Next-Generation Java Card™ Technology for Secure Mobile Applications

Eric Vétillard

CTO
Trusted Labs
www.trusted-labs.com

Saqib Ahmad
Java Card Engineer

Florian Tournier
Group Marketing Manager

Sun Microsystems
www.sun.com

TS-5686

Goal

What will you gain from this session

Learn about the security requirements of new mobile applications and about the upcoming Java Card technology to support them

About the Speakers

- Eric Vétillard
  - Chairman, Java Card Forum Technical Committee
  - CTO of Trusted Labs, consulting and evaluation services for Java Card technology and more
- Saqib Ahmad
  - Engineering lead of Java Card platform engineering group at Sun Microsystems, Inc.
- Florian Tournier
  - Group Marketing Manager at Sun Microsystems, Inc.

Agenda

The new wave of mobile applications
- **The smarter mobile environment**
- NFC or the phone as a smart card
- A smart card web server
- **New s**ecurity challenges

Next-Generation Java Card technology

Java™ Technology in a Mobile Environment

- The SIM hosts Java Card platform applications
  - User interface encoded on the card
  - Fairly basic user experience
  - Provides strong security and management services
  - SIM Toolkit APIs enable access to phone resources
- Mobile phones host Java Platform, Micro Edition, (Java ME platform) applicatio
  - Good level of user interface
  - Limited security level
  - Connection to cards using SATSA (Java Specification Request (JSR) 177)

What's New with Handsets ?

- The mobile phone keeps evolving
  - Wider application range
  - Ever improving user experience
  - More external interfaces
  - Enhanced network connectivity

- One small revolution gets overlooked : NFC
  - Standard RFID interface for phones
  - A phone can behave like a card/token, or like a reader
  - The interface is very natural: just swipe your phone

What's New with SIM card[illegible]?

- Moore's law applies to smart cards
  - Very soon: 16K or 32K of RAM
  - 100's KB of EEPROM/Flash, 1MB of ROM
- Price of basic cards has fallen to commodity level
  - Card management is the bulk of the cost
- The security level of SIM cards is increasing
  - The convergence with banking cards has started
- But the model is reaching a limit
  - SIM Toolkit is aging
  - Card protocols (APDU) are a developer bottleneck

Phone/SIM Integration Is Changing

- USB as standard interface
  - USB is widely available on PCs
  - USB has also on selected mobile phones
- TCP/IP as standard protocol
  - Cards and tokens can communicate with the world
- Then, remember: A smart card is a server
  - A Secure and Personal Server
  - A server programming model is next

Agenda

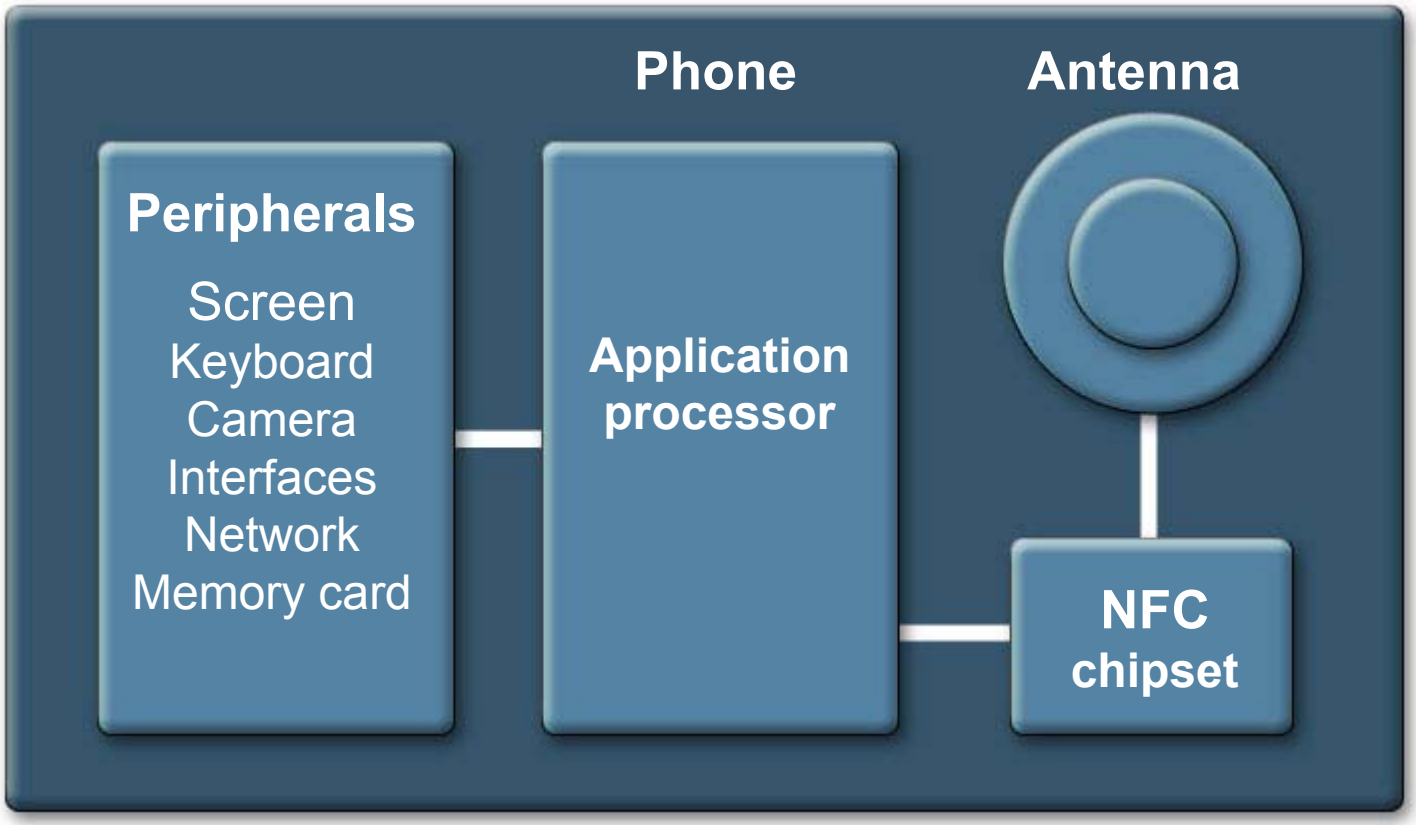
The new wave of mobile applications
- The smarter mobile environment
- **NFC or the phone as a smart card**
- A smart card web server
- **New s**ecurity challenges

