Smart Card Application Development Using the Java Card Technology

Milan Fort

27.01.2006

SeWeS 2006

Agenda

- Overview of Smart Cards
- Introduction to Java Card Technology
- Developing a Java Card Applet
- Summary
- Q&A

Agenda

- Overview of Smart Cards
- Introduction to Java Card Technology
- Developing a Java Card Applet
- Summary
- Q&A

Smart Card Overview

- Small, plastic card with embedded integrated circuitry
- Same size as magnetic stripe card
- Portable, tamper-proof computer
- High level of security
- Physical and electronic characteristics defined by ISO 7816
- Contact or contactless communication
- No internal power source

Memory Cards vs. Microprocessor Cards

Memory Cards

- Most common type
- Contain only memory chip
- Optionally with protected memory access
- Main advantage: low cost
- Areas used: prepaid phone cards, etc.

Microprocessor Cards

- Contain a microprocessor
- Tamper-proof
- More expensive
- Areas used: financial cards, electronic purses, access control, etc.

Smart Card Memory Types

- ROM
  - Read Only Memory
  - Persistent and nonmutable
- EEPROM
  - Electrical Erasable Programmable Read Only Memory
  - Persistent and mutable
- RAM
  - Random Access Memory
  - Nonpersistent and mutable

Typical Smart Card Hardware

- 8-32 bit CPU
- 2 kB RAM
- 32-64 kB ROM
- 8-32 kB EEPROM
- External Power: 5V
- External Clock: 1-5 Mhz
- Half duplex serial I/O: 420 Kbps
- Crypto Coprocessor

Contact Cards vs. Contactless Cards

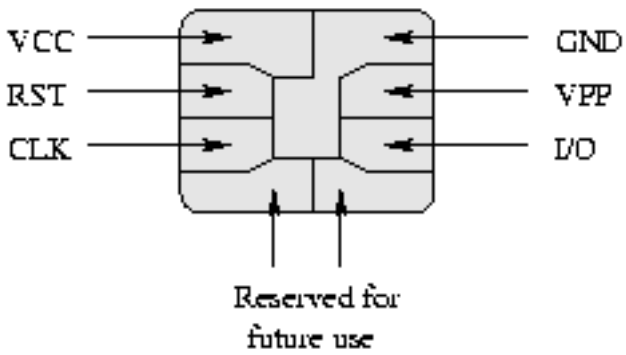
Contact Cards

- Most common type
- Require insertion into the reader
- Have 8 gold plated contacts
- Disadvantages: can get worn or damaged

Contactless Cards

- No insertion required
- Data/Power transfer via RF
- Used when only limited amount of data has to be exchanged
- Advantages: higher reliability, longer lifetime
- Disadvantages: more expensive, not suitable when large amount of data has to be transferred
- Usage: transport systems, access control

Smart Card Contact Points



- VCC – power supply
- RST – reset signal
- CLK – clock signal
- GND – reference voltage
- VPP – write voltage
- I/O – data transfer

Card Acceptance Device

Smart Card Readers

- Basic connector between PC and smart card
- No intelligence to process transmitted data
- Attached to serial, parallel, or USB port
- Optionally equipped with display and PIN-pad

Smart Card Terminals

- Small computer on its own
- Integrates smart card reader as one of its components
- Usually has also a small display, keypad and printer

Smart Card Communication Model

- Half-duplex, master-slave model
- Application Protocol Data Unit (APDU)
  - Top level protocol
  - Specified in ISO 7816-4
  - Defines two types of messages
    - Command APDU
    - Response APDU
- Transmission Protocol Data Unit (TPDU)
  - Specified in ISO 7813-3
  - Transmits APDUs
  - Two common variations:
    - T=0 (byte oriented)
    - T=1 (block oriented)
- Answer to Reset (ATR)
  - Byte sequence returned by the card to the reader on power-on

