Java ME Update

JCP EC Meeting May, 2012

Agenda

- Directions for Java ME evolution
- Java ME 7 Overview
- Roadmap
- Next Steps

Towards a common Java

Ensure close alignment between ME/SE

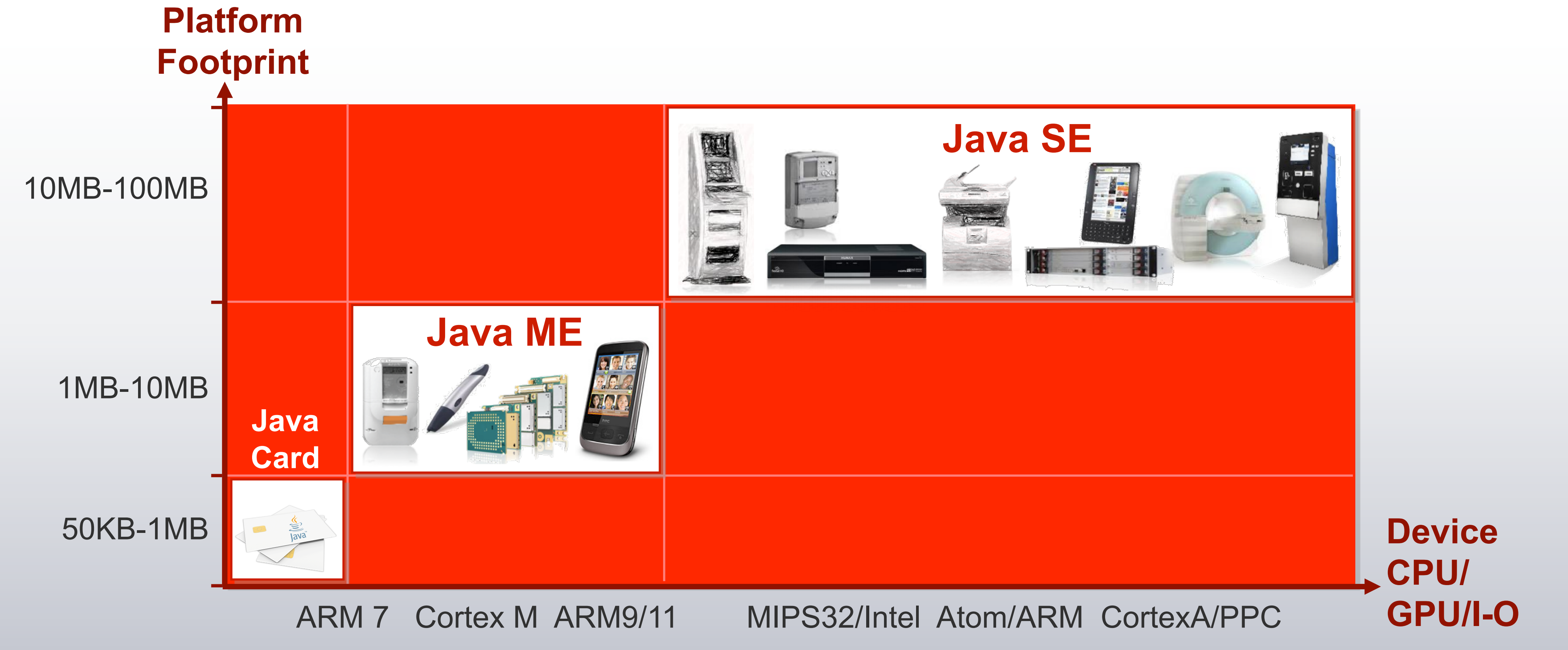
Key Principles

- ME is the “little sibling” of SE.
- CLDC is a strict subset of SE
- Any ME app/library works on SE.
- ME vs. SE is a footprint/functionality tradeoff.
- ME & SE release cycles are in sync

Benefits

- Unified development experience & community
- Align language, core APIs, development and management tools
- Enable value in SE by reusing ME APIs : Location, Messaging, Sensors, Payment, Bluetooth, ….
- Enable value in ME by reusing SE tools and management features

