación de memoria, entre 40Kb y 80Kb).**
- **Se trata de una implementación de Máquina Virtual reducida y especialmente orientada a dispositivos con bajas capacidades computacionales y de memoria.**
- **La KVM está escrita en lenguaje C, aproximadamente unas 24000 líneas de código.**

- Pequeña, con una carga de memoria entre los 40Kb y los 80 Kb, dependiendo de la plataforma y las opciones de compilación.
- Alta portabilidad.
- Modular.
- Lo más completa y rápida posible y sin sacrificar características para las que fue diseñada.

- No hay soporte para tipos en coma flotante, no existen los tipos `double` ni `float`.
- No existe soporte para JNI (*Java Native Interface*).
- No existen cargadores de clases (*class loaders*) definidos por el usuario, sólo existen los predefinidos.
- No se permiten los grupos de hilos o hilos *daemon*.
- No existe la finalización de instancias de clases.
- No hay referencias débiles.
- Limitada capacidad para el manejo de excepciones
- Reflexión.

CVM

- **La CVM (Compact Virtual Machine) ha sido tomada como Máquina Virtual Java de referencia para la configuración CDC y soporta las mismas características que la Máquina Virtual de J2SE.**
- **Está orientada a dispositivos electrónicos con procesadores de 32 bits de gama alta y en torno a 2Mb o más de memoria RAM.**

- Sistema de memoria avanzado.
- Tiempo de espera bajo para el recolector de basura.
- Separación completa de la VM del sistema de memoria.
- Recolector de basura modularizado.
- Portabilidad.
- Rápida sincronización.
- Ejecución de las clases Java fuera de la memoria de sólo lectura (ROM).
- Soporte nativo de hilos.
- Baja ocupación en memoria de las clases.
- Proporciona soporte e interfaces para servicios en Sistemas Operativos de Tiempo Real.
- Conversión de hilos Java a hilos nativos.
- Soporte para todas las características de Java2 v1.3 y librerías de seguridad, referencias débiles, Interfaz Nativa de Java (JNI), invocación remota de métodos (RMI), Interfaz de depuración de la Máquina Virtual (JVMDI).

Configuraciones

- Las configuraciones, que son un conjunto de clases básicas orientadas a conformar el corazón de las implementaciones para dispositivos de características específicas.
- Existen 2 configuraciones definidas en J2ME:
- **Connected Limited Device Configuration (CLDC)** enfocada a dispositivos con restricciones de procesamiento y memoria. La VM (Virtual Machine) de la configuración CLDC se denomina KVM.
- **Connected Device Configuration (CDC)** enfocada a dispositivos con más recursos. La VM (Virtual Machine) de la CDC se denomina CVM.

Java 2 Micro Edition

- CDC
- Targeted device types:
 - Screen phones
 - Set-top boxes
 - High-end PDAs

CVM

- CDLC
- Targeted device types:
 - Cell phones
 - Low to mid-range PDAs
 - Low-end set-top boxes

