



Java is a trademark of Sun Microsystems, Inc.



NOKIA

JavaOneSM

Mobile Service Architecture 2: Introducing New Features in Mobile Devices

Kay Glahn

Consultant Mobile
Service Architecture
Vodafone

Erkki Rysä

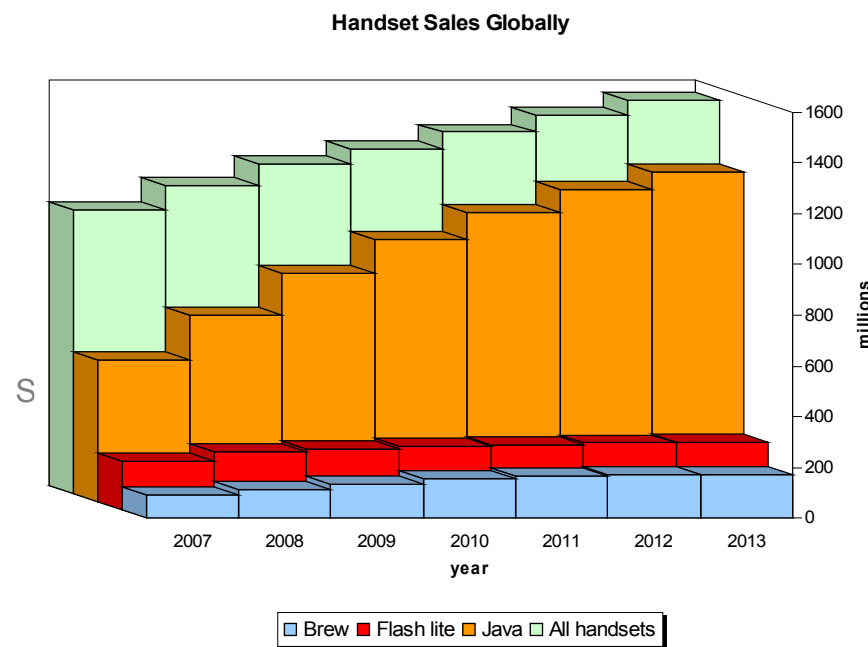
Technology Manager
Nokia

Agenda

- > Mobile Service Architecture Initiative
- > MSA 1
- > MSA 2
- > Summary

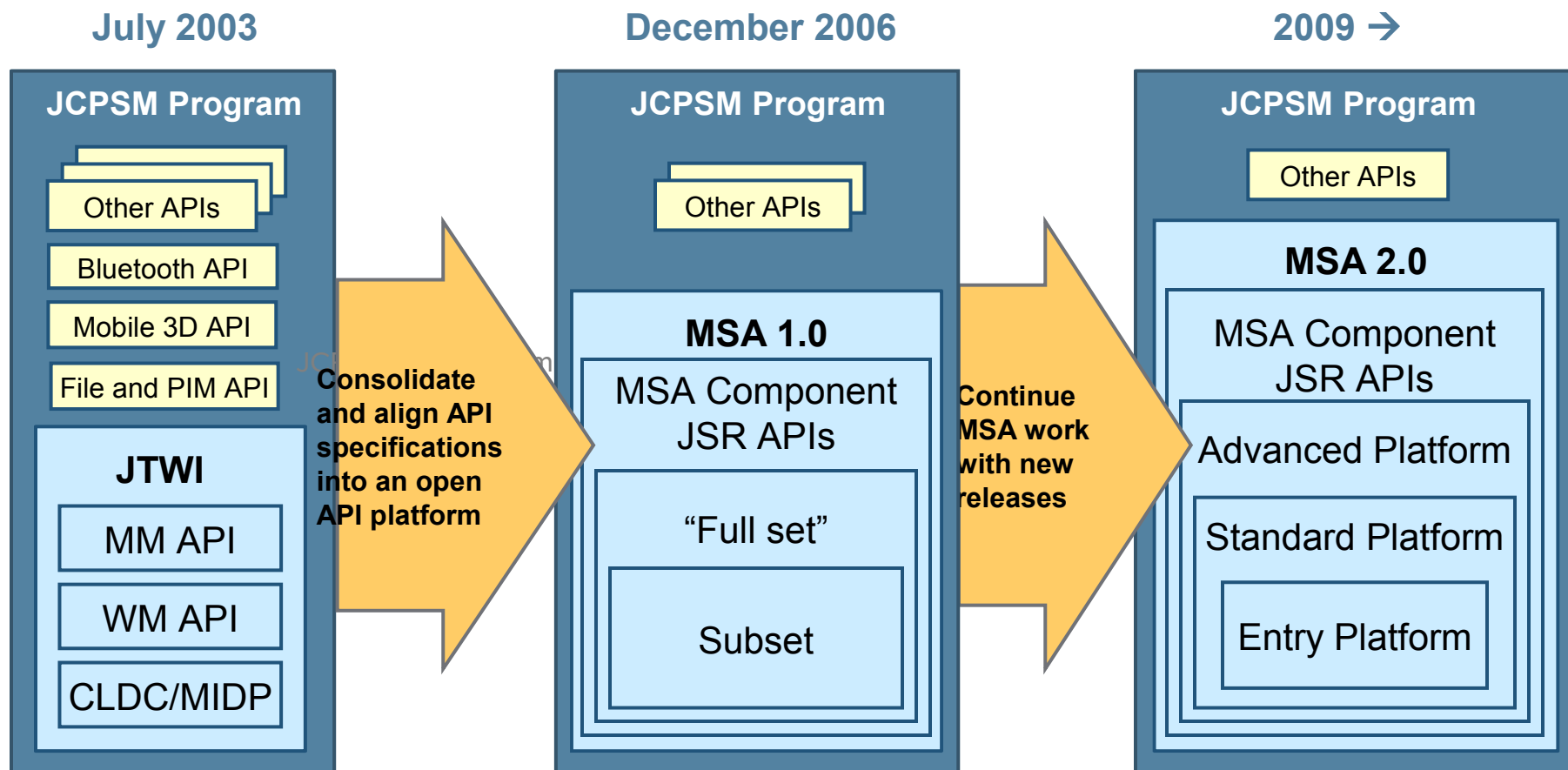
Java Technology Deployment Globally

- > 60% of handsets shipped in 2008 were estimated to be Java technology-enabled.
- > 70% of handsets shipped in 2009 are expected to be Java technology-enabled.
- > Over 900 million new Java technology-enabled handsets this year alone.
- > **But** Java technology implementations in handsets are fragmented:
 - Different APIs
 - Different optional packages
 - Different protocols and content formats
 - Different interpretations of the same specifications



