Mobile Service Architecture Initiative: MSA Hitting the Market Soon

Kay Glahn

Consultant Mobile Service
Architecture, Vodafone
http://www.vodafone.com

Asko Komsi

Director
Nokia
http://www.nokia.com

TS-4936

Goal of Our Talk
What You Will Gain

Get a glimpse of the near future of the Java™ Platform, Micro Edition (Java ME)

Learn about the status of MSA (JSR 248) and MSA Advanced (JSR 249)

Get prepared for the new features available for Java ME platform-based applications

Agenda

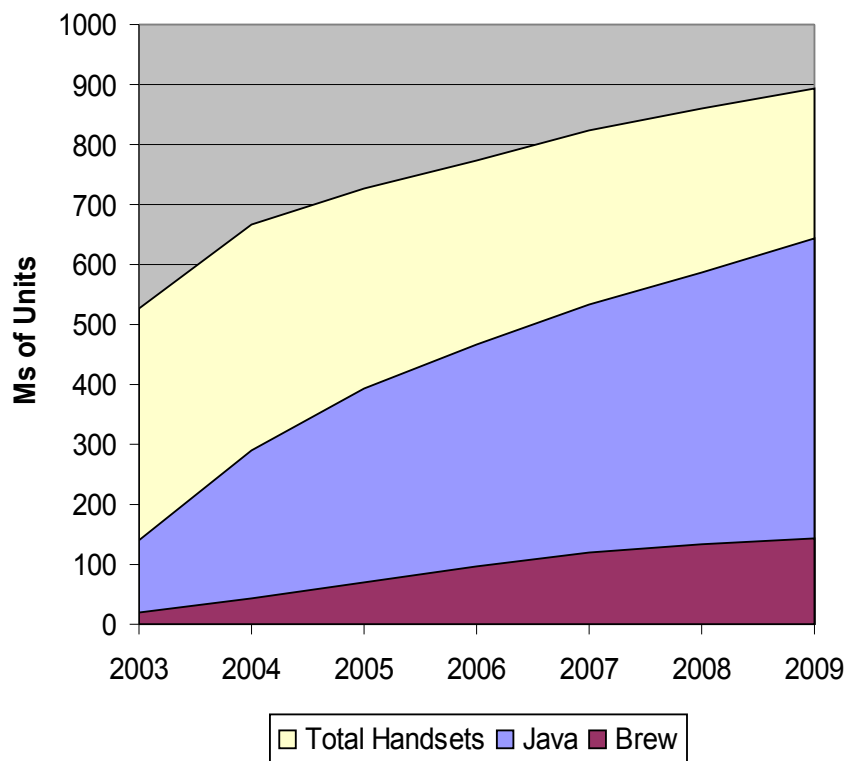**MSAInitiative**

MSA (JSR 248)

MSA Advanced
(JSR 249)