Command APDU

- CLA – Class of instruction
- INS – Instruction code
- P1, P1 – Parameters
- Lc – Length of the optional data
- Le – Expected length of data returned

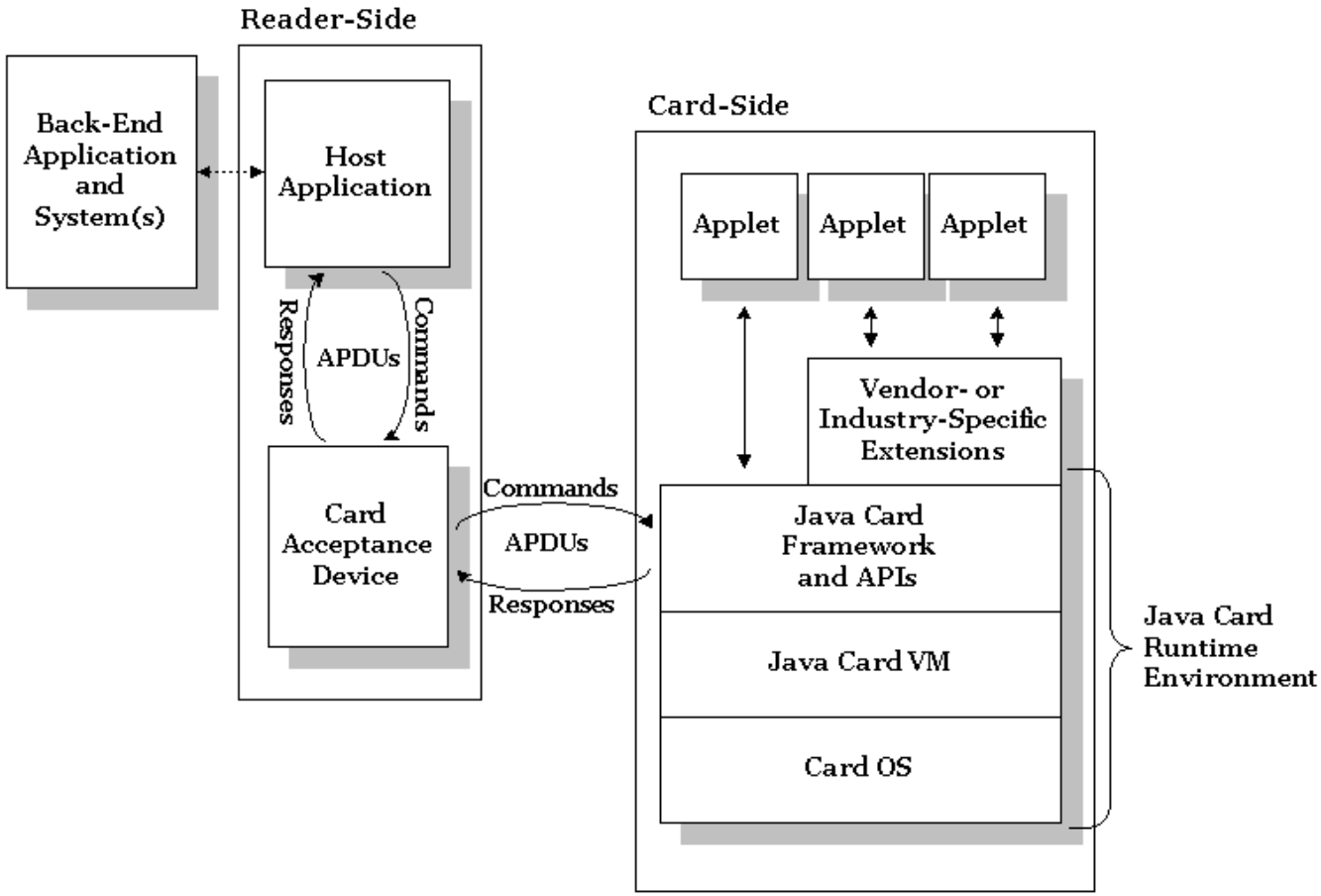
Header (required)				Body (optional)		
CLA	INS	P1	P2	Lc	Data Field	Le