MSA Initiative

Simplifying the Java API Landscape



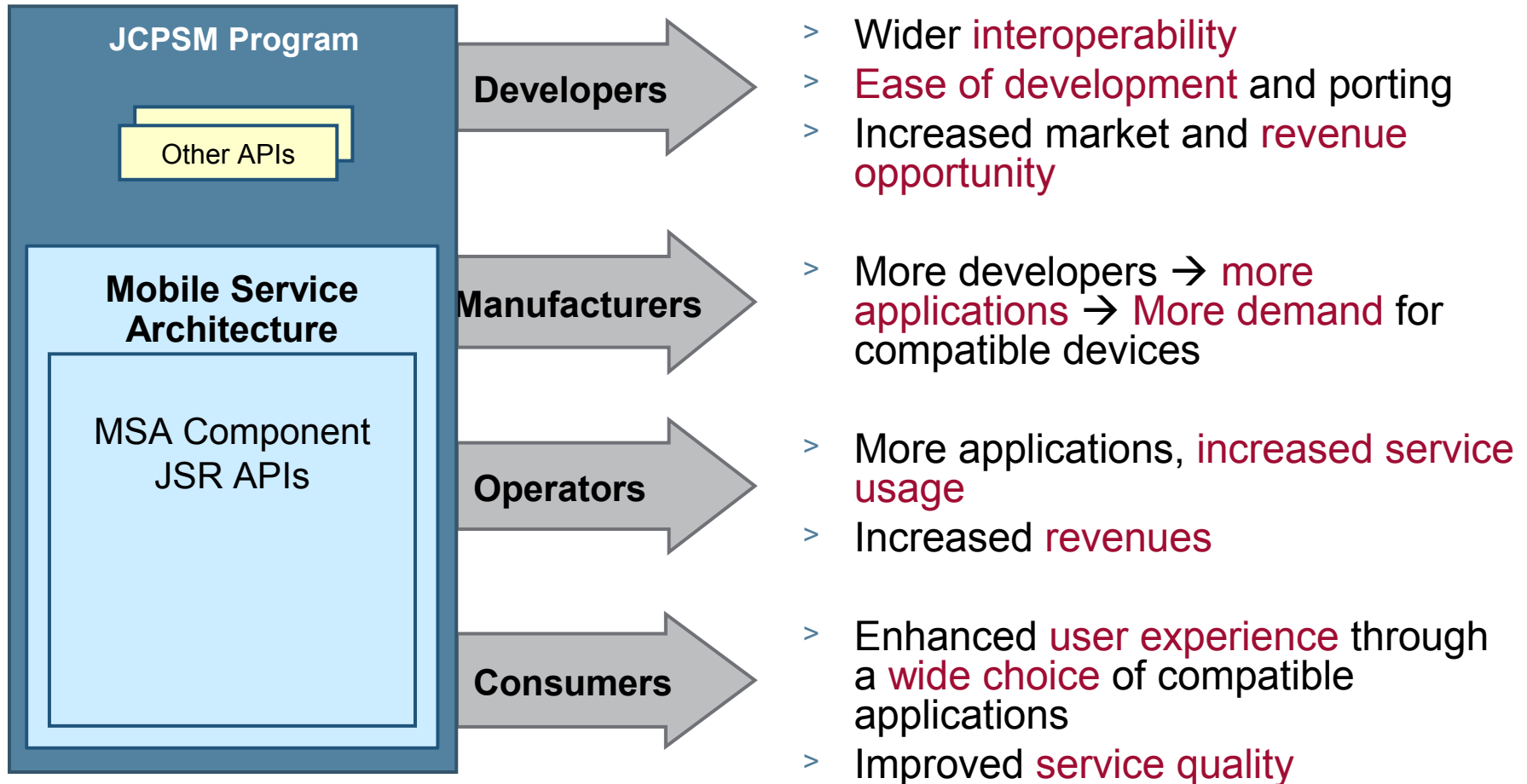
MSA Initiative

Five Steps Towards a Platform

- > **Selecting JSRs** to form the MSA platform
 - Deciding on necessary functionality, time-to-market, overall resource requirements, end-to-end availability, etc.
- > **Specifying clarifications** to reduce ambiguity and fragmentation
 - Some JSRs are targeting a wider scope by providing options
 - Interaction of JSRs is not always specified
 - Interoperability with the device OS or the network infrastructure is missing in most JSRs
 - Some early implementations showed room for improvement
- > **Specifying additional requirements**
 - Protocols, media types, security, hardware, etc.
- > **Providing compliance testing**
 - Technology Compatibility Kit (TCK)
 - Reference Implementation (RI)
- > **Providing a consistent licensing framework**
 - Predictable licensing conditions for all component JSRs

MSA Initiative

Industry Benefits



Current MSA 2 EG Members

Operators

- > AT&T (SBC)
- > China Mobile Communications Co. Ltd
- > NTT DoCoMo, Inc.
- > Orange France SA
- > Sprint
- > T-Mobile
- > Telefonica
- > TeliaSonera AB
- > Vodafone Group Services Limited

Device Manufacturers

- > LG Electronics Inc.
- > Motorola
- > Nokia Corporation
- > Research In Motion, LTD (RIM)
- > Samsung Electronics Corporation
- > Sony Ericsson

Software/Middleware Vendors

- > Aplix Corporation
- > BEA Systems
- > Ericsson AB
- > Esmertec AG
- > IBM
- > Intel Corp.
- > ProSyst Software GmbH
- > Siemens AG
- > Sun Microsystems, Inc.

Agenda

- > Mobile Service Architecture Initiative