Java technology for Embedded Device - 2013



Platform Footprint

10MB-100MB

1MB-10MB

50KB-1MB

Java SE

Java ME

Java Card

ARM 7 Cortex M ARM9/11 MIPS32/Intel Atom/ARM CortexA/PPC

Device CPU/ GPU/I-O

Java ME7 in Phones and Embedded Devices

Agenda

- Directions for Java ME evolution
- Java ME 7 overview
- Roadmap
- Next Steps

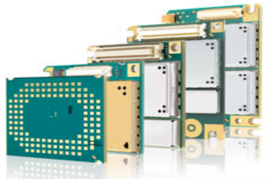
Java ME 7 Overview

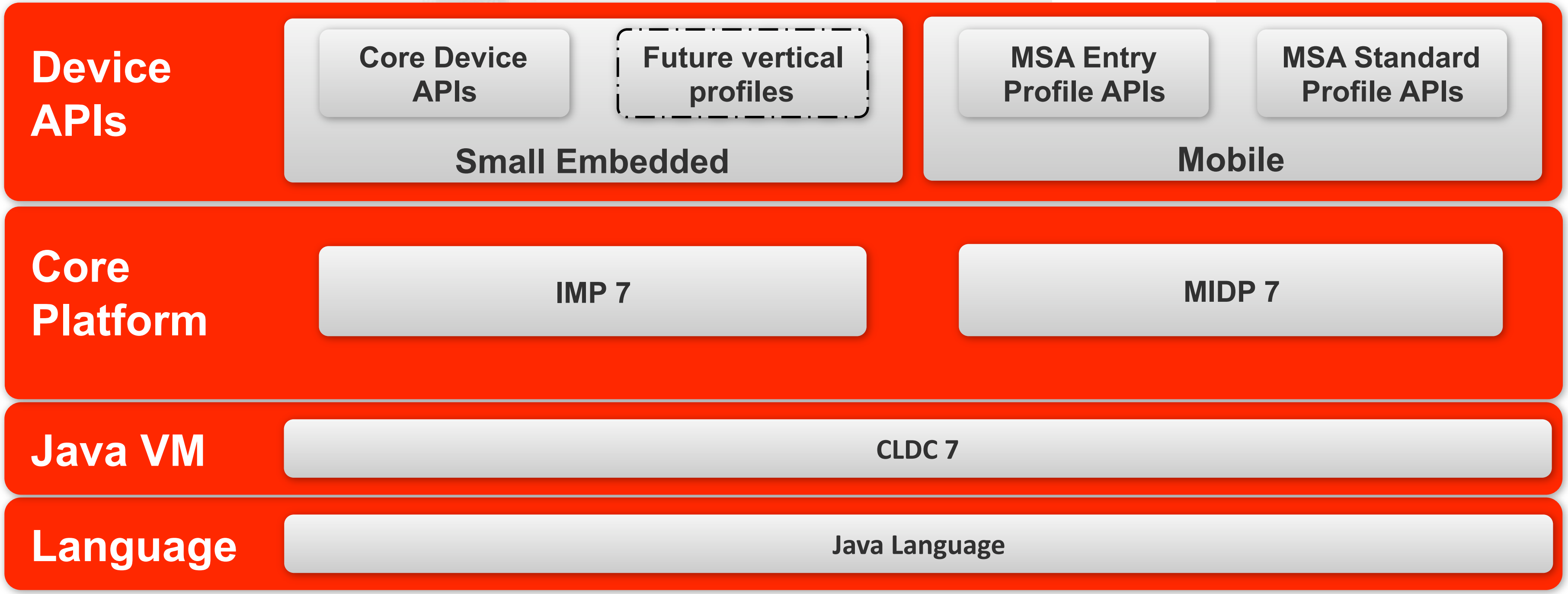
- **Release Themes**
  - Modernized mobile platform
  - Standard APIs for mobile services
  - Standard APIs for embedded

- **Target Markets**
  - Feature phones
  - Small Embedded

- **Key Features**
  - Align language/tooling with SE7
  - Redesigned MSA / MIDP to address usability issues
  - New or updated mobile APIs: AMS, JAX/RS, SATSA…
  - Dedicated APIs for small embedded