Response APDU

- Optional data – sent only if Le was specified in Command APDU
- SW1, SW2 – two status word bytes containing status information

Body (optional)	Trailer (required)	
Data Field	SW1	SW2

Smart Card System Development



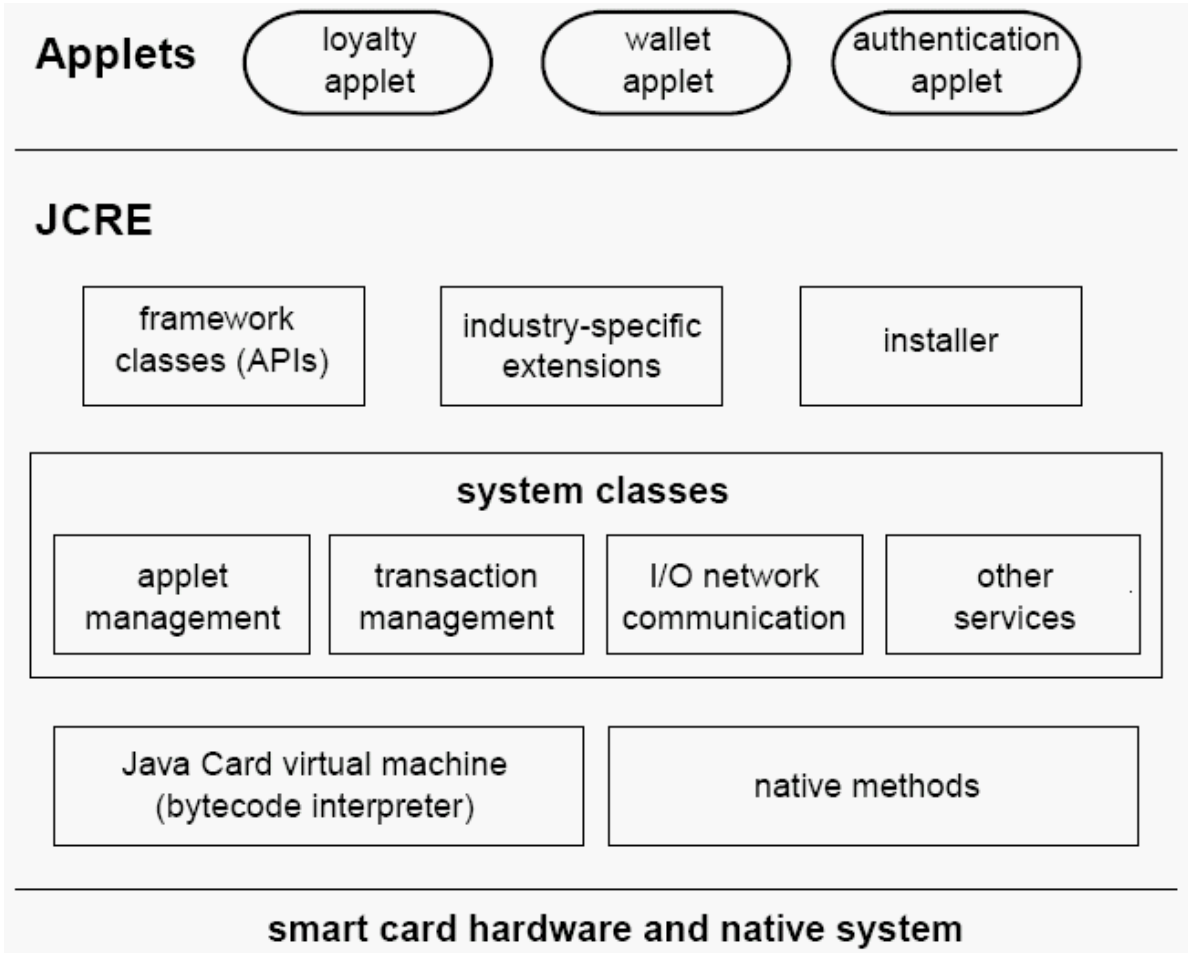
Agenda

- Overview of Smart Cards
- Introduction to Java Card Technology
- Developing a Java Card Applet
- Summary
- Q&A