- > MSA 1
- > MSA 2
- > Summary

MSA 1 Architecture

**High
Device
Segment**

**Mid
Device
Segment**

MSA 1

MSA 1.1 “Full set”

- JSR 180 - SIP
- JSR 177 - SATSA - PKI **
- JSR 177 - SATSA - CRYPTO
- JSR 172 - Web Services
- JSR 234 - Multimedia Suppl.
- JSR 179 - Location API 1.0.1 **
- JSR 211 - Content Handler
- JSR 177 - SATSA - APDU **
- JSR 238 - Internationalization

MSA 1.1 Subset

- JSR 226 - Vector Graphics 1.1
- JSR 184 - 3D Graphics
- JSR 082 - Bluetooth **
- JSR 075 - File and PIM
- JSR 205 - Messaging 2.0
- JSR 135 - Mobile Media
- JSR 118 - MIDP 2.1
- JSR 139 - CLDC 1.1

MSA 1 Component JSRs

MSA Subset

- > JSR 135: Mobile Media
 - Sampled audio
 - Synthetic audio
 - Video and Still image capture

- > JSR 205: Messaging
 - SMS (Text) message send and receive
 - MMS (Multimedia) message send and receive

- > JSR 75: File and PIM
 - Accessing device file system (File)
 - Supports removable media, such as memory cards (File)
 - Accessing calendar and contacts (PIM)

- > JSR 82: Bluetooth
 - Bluetooth service/device discovery and communication
 - OBEX

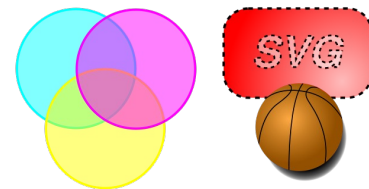


MSA 1 Component JSRs (continued)

MSA Subset

- > JSR 184—3D Graphics
 - 3D Graphics API
 - 3D Graphics file format

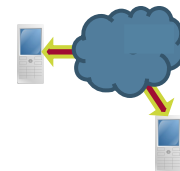
- > JSR 226—Scalable Vector Graphics
 - W3C Scalable Vector Graphics (SVG) support
 - Loading Java 2D™ API content
 - Modifying Java 2D API content through API calls
 - Rendering and playing Java 2D API content
 - Interacting with Java 2D API content using event listeners



MSA 1 Component JSRs

MSA “Full set”

