



# Minne menet, Mobiili-Java?

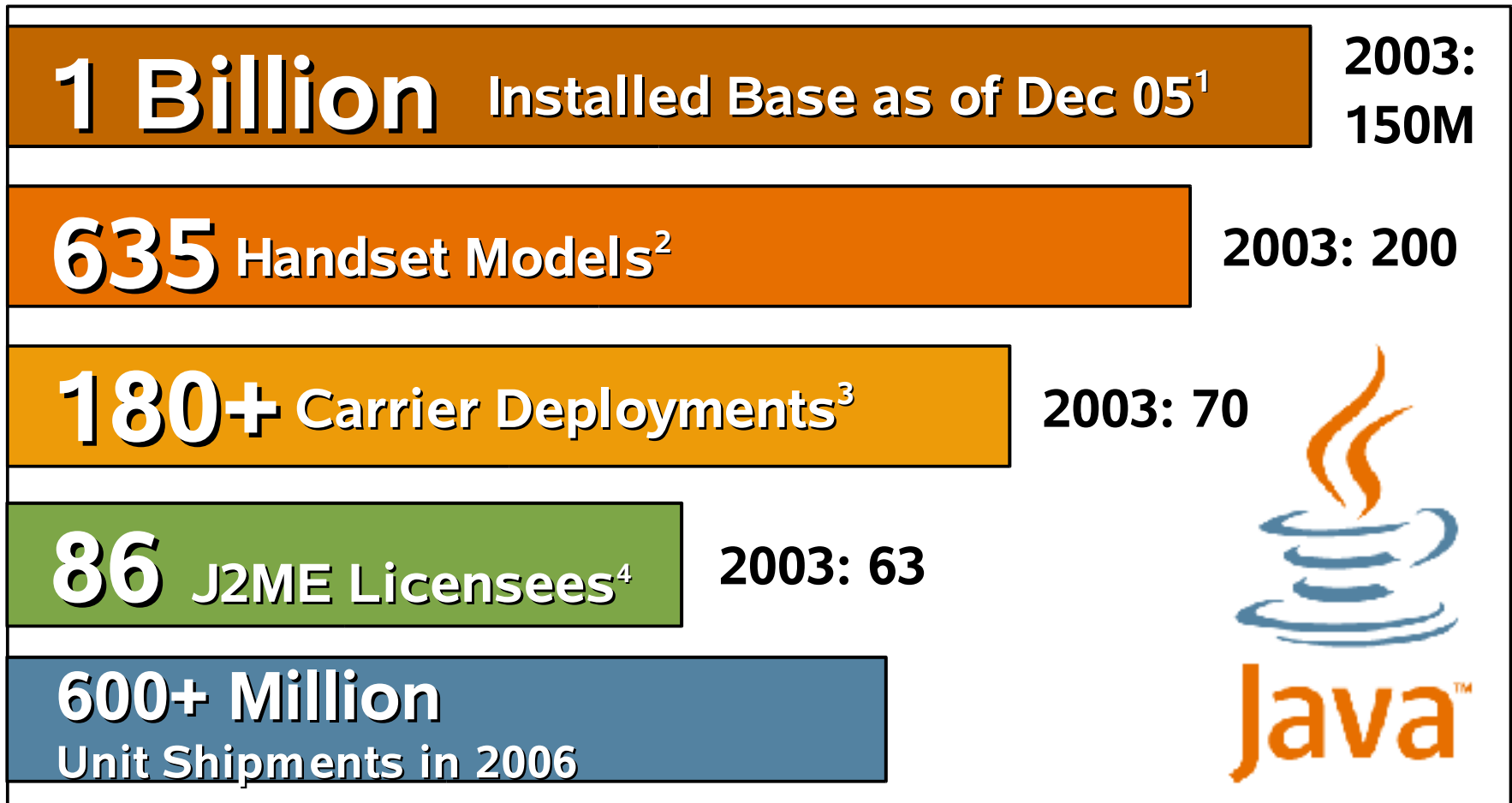
Java Platform, Micro Edition – Status and Future Directions

**Antero Taivalsaari**

Sun Microsystems, Inc.