Next-Generation Java Card technology

Sensitive Mobile Applications

- NFC brings smart card applications to phones
  - Payment: pay with your phone
  - Transport: enter the subway with your phone
  - Identity: enter your company with your phone
- The phone form factor has advantages over all other form factors
  - A card, or other specific token (key fob)
  - A USB memory stick with an antenna
- Integration into one device is the key factor

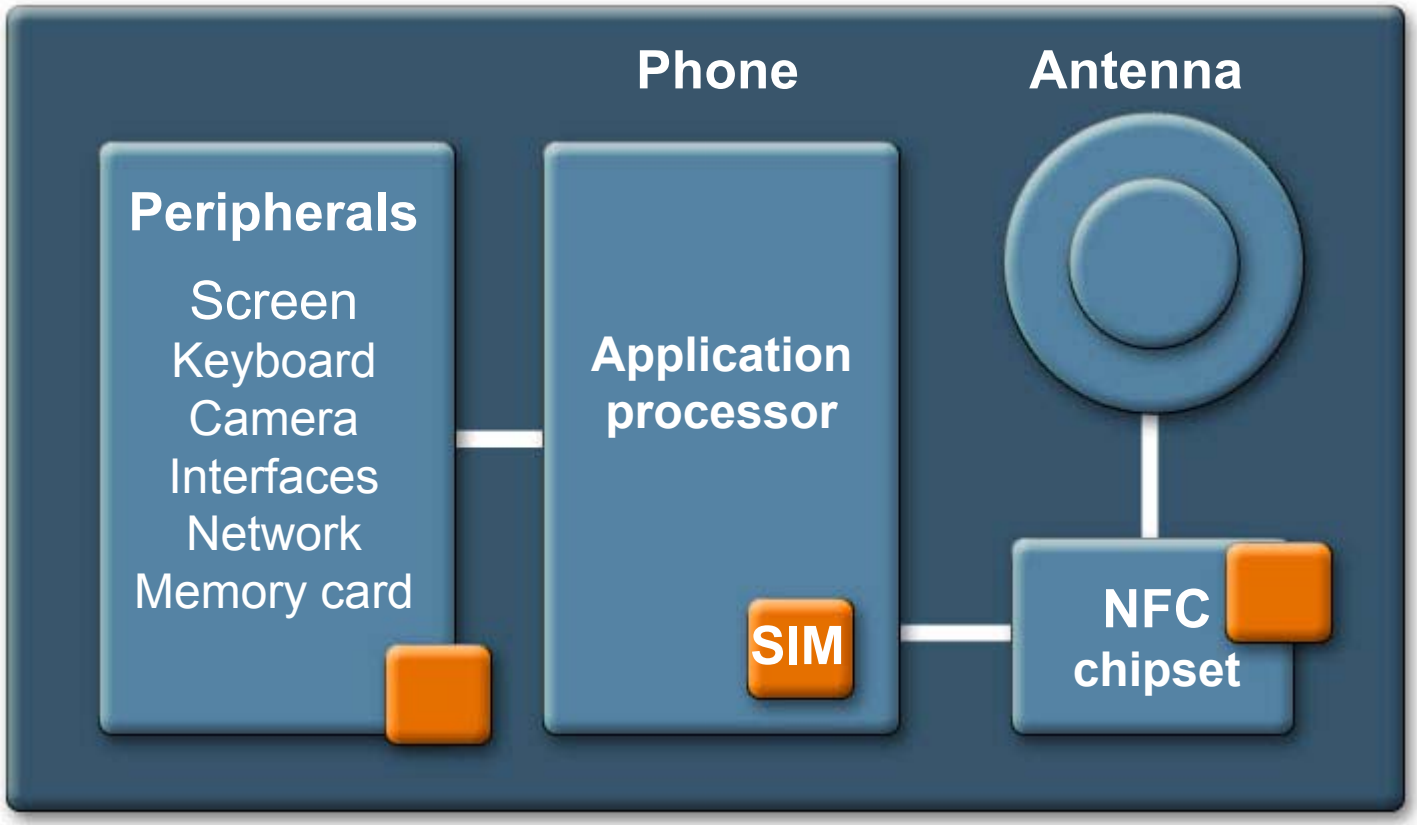
Integrating NFC Into a Mobile Phone



Running Sensitive Applications

- Payment, transport, identity…
  - They are all sensitive applications
  - Usually running on secure smart card cores
  - Can they run on a mobile phone?
- Mobile phones need secure tokens

Adding a Secure Token



Possible Scenario

Evolution of the mobile smart card application

- We start by a standard smart card application
  - Using the phone's NFC interface
- Some interaction is required by the end user
  - User interaction is provided by a MIDP application
  - The application accesses the SIM through JSR 177 or through a standard HTTPS connection
  - A HTTPS connection can also be used with a server
- The smart card application is more complex
  - In particular, it must manage Internet

Possible Scenario

A transport application

- The application is on the secure token
  - With NFC interface, it behaves like a card/token
  - The user can get on the network, buy tickets, etc.
- The application is accessible from the phone
  - The user can buy tickets online, look at the balance
- The application is accessible from the network
  - It can connect to a remote server for maintenance
  - It can be disabled remotely if required

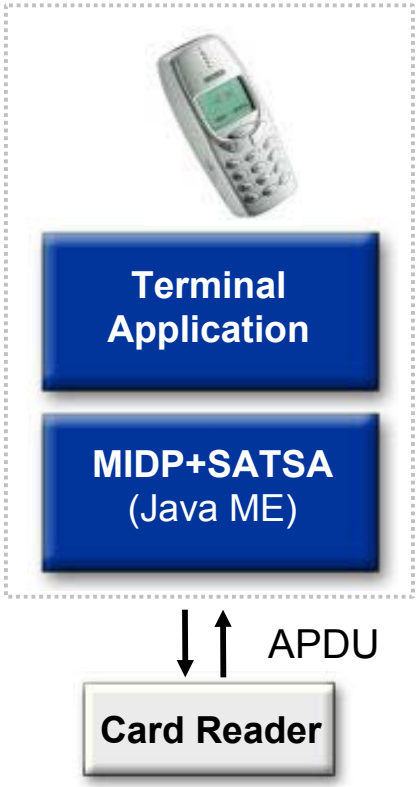