Java ME 7 Platform Architecture





Device APIs

Core Device APIs

Future vertical profiles

Small Embedded

MSA Entry Profile APIs

MSA Standard Profile APIs

Mobile

Core Platform

IMP 7

MIDP 7

Java VM

CLDC 7

Language

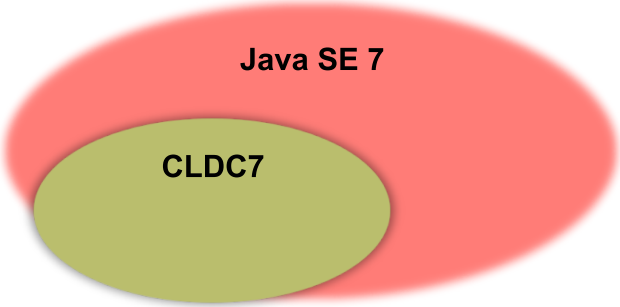
Java Language

ME7 - JSRs overview

JSRs	Description
CLDC7 **(Major Revision)**	New JVM Spec. aligned with JavaSE 7 language features
MIDP7 **(Major Revision)**	Refine MIDP to address usability issues and feature gaps New Application Management APIs
IMP7 **(Major Revision)**	Evolution of IMP for the small embedded market
MSA7 **(Major Revision)**	Umbrella JSR specifying two API profiles for mobile devices
JAX-RS **(New)**	Generic restful API framework
SATSA **(Maint. Release)**	Multiple SIM / Secure Elements interfaces
CHAPI **(Maint. Release)**	Light weight invocation API

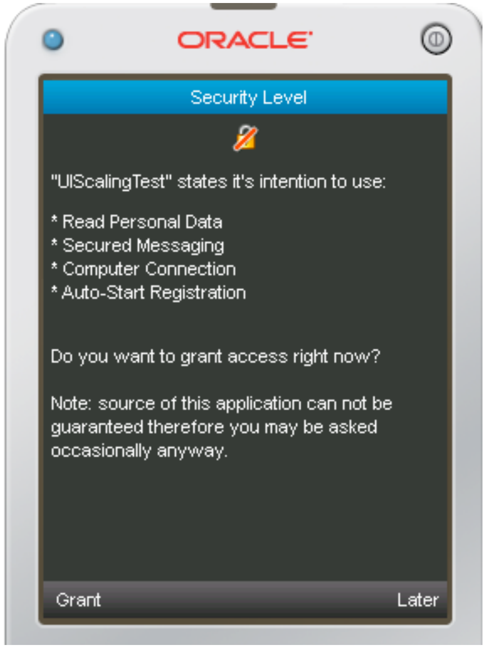
CLDC7 High-level overview

- CLDC7 is an evolutionary update for CLDC 1.1.1 to bring the VM, Java Language and libraries in alignment with Java SE 7
- Key Features
  - Synchronize with Java SE 5/6/7 Language Features into ME
  - Virtual Machine Update
  - Remain as small as possible - footprint optimizations
- Specification Requirements
  - CLDC7 to be an extended strict subset of SE7
  - Consolidated Generic Connection Framework
  - Backward binary compatibility



MIDP High-level overview

- **Description :**
  - Core ME platform specification update to address usability issues and feature gap identified through developer panels
- **Key Features:**
  - Critical MIDP 3 features
  - Simplified security model
  - Align LIBlets with SE 8 modules
  - Service Loader framework
  - Gesture API, AMS API
  - Multiple-SIM support



Security Level

"UIScalingTest" states it's intention to use:

* Read Personal Data
* Secured Messaging
* Computer Connection
* Auto-Start Registration

Do you want to grant access right now?

Note: source of this application can not be guaranteed therefore you may be asked occasionally anyway.

MIDP7: Simplifying and Improving MIDP3

- All features from MIDP3 are considered, except:
  - Application Level Access Authorization
  - RMS interchange format
- Changes to Security
  - Installation of signed unverified applications is allowed
  - Implementation is suggested to minimize the amount of user interventions related to security (security prompts)
- Shared Libraries
  - Descriptor format to be reassessed to align with module system in Java SE