Summary

Background

Better Than Ever

CDC and Beyond

Key Takeaways

Java Technology Deployment Globally

Handsets Annually

Ms of Units

1000
900
800
700
600
500
400
300
200
100
0

2003 2004 2005 2006 2007 2008 2009

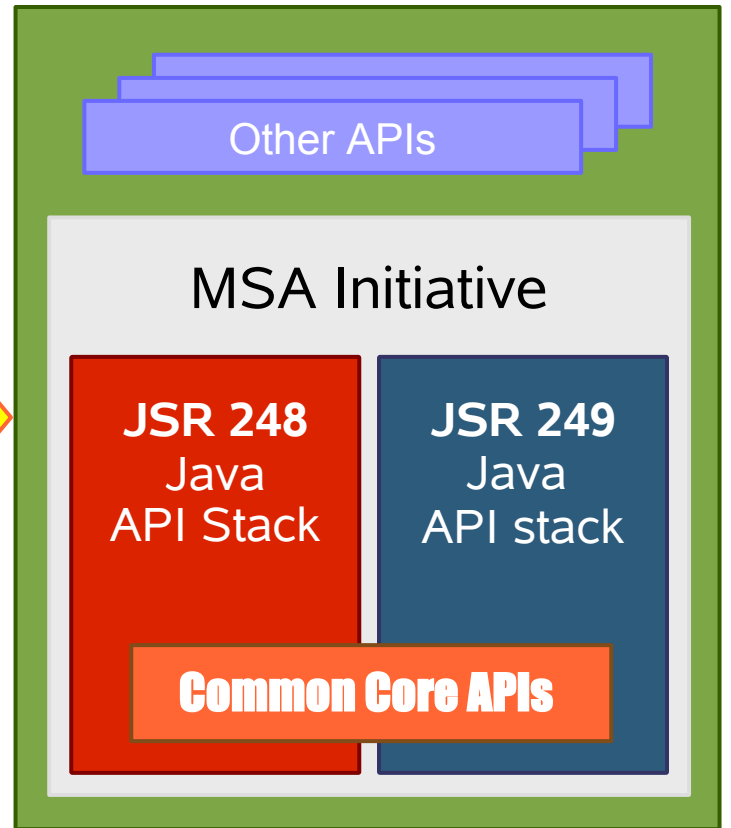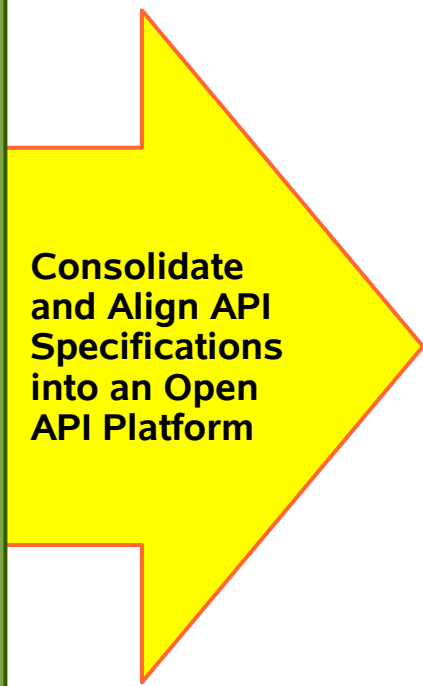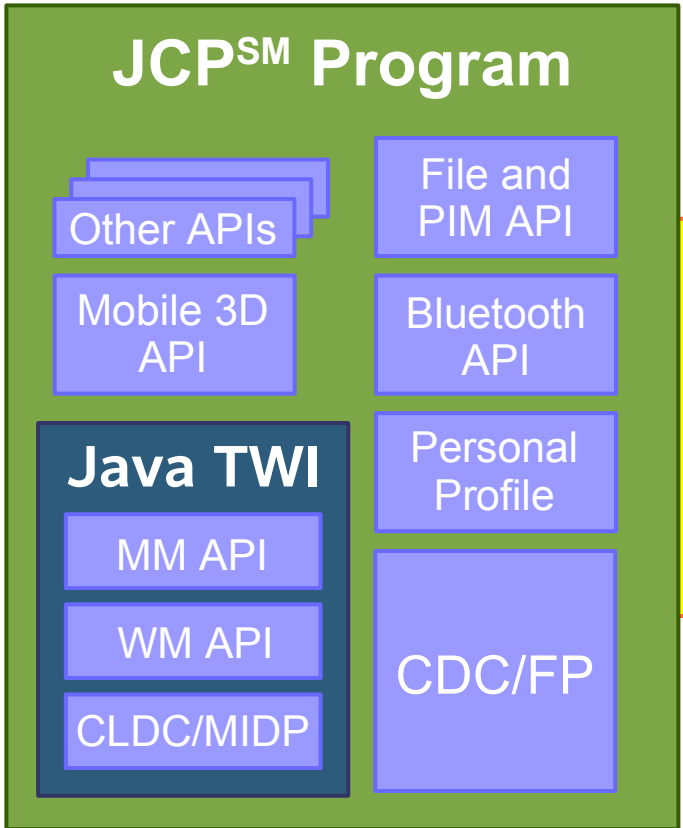□ Total Handsets ■ Java ■ Brew

Source: Nokia estimates

- 190+ operators deployed Java technology*

- 635+ Java technology-enabled handset models by 32 vendors on the market**

- 708+ million Java technology-enabled handsets on the market***

- 750,000 Java technology developers focusing on mobile***

- 45,000+ mobile Java technology-based applications on the market****

- 1 million mobile Java technology developer toolkits downloaded*

- ~23 million mobile Java technology downloads globally/month*

Sources: * Nokia, November 2005; **Ovum, June 2005; ***Sun Microsystems, June 2005; **** Informa Telecoms and Media, June 2005.