Java Card Technology

- Subset of Java SE platform and Java programming language for smart cards
- Brings smart card application development into mainstream
- Enables multiple applications from different vendors to run on the same card
- 1 Billion cards deployed
- Three specifications (currently in version 2.2.1):
  - Java Card Virtual Machine specification
  - Java Card Runtime Environment specification
  - Java Card API specification

Java Card Technology



Java Card Language Subset

Supported Java Features

- Small primitive data types: `boolean`, `byte`, `short`
- One-dimensional arrays
- Java packages, classes, interfaces, and exceptions
- Java object-oriented features: inheritance, virtual methods, overloading and dynamic object creation, access scope, and binding rules
- The `int` keyword and 32-bit integer data type support are optional

Unsupported Java Features

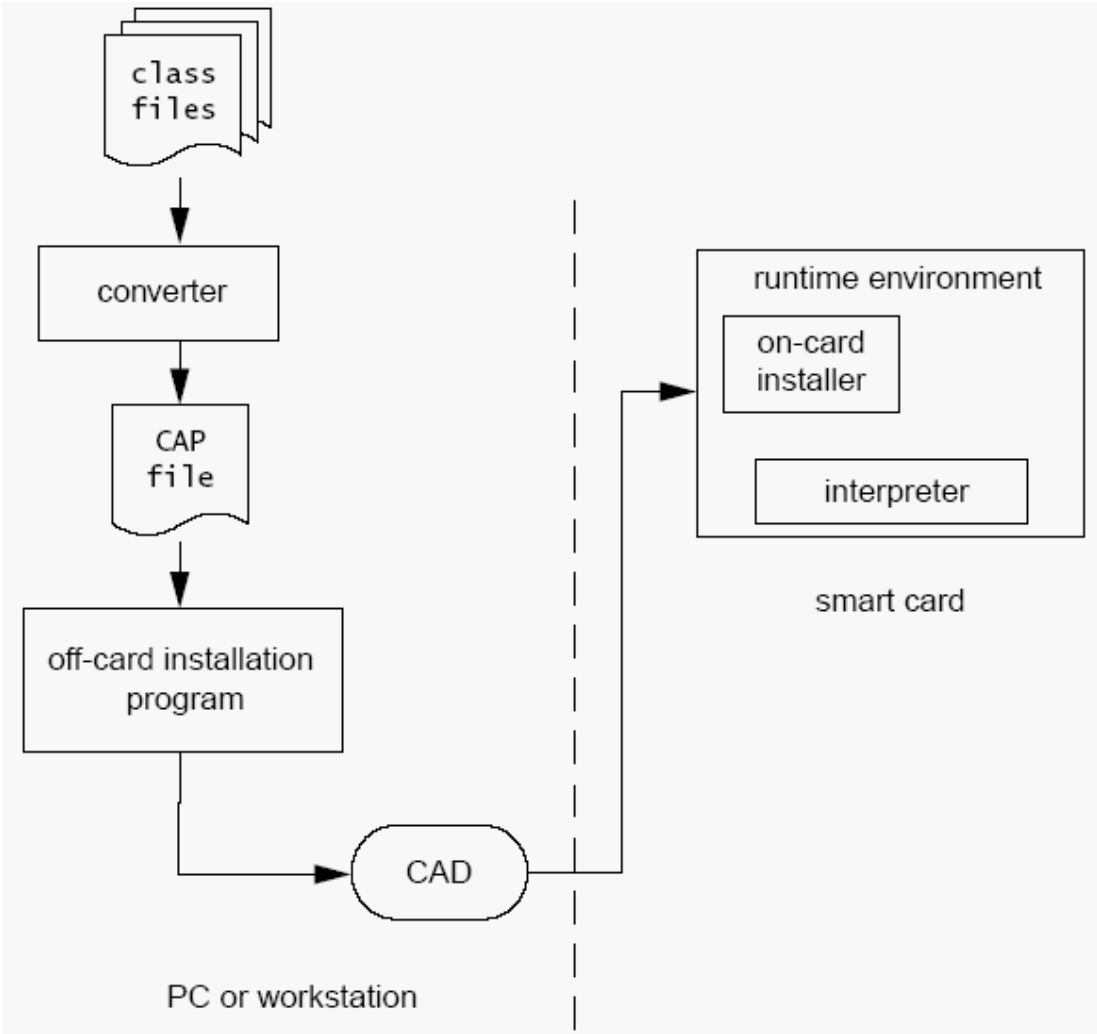
- Large primitive data types: `long`, `double`, `float`
- Characters and strings
- Multidimensional arrays
- Dynamic class loading
- Security manager
- Garbage collection and finalization
- Threads
- Object serialization
- Object cloning