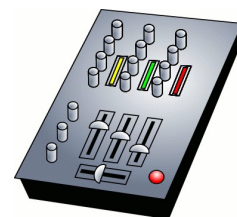
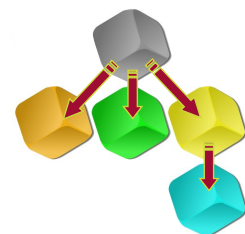
- > JSR 172—Web Services
 - Subset of Java SE JAXP 1.2 and SAX 2.0 support (XML Package)
 - Subset of Java SE JAX-RPC 1.1 (Web Services Package)
- > JSR 172—Security and Trust (SATSA)
 - Communication with smart cards using the APDU protocol (APDU)
 - Digital signatures and basic user credential management (PKI)
 - Subset of Java SE platform Cryptography API (CRYPTO)
- > JSR 179—Location
 - Obtaining the current location and make calculations
 - Different location methods (internal GPS, external GPS)
 - Landmark support
- > JSR 180—SIP (Session Initiation Protocol)
 - Support for sending and receiving SIP messages
 - P2P communication over the network



MSA 1 Component JSRs (continued)

MSA “Full set”

- > JSR 211—Content Handler
 - Launching external applications from Java applications
 - Launching Java applications to handle content
- > JSR 234—Multimedia Supplements
 - Audio effects and 3D audio
 - Image post-processing and encoding
 - Camera and Radio tuner control
- > JSR 238—Internationalization
 - Locale-specific formatting of dates, times, numbers (including percentages) and currency amounts
 - Retrieving application and device specific resources
 - Locale-specific collation (sorting) of strings



MSA Phones from Nokia (MSA Subset)

Series 40

> Series 40 5th Edition

- Nokia 6555 (from AT&T in US)
- Nokia 6263 (from T-Mobile in US)
- Nokia 6267
- Nokia 7500 Prism

> Series 40 5th Edition Feature Pack 1

- Nokia 2700 classic
- Nokia 2730 classic
- Nokia 3120 classic
- Nokia 3555
- Nokia 3600 slide
- Nokia 5130 XpressMusic
- Nokia 5220 XpressMusic
- Nokia 5310 XpressMusic
- Nokia 5610 XpressMusic
- Nokia 6208 classic
- Nokia 6212 classic
- Nokia 6216 classic
- Nokia 6300i
- Nokia 6301
- Nokia 6500 classic
- Nokia 6500 slide
- Nokia 6600 fold
- Nokia 6600 slide
- Nokia 6600i slide
- Nokia 7210 Supernova
- Nokia 7310 Supernova
- Nokia 7610 Supernova
- Nokia 7900 Prism
- Nokia 8800 Arte
- Nokia 8800 Carbon Arte
- Nokia 8800 Gold Arte

> Series 40 6th Edition

- Nokia 5330 XpressMusic
- Nokia 6700 classic
- Nokia 6303 classic
- Nokia 6260 slide
- Nokia 7020
- Nokia 7510 Supernova



MSA Phones from Nokia (MSA Subset)

S60

> S60 3rd Edition, Feature Pack 2

- Nokia 5320 XpressMusic
- Nokia 5630 XpressMusic
- Nokia 5730 XpressMusic
- Nokia 6210 Navigator
- Nokia 6220 classic
- Nokia 6650
- Nokia 6710 Navigator
- Nokia 6720 classic
- Nokia E52
- Nokia E55
- Nokia E71/E71x
- Nokia E75
- Nokia N78
- Nokia N79
- Nokia N85
- Nokia N86 8MP
- Nokia N96/N96-3



> S60 5th Edition

- Nokia N97
- Nokia 5800 XpressMusic



MSA Phones from Sony Ericsson

- > Java Platform 8.0-8.2
 - Sony Ericsson G502
 - Sony Ericsson K630
 - Sony Ericsson K660
 - Sony Ericsson K850/K858
 - Sony Ericsson V640
 - Sony Ericsson W890
 - Sony Ericsson W910/W908
 - Sony Ericsson Z750
 - Sony Ericsson Z770



- > Java Platform 8.3
 - Sony Ericsson C702
 - Sony Ericsson C902
 - Sony Ericsson W595
 - Sony Ericsson W760
 - Sony Ericsson W902
 - Sony Ericsson W980
 - Sony Ericsson Z780
 - Sony Ericsson T700



MSA Phones from Sony Ericsson

- > Java Platform 8.4
 - Sony Ericsson C510
 - Sony Ericsson C901
 - Sony Ericsson C903
 - Sony Ericsson C905
 - Sony Ericsson G705
 - Sony Ericsson T707
 - Sony Ericsson W508/W518
 - Sony Ericsson W705
 - Sony Ericsson W715
 - Sony Ericsson W995
- > Java Platform 8.5
 - Sony Ericsson Aino
 - Sony Ericsson Yari