MSA Initiative—Simplifying the Java Technology API Landscape

2003

JCP^SM Program

Other APIs

Mobile 3D API

File and PIM API

Bluetooth API

Java TWI

MM API

WM API

CLDC/MIDP

Personal Profile

CDC/FP

Consolidate and Align API Specifications into an Open API Platform

2006 →

Other APIs

MSA Initiative

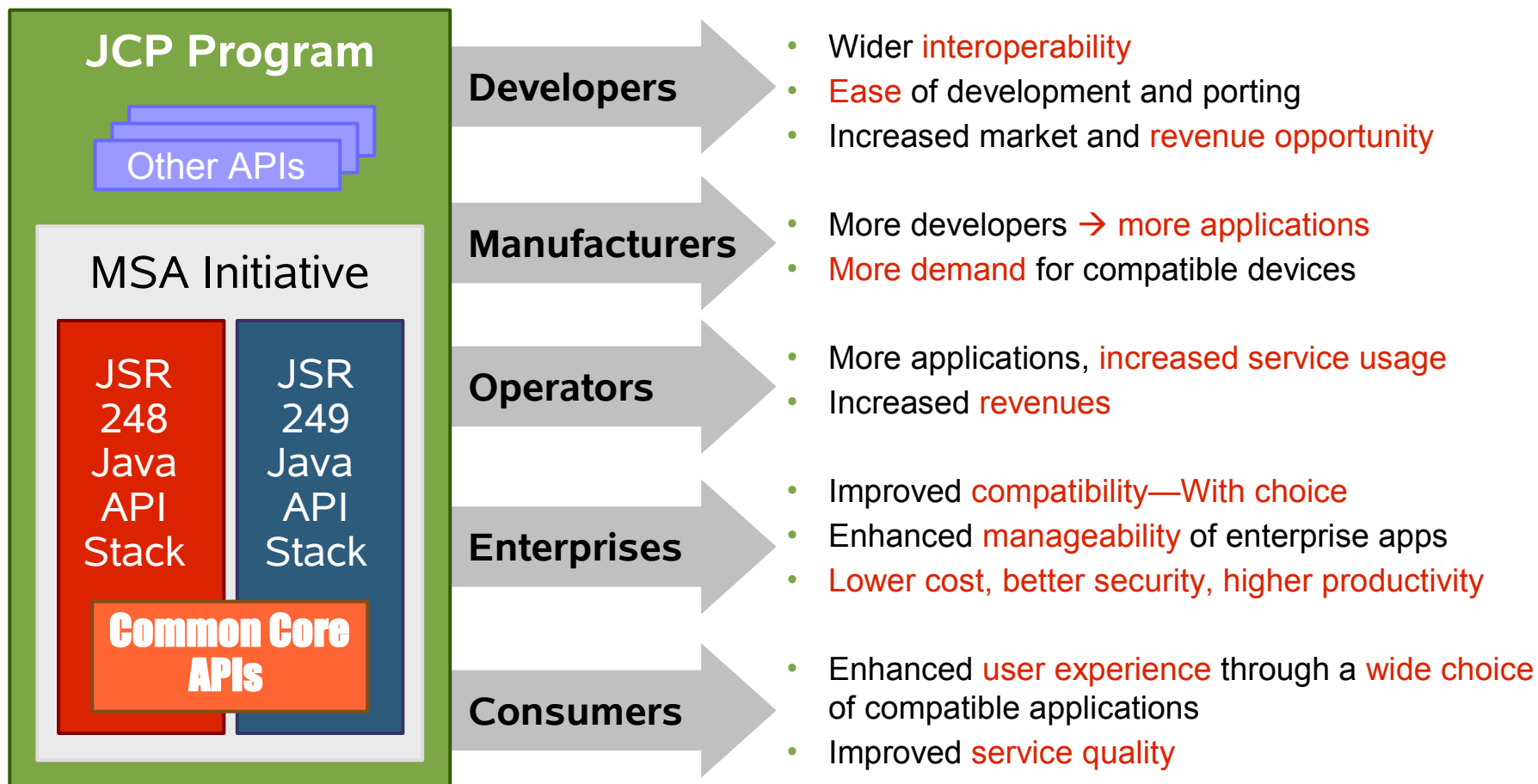JSR 248 Java API Stack

JSR 249 Java API stack

Common Core APIs

MSA Initiative—Four Steps Towards a Platform

- Selecting JSRs to form the MSA platform:
  - Decide on necessary functionality, time-to-market, overall resource requirements, end-to-end availability, etc.

- Providing additional clarifications to reduce ambiguity and fragmentation:
  - Some JSRs are targeting a wider scope by providing options
  - Interaction of JSRs is not always specified
  - Some early implementations showed room for improvement

- Specifying additional requirements:
  - Java TWI Requirements, Security, Hardware, etc.