Java Card Virtual Machine

Split Architecture

- Off-card converter
  - Checks compliance with Java Card VM specification
  - Performs security checks
  - Optimizes bytecode
  - Initializes static variables
  - Outputs Converted Applet (CAP) file
- On-card installer
  - Communicates with the off-card installation program
  - Writes the CAP file into smart card memory
  - Links it with other classes that are already on the card
- On-card interpreter
  - Executes code found in the CAP file

Applet Installation Process



Java Card API

- `java.lang`
- `java.rmi`
- `java.io`
- `javacard.framework`
- `javacard.framework.service`
- `javacard.security`
- `javacardx.crypto`
- `javacardx.rmi`

Java Card Runtime Environment

- Initialized at card initialization time
- Responsible for resource management, network communication, applet execution, on-card system and applet security enforcement
- Special features include:
  - Persistent and transient objects
  - Atomic operations and transactions
  - Applet firewall and the sharing mechanisms

Agenda

- Overview of Smart Cards
- Introduction to Java Card Technology
- Developing a Java Card Applet
- Summary
- Q&A

Java Card Applet Development

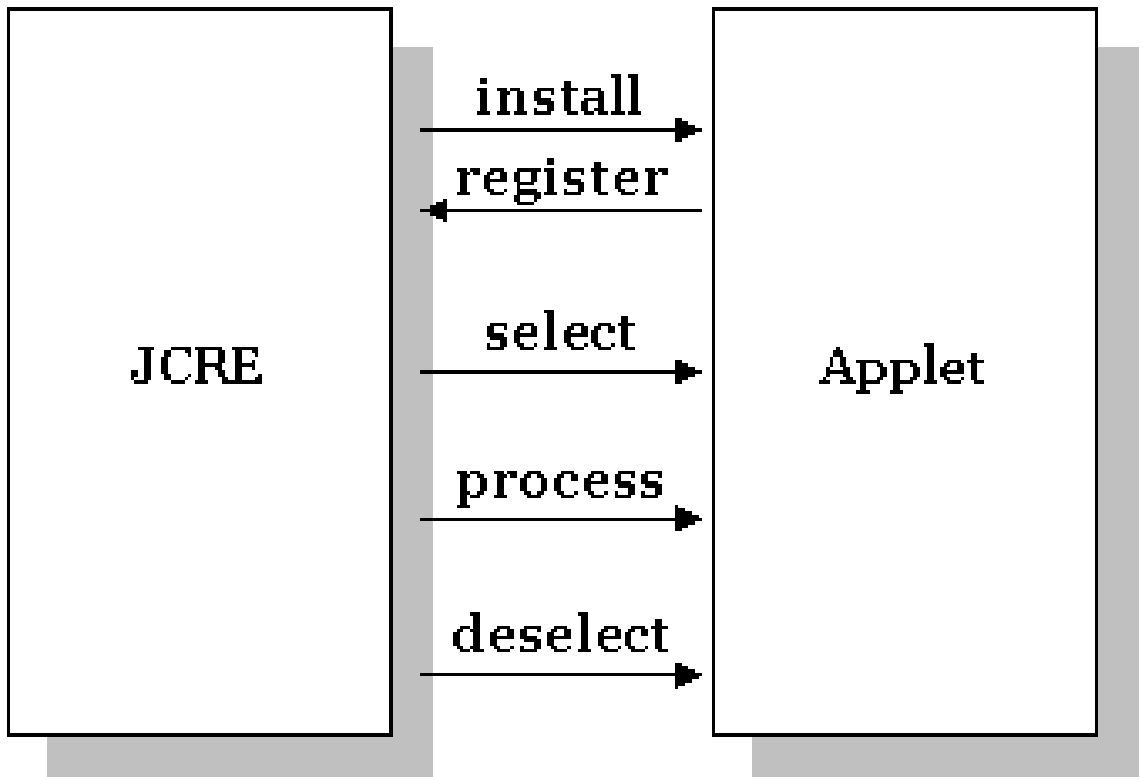
Two different programming models

- Message-passing model
  - Designed around the APDU protocol
  - Set of APDU instructions is the interface between the applet and the host application
- Java Card Remote Method Invocation
  - Subset of Java SE RMI
  - Provides distributed object model mechanism on top of APDU-based messaging model

Typical Java Card Applet Structure

```
import javacard.framework.*;
...
public class MyApplet extends Applet {
    // Definitions of APDU-related
    // constants
    ...
    // Constructor
    MyApplet() {...}
    // Life-cycle methods
    install() {...}
    select() {...}
    deselect() {...}
    process() {...}
    // Private methods