MSA Phones from LG

> MSA

- KC910/KC910I/KC910E/
KC910QA/KC910Q



> MSA Subset

- KF690, KF700, KF701
- KF750, KF755, KF757
- CF360
- KS500
- KB770
- KT520
- KT525
- HB620T, HB620/KB620



MSA Phones from Samsung (MSA Subset)

- > MSA Subset
 - Samsung GT-M8800 (Pixon)
 - Samsung GT-S5230, GT-S5233 and GT-S5320 (Star)
 - GT-S5600 (Preston)
 - Samsung I7410



MSA Phones from Motorola (MSA subset)

- > ic902 (Only North America)



Agenda

- > Mobile Service Architecture Initiative
- > MSA 1
- > MSA 2
- > Summary

MSA 2

Scope

- > Originally JSR 249 focused on CDC only
- > Now addresses both limited and more advanced mobile handsets
- > Covers the CDC and CLDC Platform
- > Development goes in line with MIDP 3 where also CDC and CLDC are supported
- > Is based on MIDP 3, MIDP 2.1 as an alternative for low-end devices

MSA 2

Key Goals

- > Build on the success of MSA 1
- > Remain backward compatible with MSA 1
- > Further defragment the mobile Java platform
- > Provide a competitive environment which can keep up with native and web environments in terms of functionality and user experience
- > Build a consistent Java platform around MIDP 3 as a basis
- > Address a broad range of devices from ultra low-end to high-end devices
- > Also cover emerging market devices which have not been addressed in the past

MSA 2

Approach

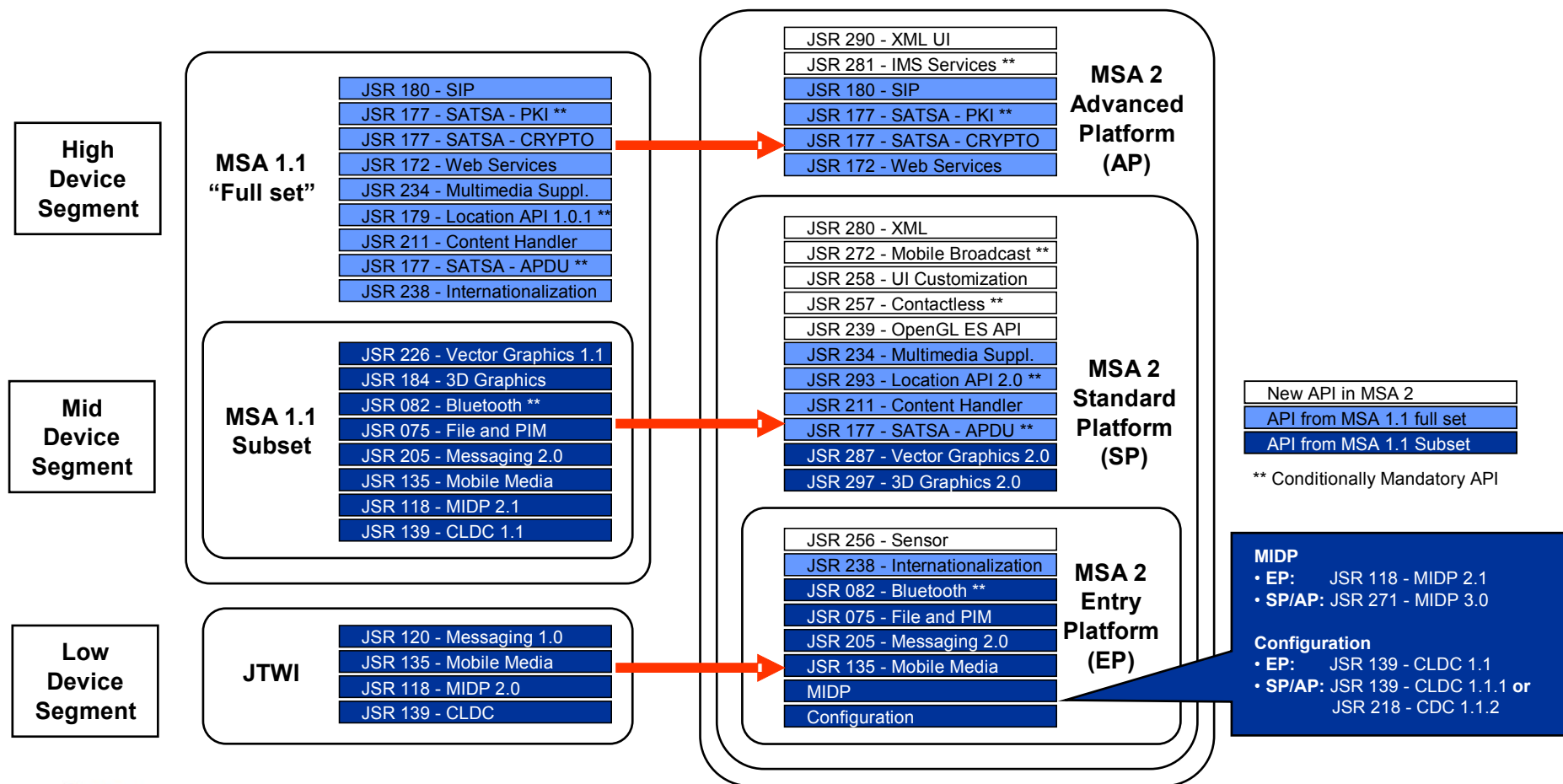
- > Reducing optionalities by additional clarifications
- > Adding interoperability requirements
- > Making as many JSRs as possible conditionally mandatory to provide the same functionality to Java as to native applications
- > Integrating latest technologies and APIs
- > Reducing fragmentation in API selection by providing three consistent API sets:
 - MSA 2 Entry Platform (EP)
 - Added to address low end devices and emerging markets
 - MSA 2 Standard Platform (SP)
 - Evolution of MSA 1 Subset with additional APIs
 - MSA 2 Advanced Platform (AP)
 - Evolution of MSA 1 “Full set” with additional APIs.