- Providing compliancy testing tools:
  - TCK/RI

MSA Initiative—Industry Benefits

JCP Program

Other APIs

MSA Initiative

JSR 248 Java API Stack	JSR 249 Java API Stack

Common Core APIs

Developers
- Wider interoperability
- Ease of development and porting
- Increased market and revenue opportunity

Manufacturers
- More developers → more applications
- More demand for compatible devices

Operators
- More applications, increased service usage
- Increased revenues

Enterprises
- Improved compatibility—With choice
- Enhanced manageability of enterprise apps
- Lower cost, better security, higher productivity

Consumers
- Enhanced user experience through a wide choice of compatible applications
- Improved service quality

Agenda

MSA Initiative

MSA (JSR 248)

MSA Advanced
(JSR 249)

Summary

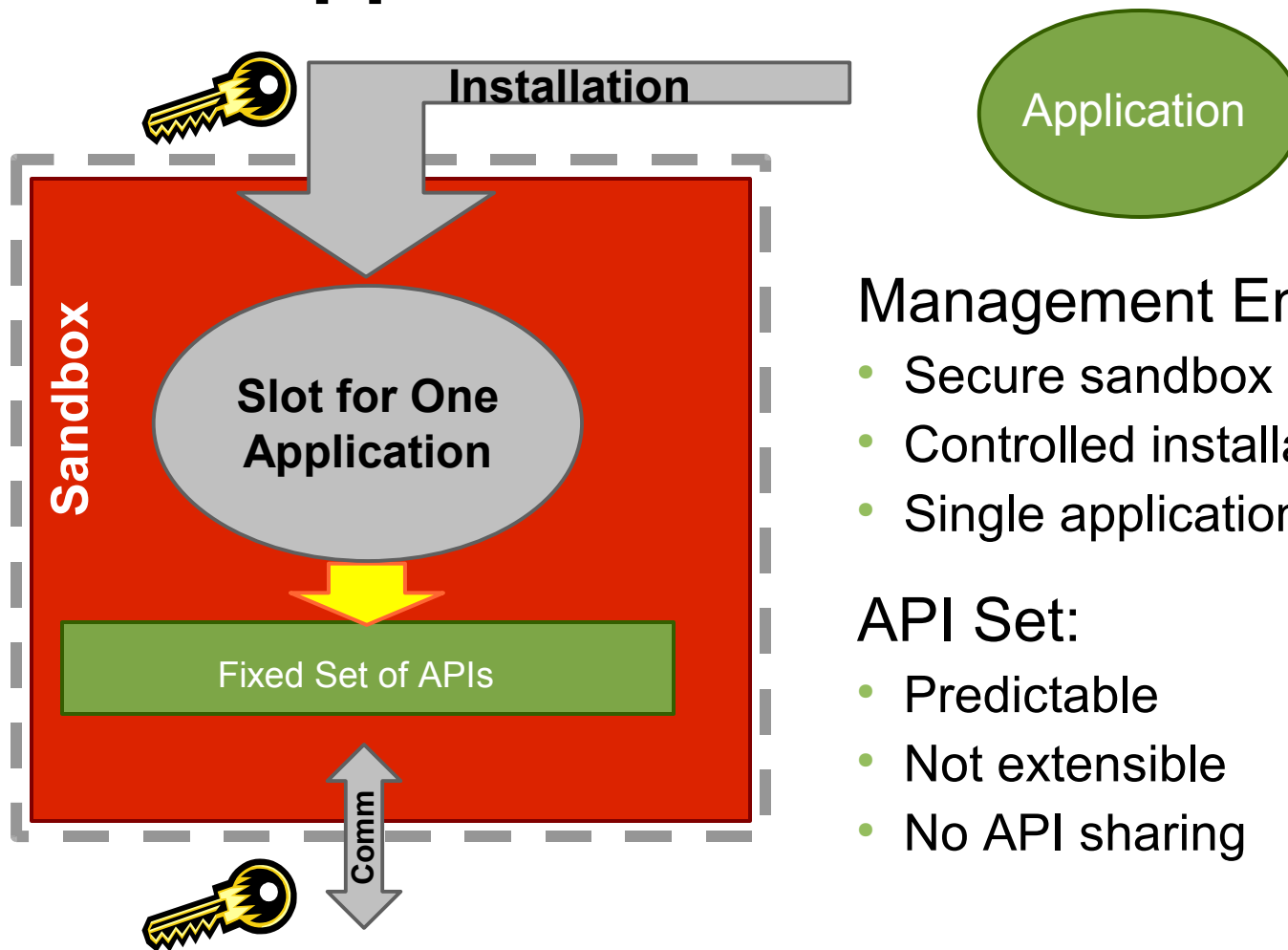Background

Better Than Ever

CDC and Beyond

Key Takeaways

java.sun.com/javaone/sf

Mobile Service Architecture (MSA—JSR 248)