December 2005

# Growth Continues (2005 vs. 2003)



Sources:

1 – Ovum, Sep'05

2 – Sun Microsystems, June'05

3 – Nokia, Nov'05

4 – Sun, Nov'05

# Agenda

- Java Platform, Micro Edition – Overview
- Forthcoming standards and APIs
  - > MSA 1.0 – the next base JavaME platform
  - > MIDP 3.0 – the next generation of MIDP
  - > Other forthcoming APIs
- Demos (time permitting...)

# JavaME for Wireless – Overview

# Java Micro Edition – Focus Areas

- CDC platform

- > Enterprise & high-end consumer products
- > <http://java.sun.com/products/cdc>

- CLDC platform

- > Consumer products
- > <http://java.sun.com/products/cldc>

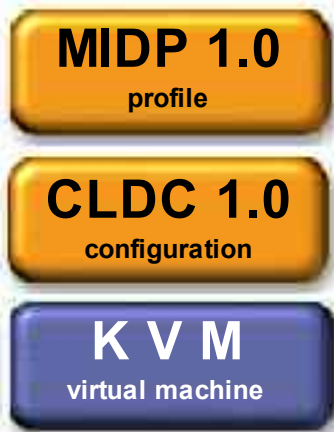
- JavaCard platform

- > Smart cards, point-of-sale terminals
- > <http://java.sun.com/products/javacard>



# Evolution of JavaME for Wireless

- Generation 1: CLDC 1.0 + MIDP 1.0
  - > Standards completed: Summer 2000
  - > Devices shipping: Spring 2001