MIDP7: Additions

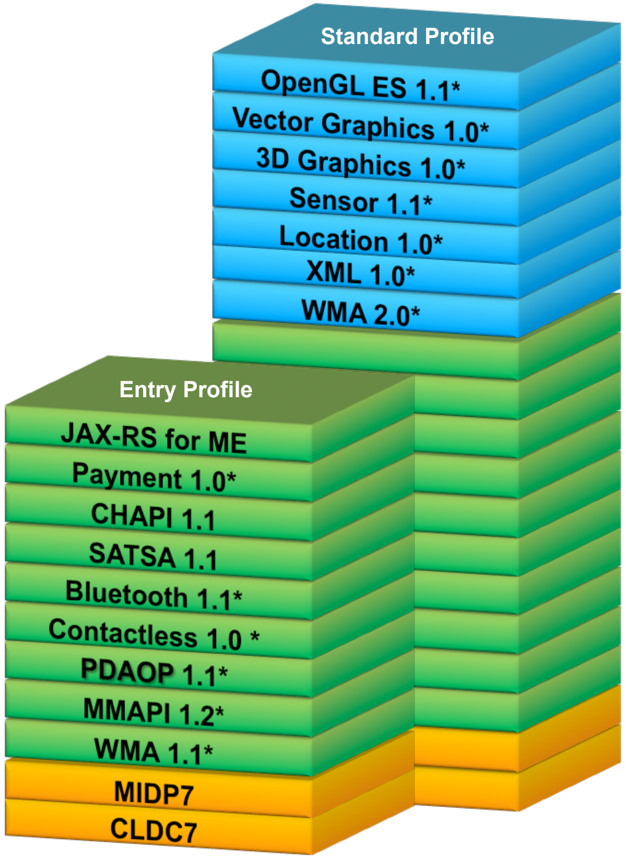
- ServiceLoader framework
  - Allows to bind two applications in form of service client-service provider
  - Virtually dynamic binding
  - SPI defined by application, not platform
  - Service provider executed in context of client application
- Connectivity API
  - SIM selection for packet data network connections
  - SIM properties for multi-SIM device
  - Connection profile selection (WiFi AP, Network I/F, etc.)

MIDP7: Updated Support

- Touch/Gesture API
  - Higher-order pointer events
- AMS API
  - Install, remove, update, start, stop, monitor,…
  - Trusted applications can use all AMS functions
  - All necessary callbacks provided
    - Status
    - Security
  - Useful to provide a customized application storefront or manage applications on the device

MSA7 High-level overview

- Evolved from MSA 2.0
- Focus on 2 profiles:
  - Revised EP focus on ARM7-2G (above) low end phone
  - Revised SP focus on ARM9/11-3G (above) feature phone
  - Drop/defer AP – not targeting high end device
- Removal of not required JSRs
  - Legacy UI and Networking related JSRs
- Adding key in-demand JSRs
- Adding new services-related JSR improvements
- MSA 7 JSRs designed to be able to work on SE



Note: * refers to unchanged JSRs

MSA Clarifications

- JSR 120/205 Clarifications for Multi-SIM support
  - Extended SMS and CBS URL connection strings to be used by application to receive/send messages via particular SIM card
  - Receive messages (SMS) from all "active" SIM cards
  - Send messages (SMS) via "preferred" SIM card
- JSR 75 Clarifications
  - JSR75 PIM implementation must support all fields of the standard types supported by the native Address Book, Calendar and Task.
  - All these fields must be accessible to Java application via predefined or extended PIM fields/attributes

JSR Maintenance Releases

- JSR 211 MR
  - Simple methods to invoke system applications
  - Email, Browser, SMS, Settings, Contact, etc.
  - Standard way of invoking installed Java ME applications
- JSR 177 MR
  - Support discovery of various static slots for secure elements present at a time
  - Notifications for insertion & removal of Secure Element
  - Retrieve dynamic features of each Secure Element

JAX – RS

New JSR to be submitted by Oracle

- Provide Client APIs for Java ME to easily access RESTful web resources / services from mass market mobile phones and small embedded devices
- Subset of JSR 339
  “JAX-RS 2.0: The Java API for RESTful Web Services”
  - javax.ws.rs.client
  - javax.ws.rs.core (subset required by javax.ws.rs.client)
  - javax.ws.rs.ext (subset required by javax.ws.rs.client)

APIs for Small Embedded

Dedicated set of APIs for sub 10MB embedded devices

- IMP7
  - Evolution of IMP2
  - App start, stop, system properties, etc
  - AMS, IO and multitasking
  - Enable diverse UI implementations
- Device Access APIs (tentative)
  - APIs to enable access from/to device interfaces and peripherals
  - For example, GPIO, MMIO, AT Commands, I2C, SPI, etc…