- Device focus: Mass-market consumer devices and applications
    - Somewhat limited device resources
    - Simple business model for application purchases and downloads
    - No component based dynamic upgrade mechanism
- Architectural focus:
    - Predefined static stack that supports key technologies
    - Backwards compatibility with Java TWI

MSA Application Execution Model

Installation

Application

Sandbox

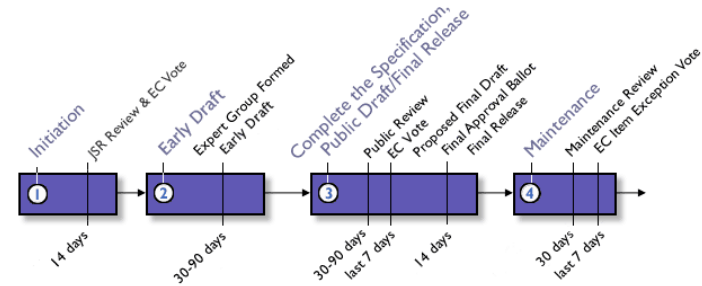**Slot for One Application**

Fixed Set of APIs

Comm

Management Environment:

- Secure sandbox
- Controlled installation
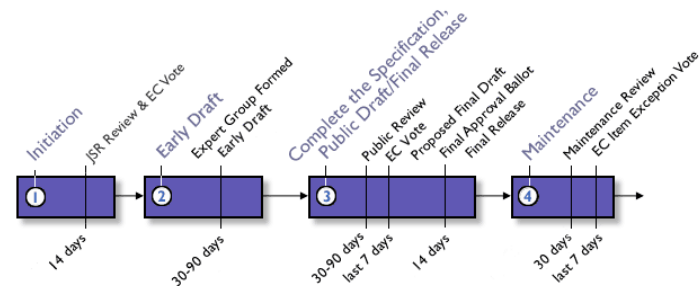- Single application isolation model

API Set:

- Predictable
- Not extensible
- No API sharing

MSA—Key Milestones

- **Early draft review**—18th April, 2005:
  - 8 months of work
  - No security or hardware requirements
  - 17 Component JSRs + ~100 clarifications
  - Review: 126 new contributions

- **First public draft**—24th September, 2005:
  - 5 months of work
  - Full specification including all parts
  - 17 component JSRs + ~75 clarifications
  - Review: 74 new contributions

MSA—Key Milestones

- Key issues to be solved:
  - Provide an intermediate step between Java TWI and MSA
  - Support flexible security policy definitions
  - Support JSR 248 implementations on CDC, too
- Proposed final draft—May, 2006:
  - 6 months of work
  - Full Specification that defines two platforms:
    - MSA Subset: 8 component JSRs with clarifications
    - MSA: 16 component JSRs with clarifications
  - New structure + summary tables for readability
  - TCK/RI work continues → FAB in August, 2006

MSA and MSA Subset

MSA:

JSR 238 (Internationalization)
JSR 234 (Multimedia Supplements)
JSR 229 (Payment)
JSR 211 (Content Handler)
JSR 180 (SIP)
JSR 179 (Location) *
JSR 177 (Security and Trust) *
JSR 172 (Web Services)
JSR 226 (Vector Graphics)
JSR 184 (3D Graphics)
JSR 82 (Bluetooth) *
JSR 75 (File and PIM)
JSR 205 (Messaging)
JSR 135 (Mobile Media)
JSR 118 (MIDP)
JSR 139 (CLDC) / CDC

MSA Subset:

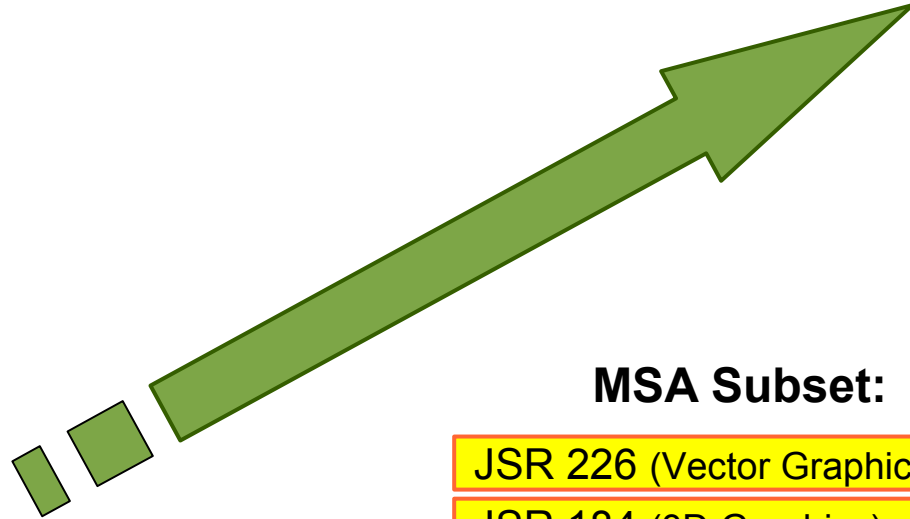| |
|---|
| JSR 226 (Vector Graphics) |
| JSR 184 (3D Graphics) |
| JSR 82 (Bluetooth) * |
| JSR 75 (File and PIM) |
| JSR 205 (Messaging) |
| JSR 135 (Mobile Media) |
| JSR 118 (MIDP) |
| JSR 139 (CLDC) / CDC |

Java TWI:

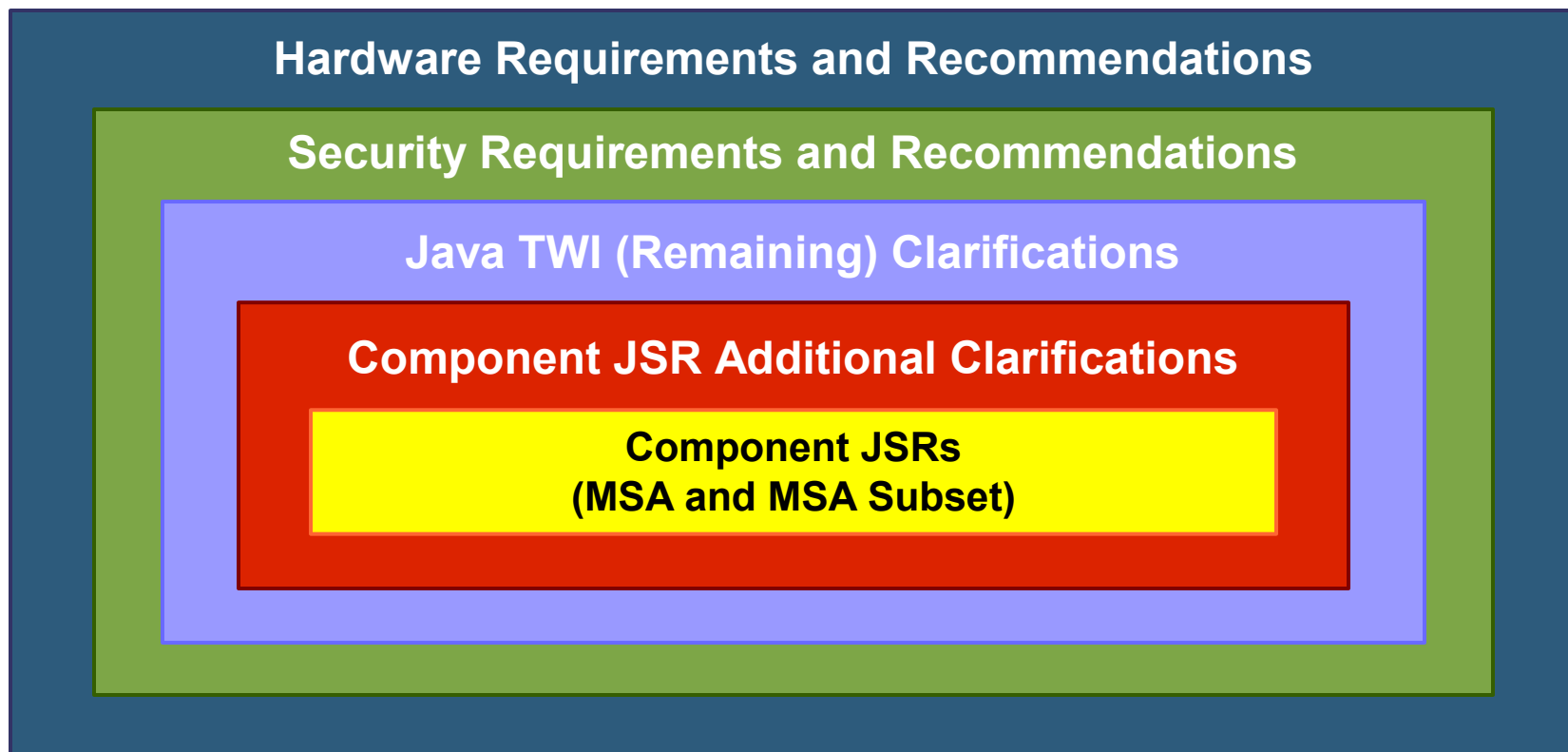| |
|---|
| JSR 120 (Messaging) |
| JSR 135 (Mobile Media) |
| JSR 118 (MIDP) |
| JSR 139 (CLDC) |