MSA Evolution

Overview of Component JSRs

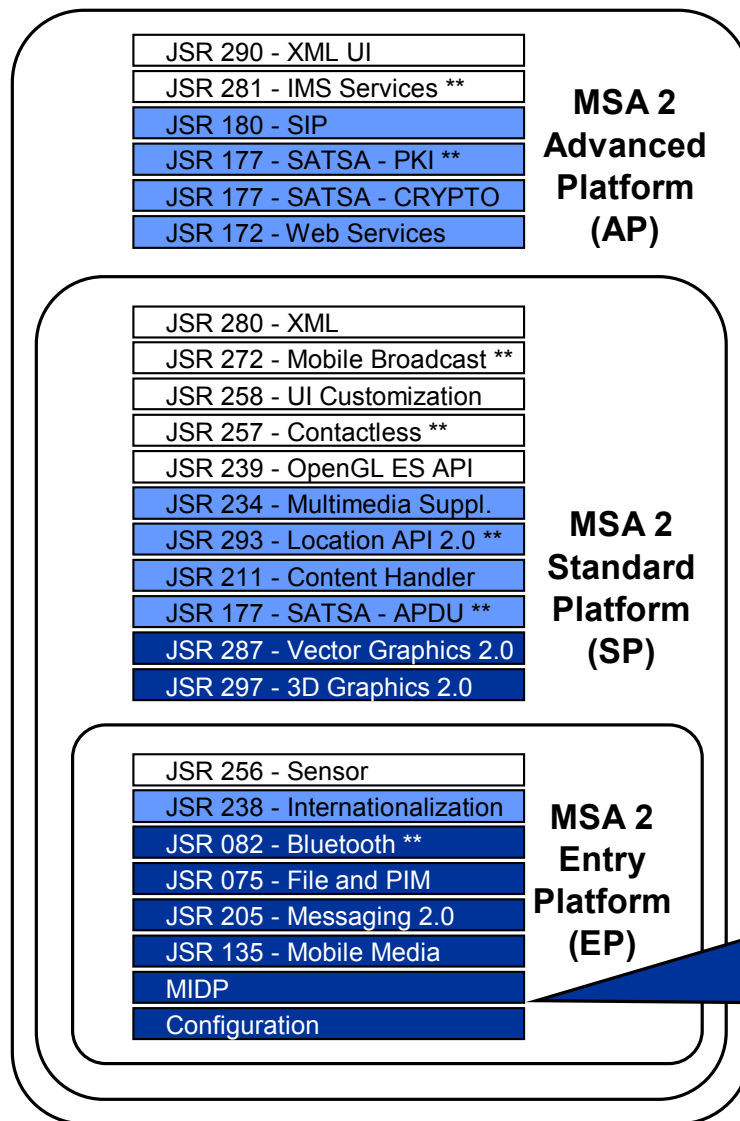
MSA 1 and JTWI

MSA 2



MSA 2 Architecture

MSA 2



New API in MSA 2
API from MSA 1.1 full set
API from MSA 1.1 Subset

** Conditionally Mandatory API

MIDP

- EP: JSR 118 - MIDP 2.1
- SP/AP: JSR 271 - MIDP 3.0

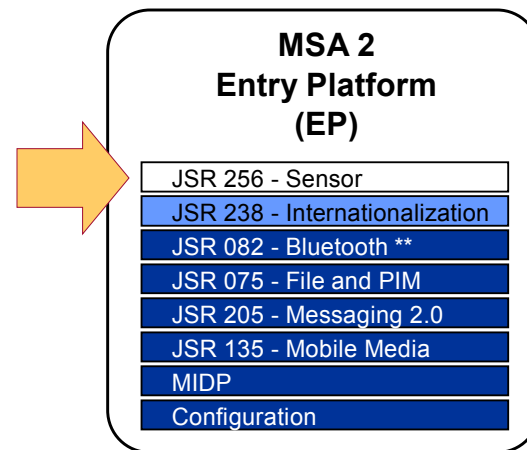
Configuration

- EP: JSR 139 - CLDC 1.1
- SP/AP: JSR 139 - CLDC 1.1.1 or JSR 218 - CDC 1.1.2

MSA 2 - New Component JSRs

Entry Platform

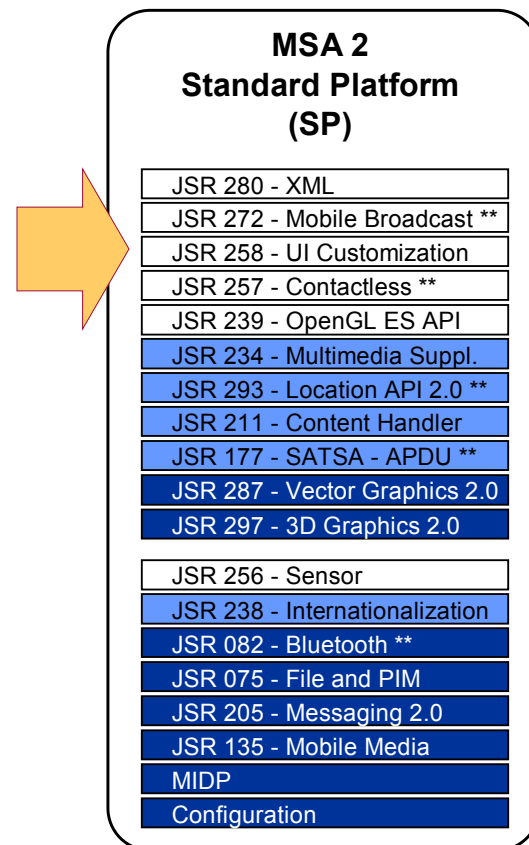
- > JSR 256: Mobile Sensor API
 - Battery charge sensor
 - Charger state sensor
 - Network signal strength sensor
 - Accelerometer sensor



MSA 2 - New Component JSRs

Standard Platform

- > JSR 239: Java Binding for the OpenGL ES API
 - Access to OpenGL ES (Embedded Subset) 3D graphics library
- > JSR 257: Contactless Communication API
 - Access to RFID/NFC tag readers and bar code readers
- > JSR 258: Mobile UI Customization API
 - Querying and modifying UI customization properties
 - Adapting to device look and feel
 - Consistent look and feel across devices using skins and themes
- > JSR 272: Mobile Broadcast Service API
 - Access to digital TV tuner on the device
- > JSR 280: XML API for Java ME
 - General purpose XML API for mobile devices
 - Replaces XML parsing in JSR 172

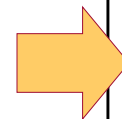


MSA 2 - New Component JSRs

Advanced Platform

- > JSR 281: IMS Services API
 - High-level API to access IP Multimedia Subsystem (IMS) services

- > JSR 290: Java Language & XML UI Markup Integration
 - Use of Web UI markup technologies with Java code
 - Supports the W3C Compound Document Format (CDF)



MSA 2 Advanced Platform (AP)

JSR 290 - XML UI
JSR 281 - IMS Services **
JSR 180 - SIP