    ...
}
```

Life-cycle of a Java Card Applet



Applet Methods

- `install()`
  - Called by the card installer when it installs the a new applet on the card
  - Must instantiate the applet
  - Must call the `register()` method to notify the JCRE that a new applet has been instantiated

Applet Methods

- `select()`
  - Invoked by the JCRE to notify the applet that it has been selected for APDU processing
- `deselect()`
  - Invoked by the JCRE to notify the applet that has been deselected, before another applet gets selected
  - Used for session cleanup
  - Is not guaranteed to be called

Applet Methods

- `process()`
  - Every time an APDU is received and an applet is selected, JCRE invokes its process method, passing it the incoming APDU as parameter
  - Applet then takes appropriate actions and generates and sends back response data or throws an exception
  - JCRE sends back any data received from applet together with appropriate status word

Agenda

- Overview of Smart Cards
- Introduction to Java Card Technology
- Developing a Java Card Applet
- Summary
- Q&A

Summary

- Smart cards represent nowadays the most portable and secure computing platform available
- Java Card technology brings smart card application development into mainstream while preserving smart card security

References

1. Z. Chen. *Java Card Technology for Smart Cards*. Addison-Wesley Professional, 1st edition, 2000.

2. C. E. Ortiz. *An Introduction to Java Card Technology - Part 1*. Sun Developer Network, 2003.
   http://developers.sun.com/techtopics/mobility/javacard/articles/javacard1/

3. C. E. Ortiz. *An Introduction to Java Card Technology - Part 2, The Java Card Applet*. Sun Developer Network, 2003.
   http://developers.sun.com/techtopics/mobility/javacard/articles/javacard2/

Agenda

- Overview of Smart Cards
- Introduction to Java Card Technology
- Developing a Java Card Applet
- Summary
- Q&A

Q&A