Agenda

The new wave of mobile applications
- The smarter mobile environment
- NFC or the phone as a smart card
- **A smart card web server**
- **New** security challenges

Next-Generation Java Card technology

Breaking Away from APDUs

- SIM cards currently use the ISO 7816 protocol (APDU) to communicate with the outside world
- Applications spend a vast portion of time parsing ISO commands
- More powerful smart cards with and TCP/IP connectivity enable a better programming model



Why Put a Web Server Inside a SIM?

- A logical evolution
  - A SIM card is already a server
  - SIM cards with MBs of memory and high speed interface can process and serve rich content
  - Servlets are well known to developers
  - All phones have browsers
- An improved user and developer experience
  - Leverage standardized web protocols to provides security services to the phone user
  - Interact with the use in familiar user interface
  - Leverages existing developer experience

Smart Card Web Servers and Java Card Technology

- The First Generation of SIM card webservers is being standardized at ETSI
  - Based on current generation Java Card technology
  - Can handle simple TCP/IP-based communication
  - Can handle multiple interfaces sequentially
  - Limitations on the size of data objects and arrays
- New chips will bring maturity to the concept
  - Designed for TCP/IP—based communication
  - Bigger and faster devices can handle rich content
  - Opportunity for a more capable Java environment

Agenda

The new wave of mobile applications
- The smarter mobile environment
- NFC or the phone as a smart card
- A smart card web server
- **New security challenges**

Next-Generation Java Card technology

Why Do We Need Security?