JSR 177 - SATSA - PKI **
JSR 177 - SATSA - CRYPTO
JSR 172 - Web Services
JSR 280 - XML
JSR 272 - Mobile Broadcast **
JSR 258 - UI Customization
JSR 257 - Contactless **
JSR 239 - OpenGL ES API
JSR 234 - Multimedia Suppl.
JSR 293 - Location API 2.0 **
JSR 211 - Content Handler
JSR 177 - SATSA - APDU **
JSR 287 - Vector Graphics 2.0
JSR 297 - 3D Graphics 2.0
JSR 256 - Sensor
JSR 238 - Internationalization
JSR 082 - Bluetooth **
JSR 075 - File and PIM
JSR 205 - Messaging 2.0
JSR 135 - Mobile Media
MIDP
Configuration

Standardized Media Format Support

Media Format Support in Mobile Media API (JSR 135)

File format	Content	Coding	Supported level	Platform support			
				EP		SP/AP	
				Playback	Capture	Playback	Capture
Tone sequence	audio			x		x	
MIDI	audio			x		x	
SP-MIDI	audio			x		x	
WAV	audio	8-bit, 8kHz, mono linear PCM		p		x	
AMR-NB	audio			p	p	x	x
AMR-WB	audio					x	x
MP3	audio	20 - 320 kbit/s @ 44.1kHz		x		x	
3GP (any supported video can be combined with any supported audio)	video	H.263	profile 0 level 10	p	c	x	c
		MPEG-4 Part 2	VSP level 0b (1)	p		x	
			VSP level 3 (2)			x	
		H.264 (MPEG-4 Part 10)	Baseline level 1b (1)			x	
			Baseline level 1.2 (1)			x	
			Baseline level 1.3			x	
	audio	AMR-NB		p	c	x	c
		AMR-WB				x	
		AAC-LC				x	
		HE-AAC (AAC+ v2)				x	
MP4 (any supported video can be combined with any supported audio)	video	MPEG-4 Part 2	VSP level 0b (1)			x	c
			VSP level 3 (2)			x	
		H.264 (MPEG-4 Part 10)	Baseline level 1b (1)			x	
			Baseline level 1.2 (1)			x	
			Baseline level 1.3			x	
	audio	AAC-LC				x	c
		HE-AAC (AAC+ v2)				x	
		MP3				x	

required
recommended

x = In all devices implementing the platform.

p = In devices that have a native player for the format.

c = In devices that have a camera.