* These JSRs or parts of them are conditionally mandatory

MSA Specification

Hardware Requirements and Recommendations

Security Requirements and Recommendations

Java TWI (Remaining) Clarifications

Component JSR Additional Clarifications

**Component JSRs
(MSA and MSA Subset)**

- Layers of requirements and recommendations

Examples

java.sun.com/javaone/sf

Development of Java Technology-Based Games

- JSR 184 (3D Graphics)
  - 3D world creation and manipulation
- JSR 135 (Mobile Media)
  - Sounds
- JSR 82 (Bluetooth)
  - P2P gaming
- JSR 180 (SIP)
  - P2P over the network
- JSR 229 (Payment)
  - Payment of new levels

Development of Mapping Applications

- **JSR 226 (Vector Graphics)**
  - Map data visualization
- **JSR 179 (Location)**
  - Finding current position
- **JSR 172 (Web Services)**
  - Requesting business addresses
- **JSR 75 (File and PIM)**
  - Mapping addresses
- **JSR 238 (Internationalization)**
  - Creating a globally available version

Development of Info Applications

- JSR 172 (Web Services)
  - Access of data and parsing
- JSR 205 (Messaging)
  - Sending info to friends
- JSR 211 (Content Handler)
  - Viewing URLs using the browser
- JSR 75 (File and PIM)
  - Saving important articles

Development of Multimedia Applications

- JSR 234 (Multimedia)
  - Access to camera
- JSR 205 (Messaging)
  - Sending media to friends
- JSR 179 (Location)
  - Location metadata for media
- JSR 75 (File and PIM)
  - Saving important media

MSA Future

- Final specification: August 2006
  - JSR 248 specification
  - TCK and RI

- Products expected 4Q2006/1Q2007

- Future roadmap:
  - New specification out in ~2 years time (depends on market requirements)
  - There are roughly ~20 new JSRs being standardized
    - MIDP3, UI customization, telephony, sensor, device management, etc.