- Security: defending assets against attackers
- Which assets are we defending?
  - Cryptographic data (keys)
  - Credentials (passwords)
- Against which attackers?
  - The cardholder is a dangerous one
  - A card thief is another

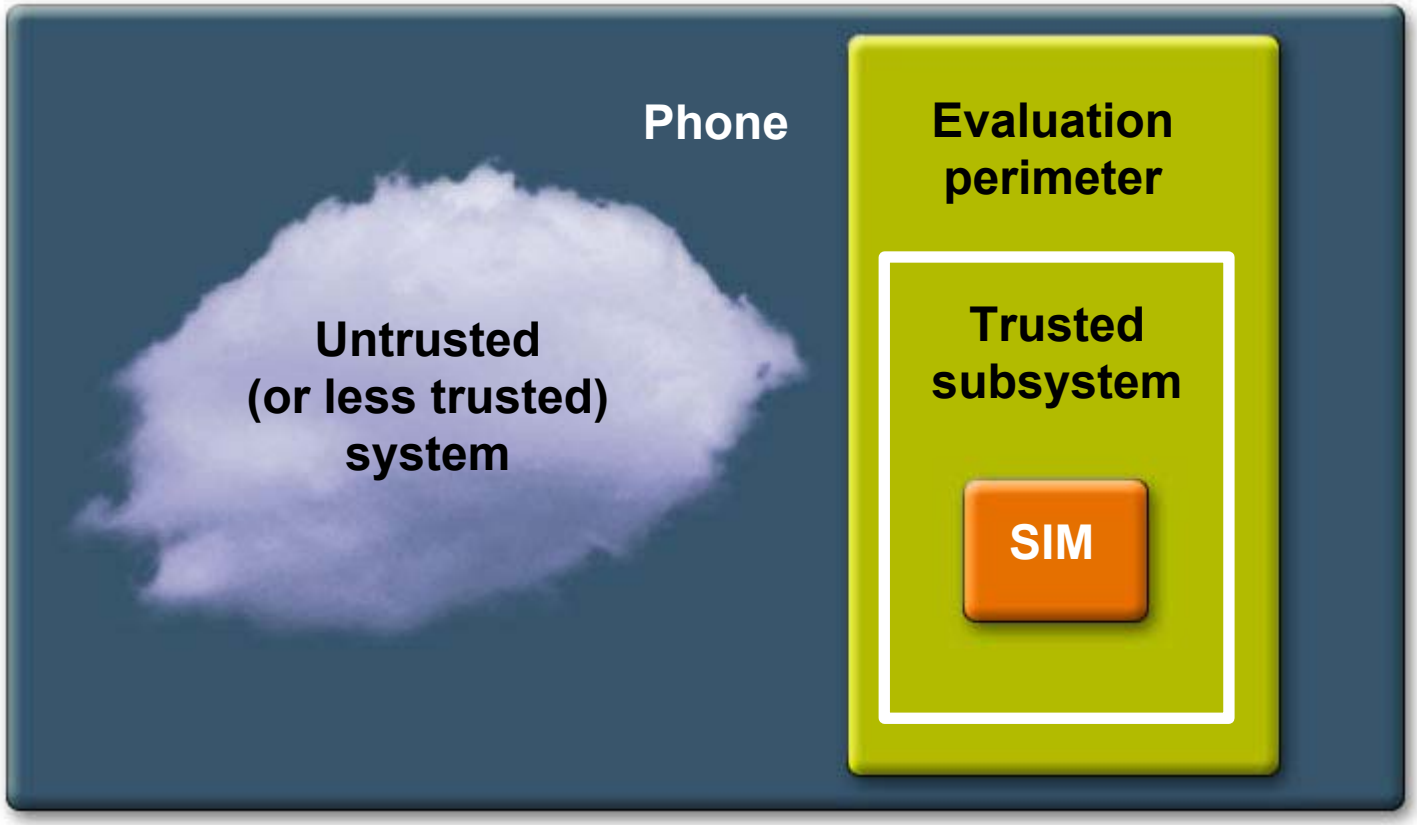
Mobile Phone Security

- Depends a lot on the threat
  - Remote threats are not too bad
    - Not much malware (for now)
    - Especially true for simpler systems
  - Local threats (by owner) quite bad
    - Code base can often be modified
- It is difficult to secure the entire code
  - The smart card can be a root of trust

Mobile Phone vs. Contactless Card

- Contactless cards have some issues
  - They can be abused without the user's consent
- Mobile phones with NFC don't have them
  - It is possible to ask for a confirmation
  - It is even possible to ask for a PIN
- Yes, but ... what is the security level of a phone?
  - Is the data displayed guaranteed to be correct?
  - Can the PIN code be kept confidential?