Motorola i85s



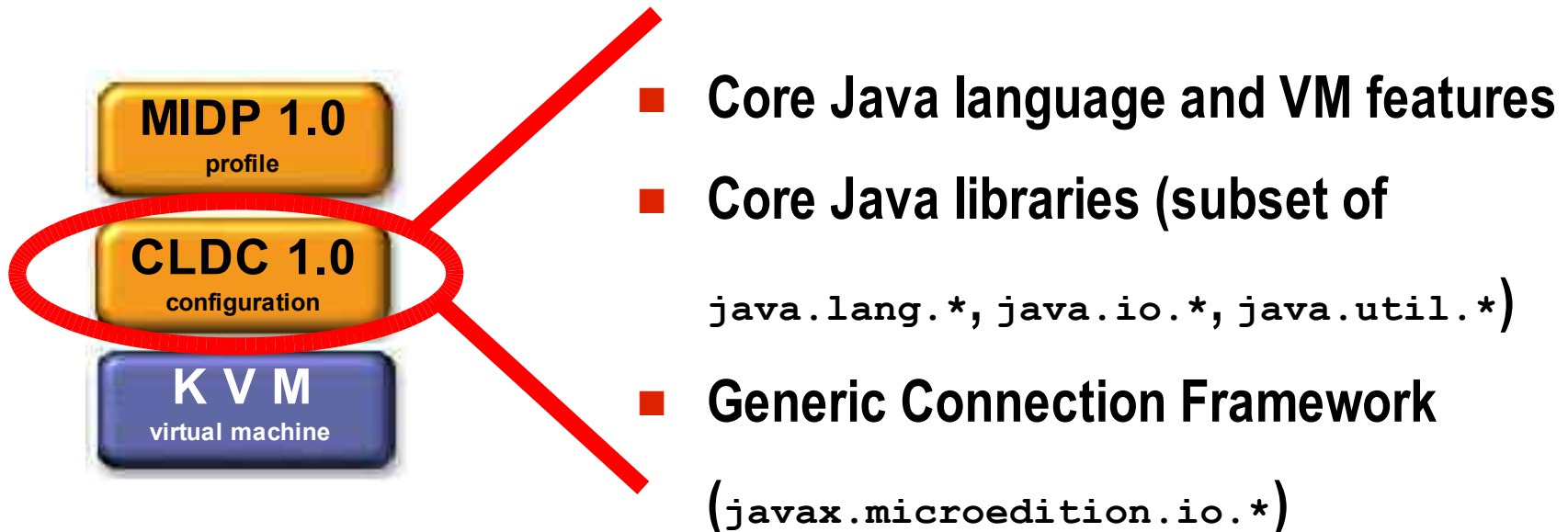
Nokia 3410



Nokia 6310i

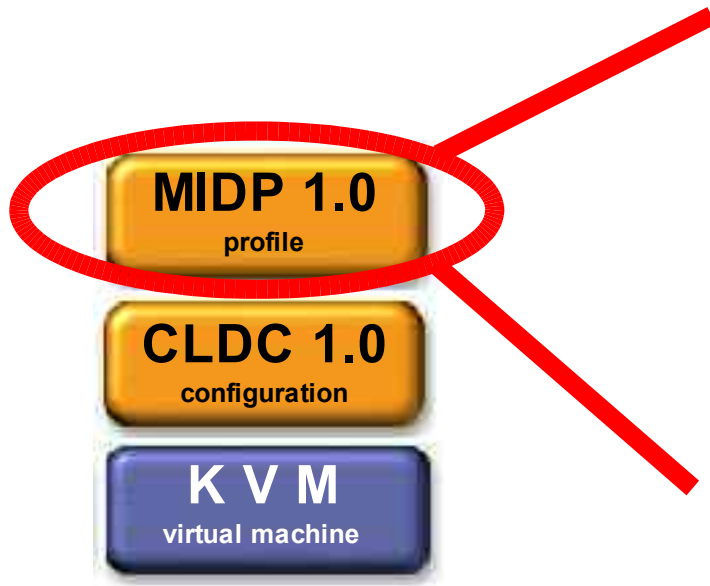
# Evolution of JavaME for Wireless

- Generation 1: CLDC 1.0 + MIDP 1.0
  - > Standards completed: Summer 2000
  - > Devices shipping: Spring 2001



# Evolution of JavaME for Wireless

- Generation 1: CLDC 1.0 + MIDP 1.0
  - > Standards completed: Summer 2000
  - > Devices shipping: Spring 2001



- MIDlet application model (life-cycle, packaging)
- User interface libraries (LCDUI)
- Persistent storage (RMS)
- Networking (HTTP)

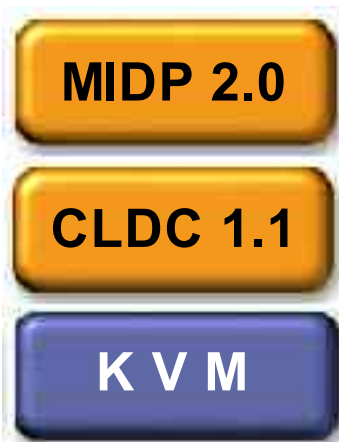


# Evolution of JavaME for Wireless

- Generation 2: CLDC 1.0/1.1 + MIDP 2.0
  - > Standards completed: Fall 2002 / Early Spring 2003

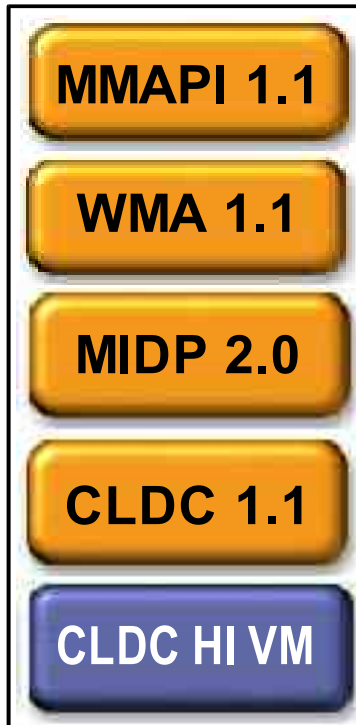
## Major new features in MIDP 2.0:

- Enhanced security model
- Game API
- Sound API
- Additional network protocols, including HTTPS
- User interface enhancements
- Floating point support (CLDC 1.1)



# Evolution of JavaME for Wireless

- Generation 3: **JTWI 1.0**
  - > Standards completed: Summer 2003



## Major new features in JTWI:

- Multimedia API
- Wireless Messaging API (SMS support)
- Various clarifications to reduce platform fragmentation

## New virtual machine technologies:

- CLDC HotSpot (CLDC HI) VM with adaptive compilation

# Current Landscape

- Over 180 mobile operators have deployed JavaME



- Over 1 billion JavaME devices shipped (Dec 2005)
- 635+ different JavaME device models from 30+ manufacturers




- Globally, over 20 million application downloads per month.
- Generated total application revenue exceeds \$3 billion annually.