KVM

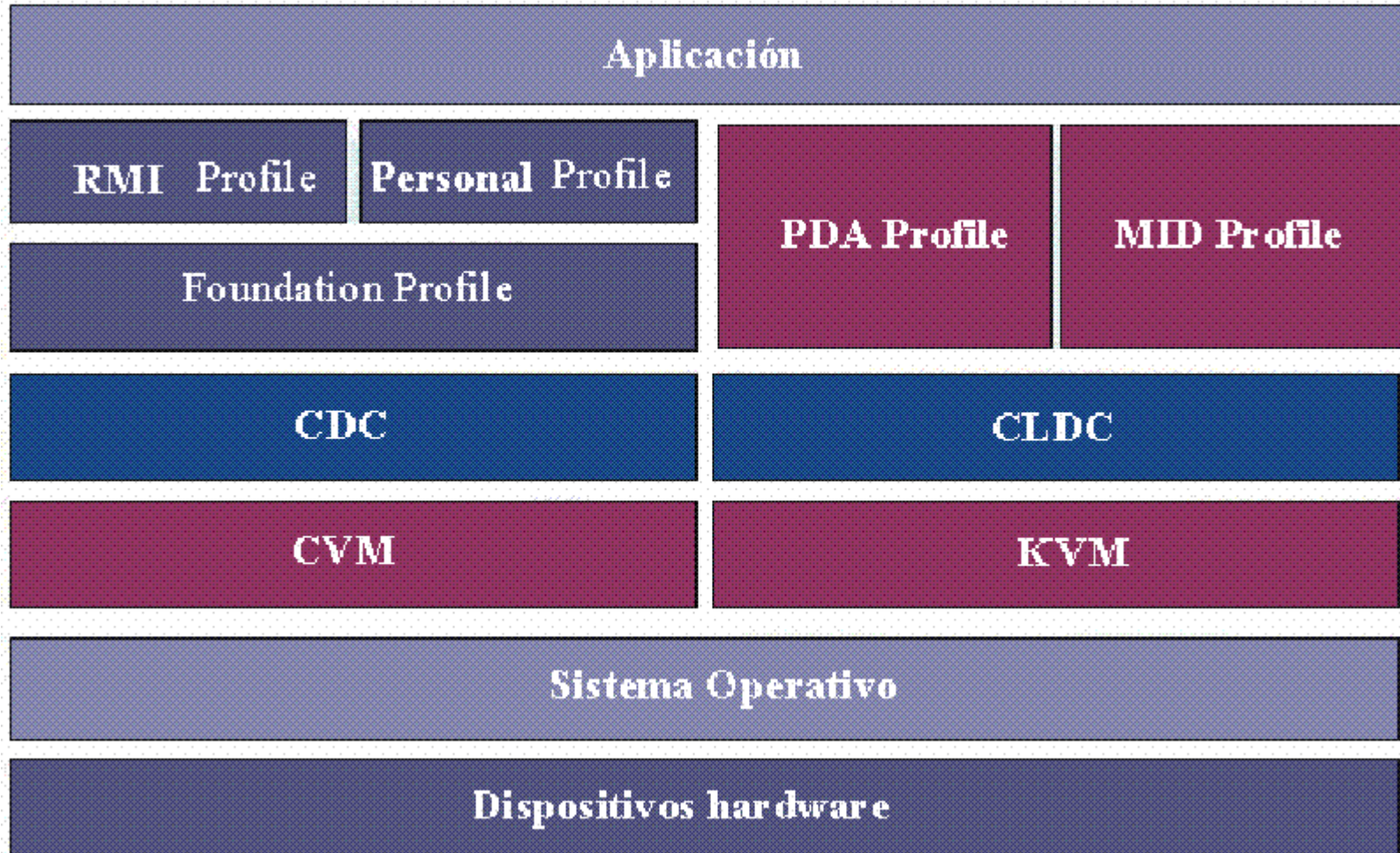


Java Programming Language

Perfiles

- Los perfiles identifican un grupo de dispositivos por la funcionalidad que proporcionan (electrodomésticos, teléfonos móviles, etc.) y el tipo de aplicaciones que se ejecutarán en ellos.
- Las librerías de la interfaz gráfica son un componente muy importante en la definición de un perfil. Aquí nos podemos encontrar grandes diferencias entre interfaces, desde el menú textual de los teléfonos móviles hasta los táctiles de los PDAs.
- El perfil establece unas APIs que definen las características de un dispositivo, mientras que la configuración hace lo propio con una familia de ellos. Esto hace que a la hora de construir una aplicación se cuente tanto con las APIs del perfil como de la configuración.

- Para la configuración CDC tenemos los siguientes perfiles:
 - *Foundation Profile.*
 - *Personal Profile.*
 - *RMI Profile.*
- Para la configuración CLDC tenemos los siguientes:
 - *PDA Profile.*
 - *Mobile Information Device Profile (MIDP).*





Rogelio Ferreira Escutia

***Instituto Tecnológico de Morelia
Departamento de Sistemas y Computación***

***Correo: rogeplus@gmail.com
rferreir@itmorelia.edu.mx***

***Página Web: <http://antares.itmorelia.edu.mx/~kaos/>
<http://www.xumarhu.net/>***

Twitter: <http://twitter.com/rogeplus>

Facebook: <http://www.facebook.com/group.php?gid=155613741139728>