Smart Card as Root of Trust



Smart Card as Root of Trust

- Smart cards are good roots of trust
  - They provide a reasonably high level of security
  - They are very easy to isolate
- They can minimize the trusted subsystem
  - It can be some kind of a browser
  - The card should then act as a Web server

Why Java Card Technology?

- Because smart cards are secure
- Because Java programming language is a high-level language
- Because it defines a standard interface
  - Platform duties are well identified
  - Application duties are complementary
- Because it is portable
  - We have seen that many architectures are possible
  - A Java Card platform application can be ported easily

Which Duties for Java Card Platform Applications?

- Protect their sensitive assets
  - Keep confidential data protected (encrypted)
  - Verify the integrity of sensitive data
  - Check the execution of sensitive code
- Protect their application interface
  - New interfaces yield new threats
  - Contactless protocols have specific threats
  - Web server architecture bring many threats
    - All classical Web attacks are now feasible
    - The good news is that they are well known

Conclusion—Part I

- New mobile apps demand more from the SIM
- Changes to the role of the SIM and the Handset induce new security constraints
- The Next-Generation Java Card platform architecture must enable new use cases while preserving security

Agenda

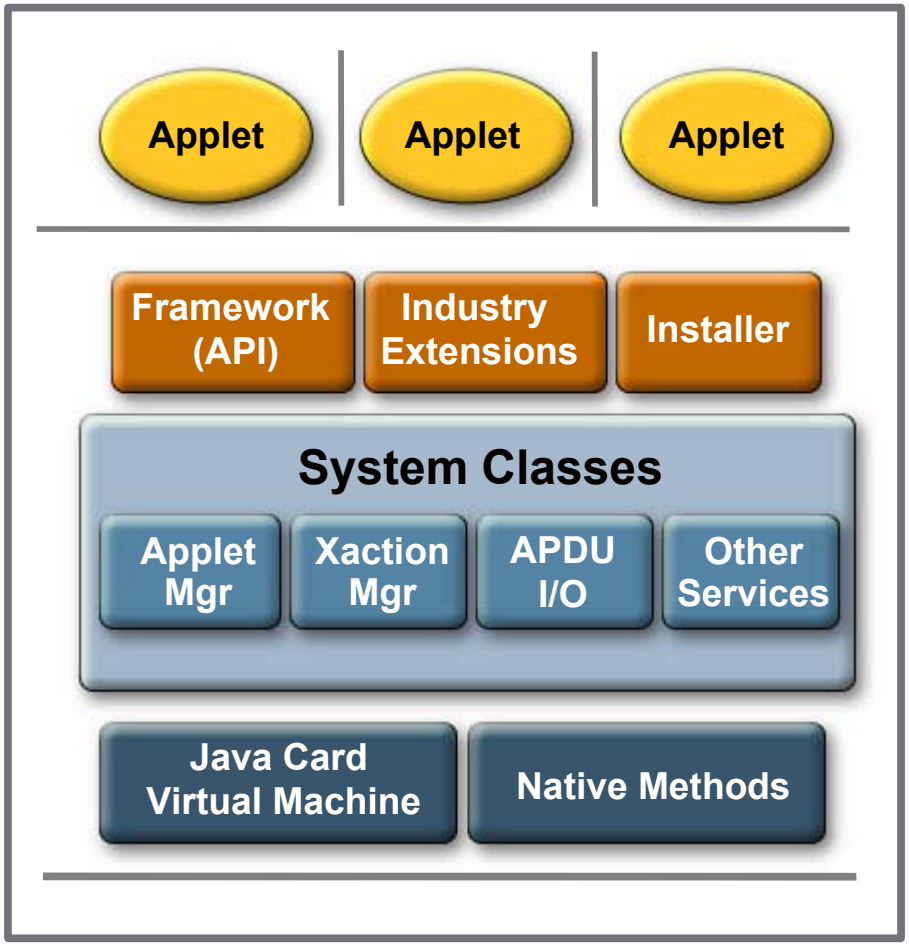
The new wave of mobile applications

Next Generation Java Card technology

- **Overview**
- **Architecture and new features**

Java Card Technology Today

- ***Java Card 2.2.2 Platform Specifications***
- ***Subset of Java™ Platform, Standard Edition (Java SE platform) and Java programming language***
- ***Split-VM Architecture***
- ***Persistent VM model***
- Firewall model isolates contexts and applets
- ISO7816 Communication I/F and protocol
- Transaction management



Drivers for Next Generation Java Card Technology

- SIM cards are in the process of developing from a multi applications smart card platform to a
  Mult-interface network connected secure token
- Technology drivers for a new Platform
  - Adoption of USB and TCP/IP over USB
  - Concurrent contact and contactless interface in parallel
  - Concurrent APDU and TCP/IP based communication
  - Http/Https driven communication for web content