# Challenges Today

- *Fragmentation* is a major problem!
  - > Ideally, one MIDlet should run on any device. In practice, there are various problems.
- Different types of fragmentation:
  - 1) Device-level fragmentation (screen sizes, etc.)
  - 2) Implementation fragmentation
  - 3) API-level fragmentation
  - 4) Standards fragmentation
- One of the top priorities for next-generation JavaME standards is to reduce fragmentation!

# How Are Java Standards Created?

- Java standards are created using the Java Community Process (JCP). 
- Key characteristics of JCP:
  - > All specifications defined collaboratively by the industry.
  - > All specifications go through a well-defined review process.
  - > All specifications have a reference implementation and a compatibility test suite.
- For further info, see: <http://jcp.org/>

# Next-Generation JavaME Standards

- There is a very active community focusing on JavaME standardization.
  - > All the key players in the wireless industry involved
  - > 33 JavaME standards already completed
  - > 24 JavaME standards activities currently in progress
- Key activities:
  - > MSA 1.0 (Mobile Service Architecture 1.0)
  - > MIDP 3.0 (Mobile Information Device Profile 3.0)

# MSA (Mobile Service Architecture)

# MSA (Mobile Service Architecture)

- MSA overall goal:
  - > “Create the next-generation mobile Java platform for the mass-market wireless handsets, continuing the work started in JSR 185 (JTWI).”
- MSA consists of two JSRs: JSR 248 and 249
  - > *JSR 248: Consumer device focus*
  - > *JSR 249: Enterprise & high-end consumer focus*
  - > The same expert group working on both activities.
  - > Key device manufacturers, operators and IT vendors involved.
  - > Specification co-leads: Nokia and Vodafone.



# MSA Technical Approach

- MSA will:
  - 1) Choose the base set of Component APIs that will be available on all next-generation JavaME devices.
  - 2) Define clarifications that will define the proper behavior of the Component APIs
  - 3) Define roadmap for the future versions of the platform.
- Guiding principles:
  - > Reduce fragmentation.
  - > Minimize optional features.
  - > Focus on the needs of mass-market handsets.

# MSA/CLDC 1.0 Platform

## J2ME MSA/CLDC Platform 1.0 (JSR 248)

PDAOP (JSR 75)

*Bluetooth/Obex (JSR 82)*

MMAPI 1.1 (JSR 135)

Web Services (JSR 172)

SATSA (JSR 177)

*Location (JSR 179)*

SIP (JSR 180)

3DGraphics (JSR 184)

CHAPI (JSR 211)

SVG (JSR 226)

WMA 2.0 (JSR 205)

Payment API (JSR 229)

Adv. media (JSR 234)

MI18N (JSR 238)

MIDP 2.0 (JSR 118)

CLDC 1.1 (JSR 139)

# MSA/CLDC 1.0 API Highlights

- PDA Optional Packages (JSR 75)
  - 1) PIM API – provides access to address book, calendar events, to do items, etc. (vCalendar, vCard)
  - 2) FileConnection – access to the local file system
- Bluetooth/OBEX (JSR 82)
  - > Enables Java developers to utilize Bluetooth services
- Web Services (JSR 172)
  - 1) XML Parser API (JAXP 1.2 subset)
  - 2) JAX-RPC – provides access to SOAP-based web services