java.sun.com/javaone/sf

Agenda

MSA Initiative — Background

MSA (JSR 248) — Better Than Ever

MSA Advanced (JSR 249) — **CDC and Beyond**

Summary — Key Takeaways

java.sun.com/javaone/sf
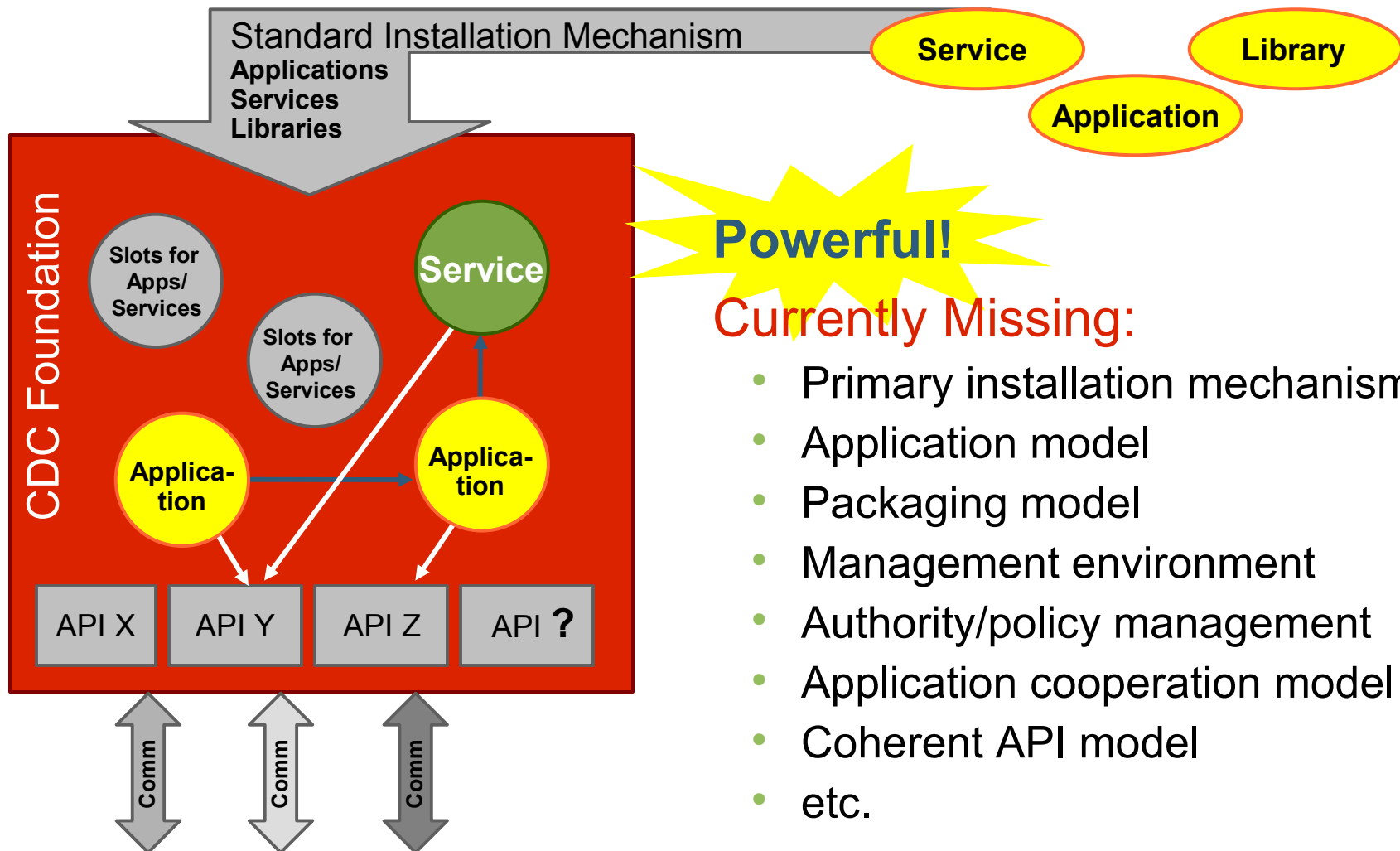
Mobile Service Architecture Advanced (MSAA—JSR 249)

- **Device focus**: Advanced mobile devices
  - More powerful, manageable handsets
  - Buy, use, upgrade and replace usage model
  - Various business models to be supported including applications, services and middleware

- **Architectural focus**:
  - Flexible, core technology foundation that supports static and dynamic extensions
  - Transparent support of MSA applications

MSA Advanced: Features

Standard Installation Mechanism
Applications
Services
Libraries

Service

Library

Application

CDC Foundation

Slots for Apps/ Services

Slots for Apps/ Services

Service

Applica-tion

Applica-tion

API X	API Y	API Z	API **?**

Comm

Comm

Comm

Powerful!

Currently Missing:

- Primary installation mechanism
- Application model
- Packaging model
- Management environment
- Authority/policy management
- Application cooperation model
- Coherent API model
- etc.

Status of MSA Advanced

- MSA work has focused on finalizing JSR 248
- JSR 249: Analysis of use cases and requirements
- MSA added CDC support to JSR 248
  - Provides an initial standard platform for CDC-based handsets
  - Is the first step towards JSR 249
- Future:
  - Finalize use case and requirement phase
  - Select appropriate technologies
  - Early draft review in 4Q2006

Agenda

MSA Initiative

MSA (JSR 248)

MSA Advanced (JSR 249)

Summary

Background

Better Than Ever

CDC and Beyond

Key Takeaways

Summary

- MSA expert group:
  - Has put out three versions of the MSA specification
  - Has incorporated a large number of industry requirements into its latest specification
  - Has created a predictable licensing framework for all MSA Component JSRs
- Get ready for JSR 248 Specification v1.0
  - MSA
  - MSA subset
  - Works both on CLDC and CDC
- Products supporting MSA are coming to the market soon!

java.sun.com/javaone/sf

For More Information

Mobile Service Architecture
(JSR 248) website:

- http://www.jcp.org/en/jsr/detail?id=248

Mobile Service Architecture Advanced
(JSR 249) website:

- http://www.jcp.org/en/jsr/detail?id=249

java.sun.com/javaone/sf

Q&A

Mobile Service Architecture Initiative:
JSR 248 Hitting the Market Soon

java.sun.com/javaone/sf

Mobile Service Architecture Initiative: MSA Hitting the Market Soon

Kay Glahn

Consultant Mobile Service Architecture, Vodafone
http://www.vodafone.com

Asko Komsi

Director
Nokia
http://www.nokia.com

TS-4936

java.sun.com/javaone/sf