Requirements for Next Generation Java Card Technology

- Support for the emerging usage patterns :
  - NFC, smart card web server
  - Support for new protocols and memory configurations
  - Concurrent support for multiple interfaces
- A development experience on par with Java ME platform
  - More capable and mainstream Java environment
  - Streamlined development tool integration
- Enhanced security features to support a more complex environment

Next-Generation Specifications

- Two stand-alone “Editions” of the Next-Gen Java Card technology specs
- Java Card Platform, **Connected Edition**
  - Includes all new network-oriented features
- Java Card Platform, **Classic Edition**
  - Leverages the existing Java Card 2.x platform architecture
  - For the more resource-constrained devices
- Both Editions are backward compatible with previous versions and share key security features

“Classic” Edition Features

- Traditional split VM
  - Resource efficient, 16-bit on-card VM
  - Off-card conversion for applet size optimization : CAP files
  - On-card or off-card byte code verification
  - On-demand Garbage Collection
- “Classic” Java Card APIs
  - Incremental extension of Java Card 2.2.2 platform framework
- APDU-based communication
  - Contact or contactless

“Connected” Edition Features

- Embedded web server with Java Servlet API support
  - Service static and dynamic content via HTTP(s)
- Multi-threaded environment
- Concurrent communication over USB, ISO, contactless
- Full backward compatibility
- Client and Server communication
- Leverage technology from Java ME platform/Java EE platform

Leveraged Java Technologies

- NG Java Card platform reuse existing Java platform building blocks
  - Proven security, tools, and developer community
- Java Card 2.x platform
  - APDU-based application model and card specific APIs
- Java ME technology
  - Connected Limited Device Configuration (CLDC)
  - Multi-threading, Strings, int, long, multi-dimension arrays
  - Generic Connection Framework, Security Model
- Java EE technologies
  - Java Servlet API for web application model