# MSA/CLDC 1.0 API Highlights

- Security and Trust (JSR 177)
  - 1) SIM card access API (APDU)
  - 2) Public Key Infrastructure (PKI) – support for app-level digital signatures & basic user credential management
  - 3) Crypto support (subset of the J2SE cryptography API)
- Location API (JSR 179)
  - > Enables the development of location-based applications
- Session Initiation Protocol (SIP) API (JSR 180)
  - > Provides access to SIP-based Internet conferencing, telephony, presence, event notification and instant messaging services

# MSA/CLDC 1.0 API Highlights

- 3D Graphics API (JSR 184)
  - > Defines an API for rendering 3D graphics, including a scene graph structure and a corresponding file format for the management and deployment of 3D content
- Wireless Messaging API 2.0 (JSR 205)
  - > Defines an API for sending & receiving SMS/MMS msgs.
- SVG (Scalable Vector Graphics) API (JSR 226)
  - > Defines an API for rendering Scalable 2D vector images, including external images in SVG format.
  - > Use cases: map visualization, scalable icons and other applications that require scalable, animated graphics.

# MSA/CLDC 1.0 API Highlights

- Content Handler API (JSR 211)
  - > Enables the registration of Java and non-Java applications in such a way that they are automatically launched when a certain URL or a certain content type is invoked (e.g., from a browser).
- Payment API (JSR 229)
  - > Defines an API to initiate payment transactions in a secure manner.
  - > The API is payment adapter agnostic, i.e., the actual supported payment protocol (SMS, credit card, ...) may vary.

# MSA/CLDC 1.0 API Highlights

- Advanced Multimedia Supplements (JSR 234)
  - > Supports advanced audio effects such as 3D audio, reverb, equalizer (EQ), and musical effects.
  - > Also provides additional capabilities for image processing, radio tuner and camera settings.
- Mobile Internationalization (JSR 238)
  - > Supports the locale-specific formatting of dates, times, numbers (including percentages) and currency amounts.
  - > Also defines a mechanism to retrieve application-specific and device-specific resources, and provides services for the locale-specific sorting of strings.

**All these specifications are available from [jcp.org](http://jcp.org)**

# MSA Status

- MSA/CLDC 1.0 Specification Public Review was completed in October 2005.
- Final Review and Final Release expected in spring 2006.
- Devices expected to ship in mid-to-late 2006.

=====

- Send your feedback to:  
['jsr-248-comments@jcp.org'](mailto:'jsr-248-comments@jcp.org')



# MIDP3

# MIDP3 Overview

- MIDP3 (JSR 271) will define a new version of the highly popular *Mobile Information Device Profile* standard.
  - > Specification lead: Motorola
- Main goals:
  - > Maintain and enhance the role of MIDP as the leading mobile application environment.
  - > Add functionality requested by the developer community.
  - > Maintain backwards compatibility with MIDP 2.0.
  - > Maintain focus on high-volume mobile handsets.
  - > Reduce ambiguity & tighten specification in all areas.

# MIDP3 Feature Highlights

- Multitasking (MVM) support:
  - > Multiple MIDlet suites running simultaneously.
  - > Background MIDlets, event-launched MIDlets
- Shared libraries for MIDlets (LIBlets)
- UI enhancements
  - > Better support for devices with larger displays
  - > Support for multiple/secondary displays
  - > More scalable graphics model, new widget types
  - > Better font support, animated images, anti-aliasing, alpha blending, better scaling/rotation/transparency support, etc. etc.
- Secure/Removable/Remote RMS stores
- Networking enhancements: Multiple network interfaces – IPv6, VPN, IPSEC, MobileIP (to be decided)

# MIDP3 Status

- MIDP3 is still in an early stage.
  - > Expert group work was started in May 2005.
  - > Use case analysis completed in Sep/Oct 2005.
  - > Requirements work is nearly complete.
  - > API design work started recently.
- Early Draft Review planned for Spring 2006.
- Public Review by JavaOne 2006 (?)
- Final Release by the end of year 2006.

# Other Important Ongoing Activities

- Mobile Telephony API (JSR 253)
- Mobile Sensor API (JSR 256)
- Contactless communication (JSR 257)
- Ad hoc networking (JSR 259)
- IMS Services API (JSR 281)
- Refer to <http://jcp.org/> for details.

**Questions?**