Java ME 8 Directions

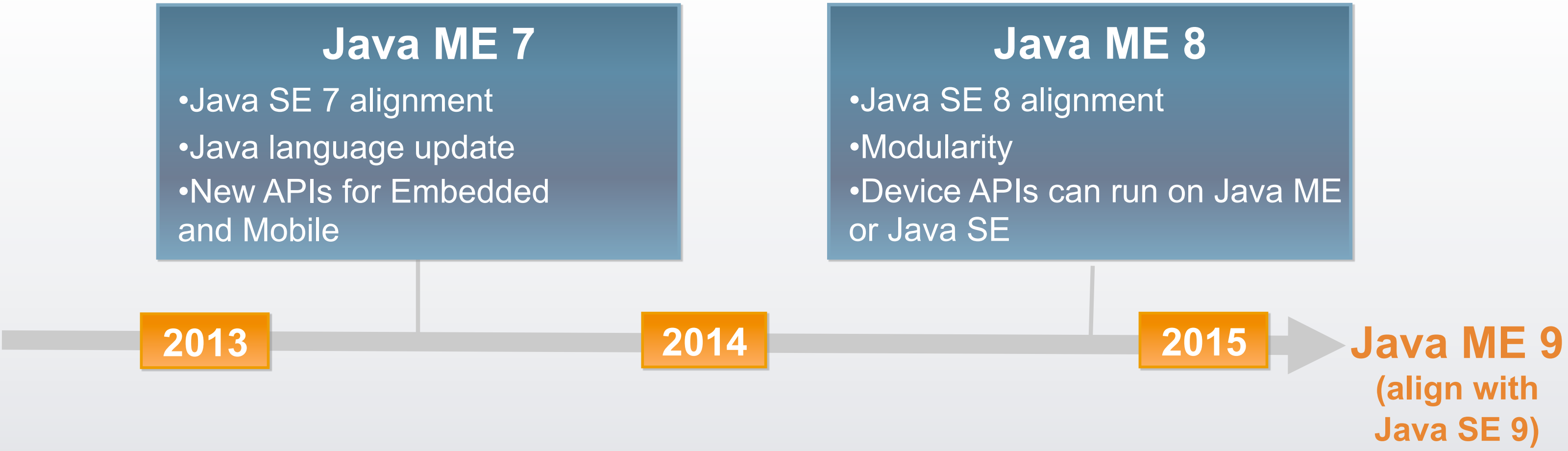
- ME / SE architectural alignment
  - Introduce SE8 Modularity to mobile/embedded
  - Support more SE 8 APIs as modules
  - Alignment with JVMS 7
- Upper stack separated from the Core VM
  - Modularize ME 7 Profiles and Optional Packages
  - Relevant Mobile and Embedded APIs can run or ME or SE
  - ME vs SE becomes a footprint/functionality tradeoff
- New APIs for embedded, tablets, smartphones
- Mobile support from SE development and management tools

Agenda

- Directions for Java ME evolution
- Java ME 7 overview
- Roadmap
- Next Steps

Java ME Roadmap

Java for Mobile and Small Embedded Devices



Java ME 7

- Java SE 7 alignment
- Java language update
- New APIs for Embedded and Mobile

Java ME 8

- Java SE 8 alignment
- Modularity
- Device APIs can run on Java ME or Java SE

2013

2014

2015

Java ME 9 (align with Java SE 9)

Agenda

- Directions for Java ME evolution
- Java ME 7 overview
- Roadmap
- Next Steps

Going forward

- Feedback on ME7 proposal
- JCP engagement
  - JSRs supporter
  - JSRs EG membership
  - Specleading JSRs

Backup slides

CLDC 7 new Java Language Features

- Assertions
- Generics
- Enhanced for Loop
- Autoboxing
- Enumerations
- Varargs
- Static imports
- Annotations

- JDK 7 features
  - Strings in switches
  - Binary integral literals and underscores in numeric literals
  - Multi-catch and more precise rethrow
  - Improved Type Inference for Generic Instance Creation (diamond)
  - Try-with-resources statement
  - Simplified Varargs Method Invocation

Library Updates for CLDC 7

- Subset of NIO Buffers
- StringBuilder and CharSequence, String formatter
- Collections update
  - Collection, List, Set, Map
  - Implementations including Hashtable and Vector
  - Iterable and Iterator
- Comparable interface
- Try with resources – Closeable and AutoCloseable
- Annotations – SuppressWarnings, Deprecated, Override

Development Tools for CLDC 7

- Standard JDK 7 tools are used for application development
- Additional tools
  - Used to target the application for CLDC platforms
  - Compiler → Preverifier
  - Integrated with ME SDK and IDEs