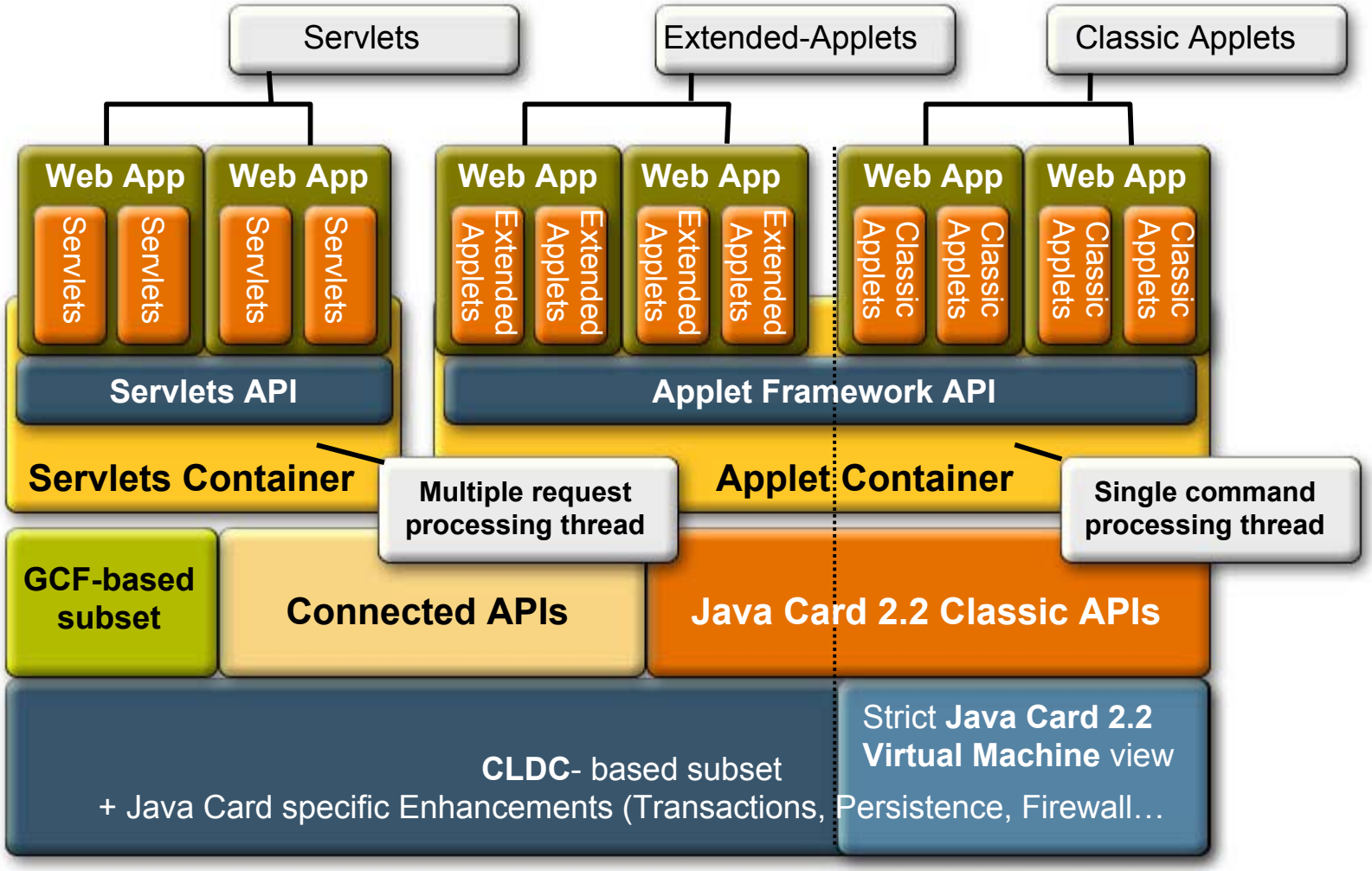
Agenda

The new wave of mobile applications

Next Generation Java Card technology

- **Overview**
- **Architecture and new features**

Next Generation "Connected" Java Card Platform



Next-Generation "Connected" Java Card Platform Features

- Virtual Machine Technology
- Security
- Network-oriented Communication
- Enhanced Programming Model

Virtual Machine

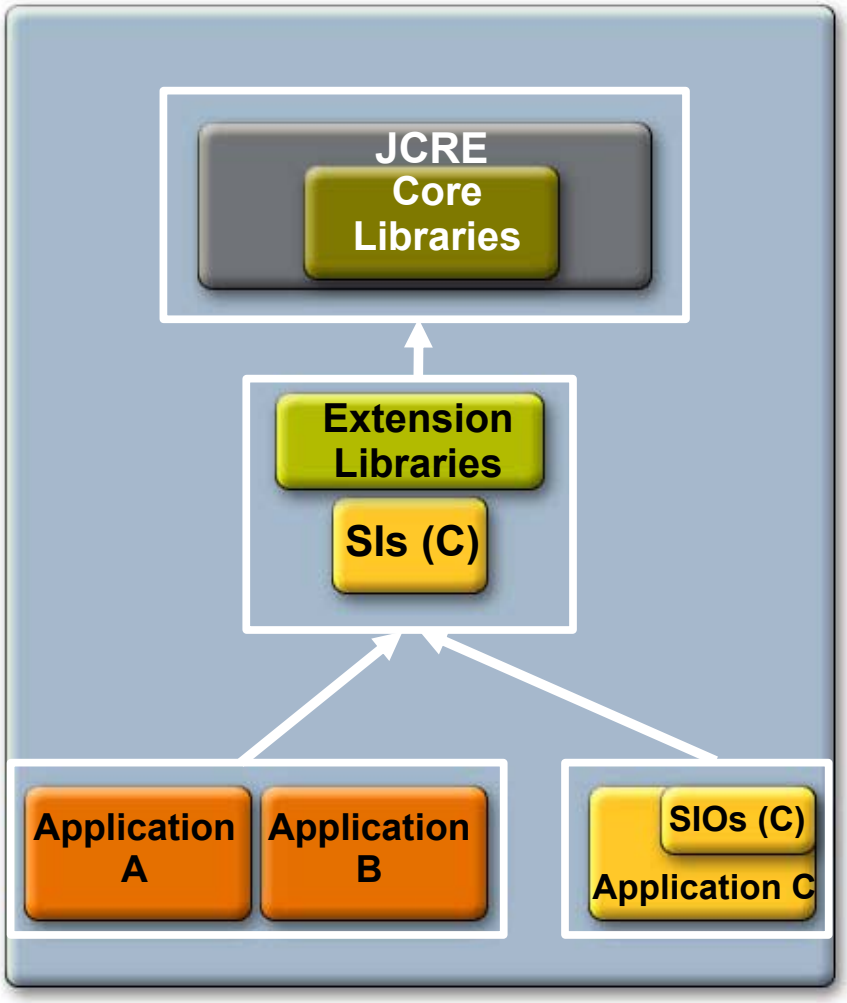
- 32-bit VM
- .class File Loading
- Concurrent execution of applications (Multi-threading)
- On-card bytecode verification
- Automatic Garbage Collection

Security—Overview

- Code isolation
- Context isolation
- Access control
- Authentication
- Enhanced shareable interface mechanism