(1) As defined in 3GPP Rel-6 packet switched streaming document: TS 26.234.

RTSP availability of a given codec subject to support of the related network speed in the device.

(2) As defined in 3GPP Rel-7 packet switched streaming document: TS 26.234.

RTSP availability of a given codec subject to support of the related network speed in the device.

Demo Video – MSA APIs in Action

> Toy car

- Java application (MIDlet) in the phone controls the car
- Used APIs
 - JSR 118: MIDP
 - Application UI
 - JSR 256: Sensor API
 - Accelerometer sensor detects movements of the handset
 - JSR 82: Bluetooth API
 - Sending commands to the car over Bluetooth wireless technology
- Wild idea for future development:
 - Making the car steer itself
 - The phone could be attached to the car.
 - Data from sensors and pictures from the camera could be analyzed and used to steer the car.
 - Multiple phones with different sensors might be needed and the car certainly could not move very fast.
 - But it would still be pretty cool! ☺

Challenges of MSA 2

- > Cover a broad spectrum of devices from highest end to lowest end:
 - Different requirements for different device types but the goal is a consistent platform over the whole range
 - MIDP 2 for low-end devices and MIDP 3 for high-end devices
 - New features of MIDP 3 will not be available in MSA 2 low-end devices → Which feature/clarification should go into MSA 2 and which one into MIDP 3?
- > Dependencies between different JSRs:
 - MSA 2 → Component JSRs → MIDP 3
 - MIDP 3 has to provide TCK/RI first
 - Component JSRs have to provide a CDC compliant TCK
 - Component JSRs have to pass the TCK on top of MIDP 3 (both CLDC and CDC)
 - MSA TCK/RI must be finalized before final approval of the specification
- > Selecting the right set of APIs in order to accommodate everybody without blowing up the memory footprint

MSA 2 - Outlook on Future Development

- > MSA 2.0 is being finalized:
 - Public Review Ballot: Passed in Feb, 2009
 - Proposed Final Draft: June 2009
 - Final Approval Ballot: Q4/2009

- > MSA initiative is an ongoing process
 - Major releases will be available every 18-24 months
 - Maintenance releases planned every 6 months
 - Intent to go in sync with market and business needs

Summary

- > MSA provides a **rich, predictable Java platform** for mobile application development
- > **MSA 1 devices are shipping in huge volumes**
- > **MSA 2 specification finalizing** in 2009
- > **MSA Initiative continues** to steer the evolution of Java ME platform in mobile devices
- > **New releases planned** every 18-24 months, maintenance releases every 6 months

Developer Resources

- > Documentation:
 - Mobile Service Architecture (JSR 248)
 - <http://jcp.org/en/jsr/detail?id=248>
 - Mobile Service Architecture 2 (JSR 249)
 - <http://jcp.org/en/jsr/detail?id=249>
 - Component JSRs
 - <http://jcp.org>
- > Tools to get started:
 - Tools and documentation by Nokia:
 - <http://www.forum.nokia.com/java>
 - Java Platform Micro Edition Software Development Kit
 - <http://java.sun.com/javame/downloads/index.jsp>
- > Discussion board for MSA:
 - <http://discussion.forum.nokia.com/forum/forumdisplay.php?f=185>

How the Community can get Engaged

- > Observer list
- > Discussion Forum
- > Blog
- > Feedback in Public Reviews
- > Engagement with EG and Spec Leads at public events like JavaOne



- > You can download the specification at: <http://jcp.org/en/jsr/detail?id=249>

Provide Feedback!

- > Your feedback is very important
 - > It allows us to further improve the MSA platform
 - > To further reduce fragmentation we need to know
 - Where the real fragmentation issues are?
 - What APIs are missing?
- > Any feedback on the MSA specification is highly appreciated and can be provided through the following email address:
jsr-249-comments@jcp.org

For More Information

> Panel discussion

- **PAN-5336:** MSA 2: How Do We Work Toward a Consistent Java™ Platform?
Thursday, June 04, 4:10 PM - 5:10 PM



JavaOneSM

Thank You

Kay Glahn
Vodafone

Erkki Rysä
Nokia

Send your improvement ideas and
comments to:

jsr-249-comments@jcp.org