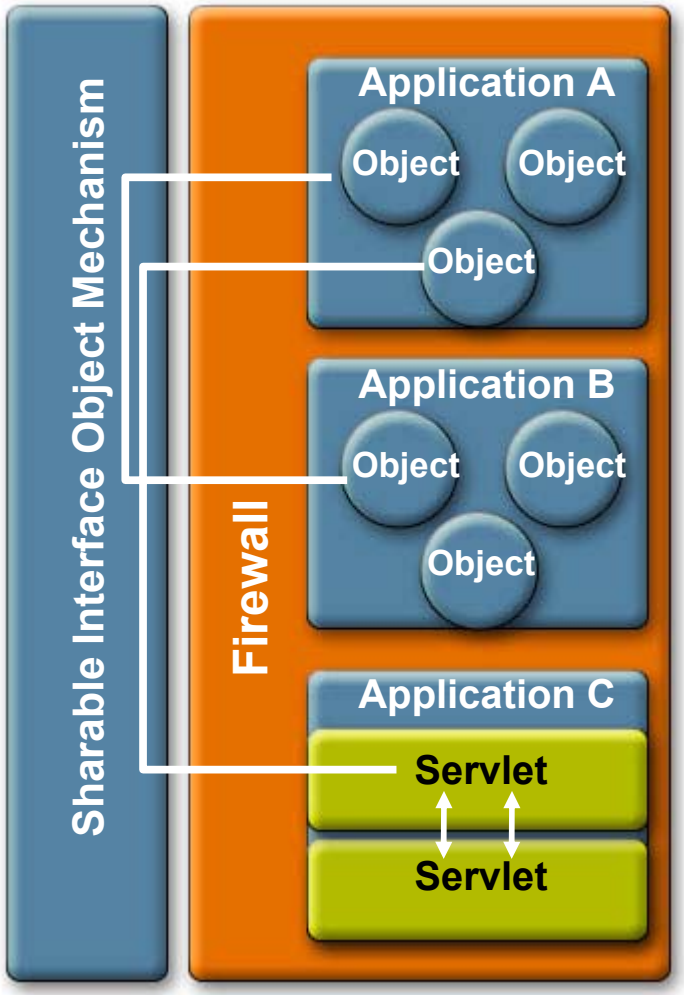
Security—Code Isolation

- Similar to Java SE class Loaders
- Uses 3 principles of Java SE class loader
  - Delegation Principle
    - Delegation to parent first
  - Visibility Principle
    - Classes loaded by parents visible to children but not vice versa
  - Uniqueness Principle
    - When a class is already loaded in the class lookup table path, it is not reloaded



Security—Context Isolation

- Enforced by firewall
- Object Ownership
  - Every object owned by the creator object's context
- Object Access
  - Only allowed from same context
- Context Switching
  - Same as Java Card 2.2.x. platform



Security—Access Control

- Policy based fine grained access control
- Leverages a subset of Java SE platform security model framework
- Protection Domain used to define a set of permissions granted to applications in the domain
- Leverages the firewall (Context Isolation)
- Per application Declarative Security
  - Allows server application to designate the clients it allows to call its service
- Defined by application developer and configured by deployer/application provider

Security—Authentication

- Card holder authentication
  - Global
  - Session
- Remote administrator authentication
- Container managed authentication

Network Oriented Communication

- ISO 7816 and TCP/IP communication support
- Communication over USB, MMC
- Concurrent contact/contact-less card access
- Embedded web server
- Service static and dynamic content through HTTP(s)
- Client and server communication mode
- Generic communication API

Enhanced Programming Model (1)

- Fully backward compatible
- Support additional Java platform language types
  - Char, long... and String
  - Multi-dimensional arrays
- Support for large data structures—Multimedia content
- Application models
  - Classic APDU-based applet model
  - HTTP Servlet-like model

Enhanced Programming Model (2)

- Enhanced inter-application communication framework
- Generic event framework
- Evolutive cryptography

Summary

- More complex card applications and deployment schemes
- Advancements in smart card hardware need corresponding changes to the Java Card platform
- Next-Generation Java Card Platform:
  - Fully backward compatible
  - Enhanced VM
  - Advanced security features
  - Network-oriented communication support
  - Advanced Programming Model

For More Information

- Java Card technology booth in the exhibition hall
- TS-5203: Web 2.0 Applications on a Next-Generation Java Card Platform
- TS-0285: JavaCard for Emerging WLAN Environments
- TS-5147: Free Mobile-to-Mobile Money Transmission
- TS-5642: What to do with APDU?
- BOF-0396: Internet Application Use Cases of Next-Generation Java Card Technology
- BOF-5593: Contract Enforcement for Embedded Java Technology Programs

Q&A

Florian Tournier

Eric Vétillard

Saqib Ahmad

Next-Generation Java Card Technology for Secure Mobile Applications

Eric Vétillard

CTO
Trusted Labs
www.trusted-labs.com

Saqib Ahmad
Java Card Engineer

Florian Tournier
Group Marketing Manager

Sun Microsystems
www.sun.com

TS-5686