ORACLE

JAVA CARD Introduction

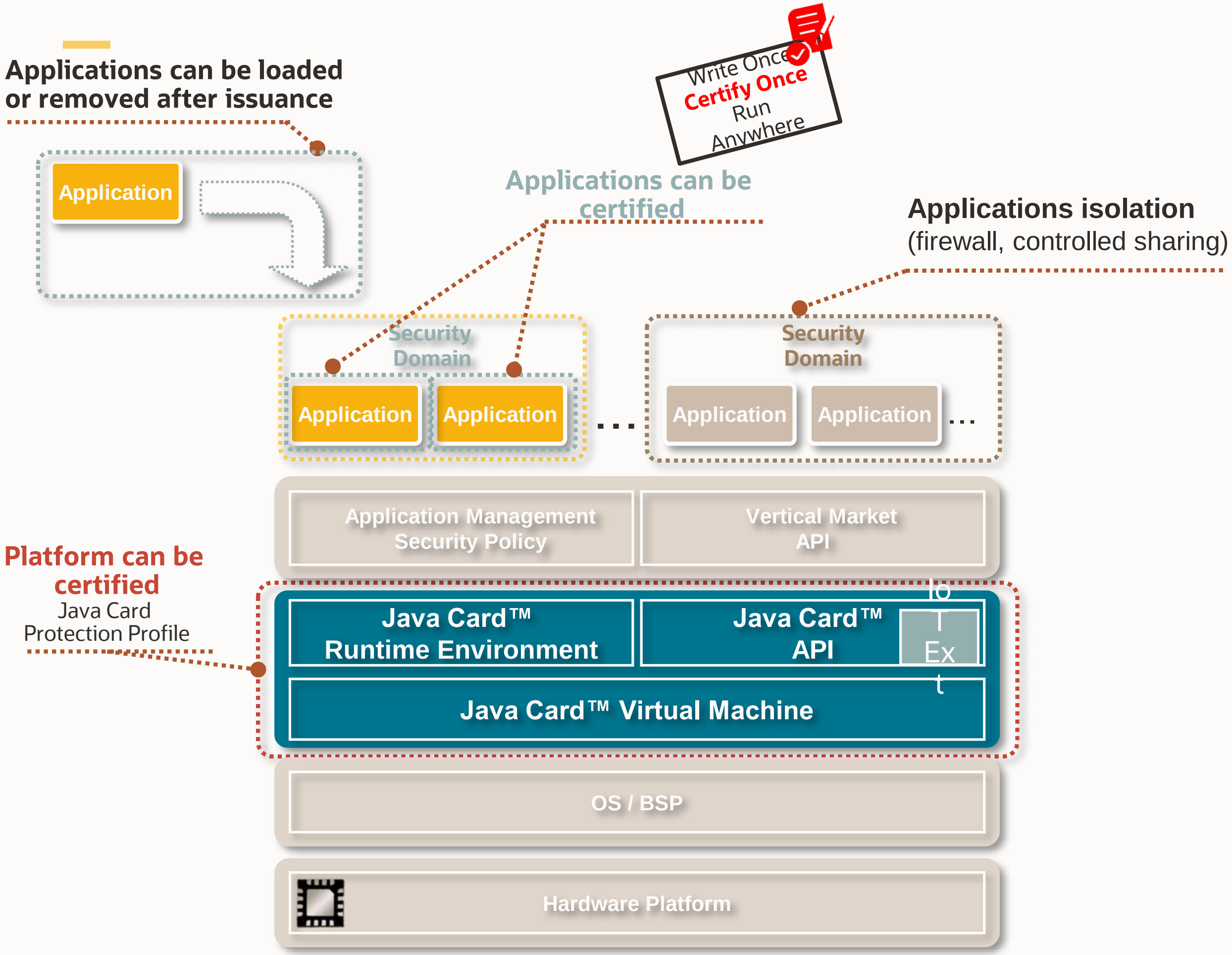
Saqib Ahmad
Product Manager, Java Card
Java Platform Group

April, 2021

Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Java Card Platform


Applications can be loaded or removed after issuance
Application
Write Once
Certify Once
Run
Anywhere
Applications can be certified
Applications isolation
(firewall, controlled sharing)
Security Domain
Application
Application
...
Security Domain
Application
Application
...
Application Management Security Policy
Vertical Market API
Platform can be certified
Java Card Protection Profile
Java Card™ Runtime Environment
Java Card™ API
IoT Ext
Java Card™ Virtual Machine
OS / BSP
Hardware Platform

COMPACT VIRTUAL MACHINE

Low footprint Split VM.
Hardware-agnostic Content.

CERTIFIABLE DESIGN

Products certified at Common Criteria EAL 5 and above.
Protection Profile available.

OPEN PLATFORM

Public specification & SDK
Community support through Oracle and Java Card forum
Multi applications

COMPLIANCE

TCK Enabling compatibility across products and implementations.
Align with standards (GlobalPlatform, ETSI, 3GPP, GSMA, ISO…)

APPLICATION FIREWALL

Allowing Secure Multi-Application and Multi-Tenancy with low memory consumption.

IoT EXTENSIONS

Introduced in Version 3.1
Security Service APIs.
GPIO, SPI, ISO support.

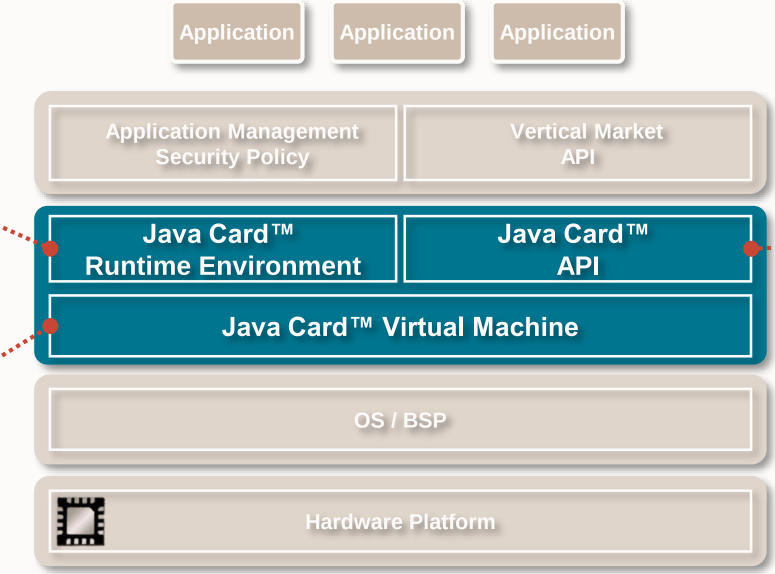
Java Card Platform
Role of the Java Card platform components

JC Runtime Environment

- Defines runtime behavior
  - Application model and lifecycle
  - Memory model
- Enforces security model
  - Isolation of applications and sharing

JC Virtual Machine

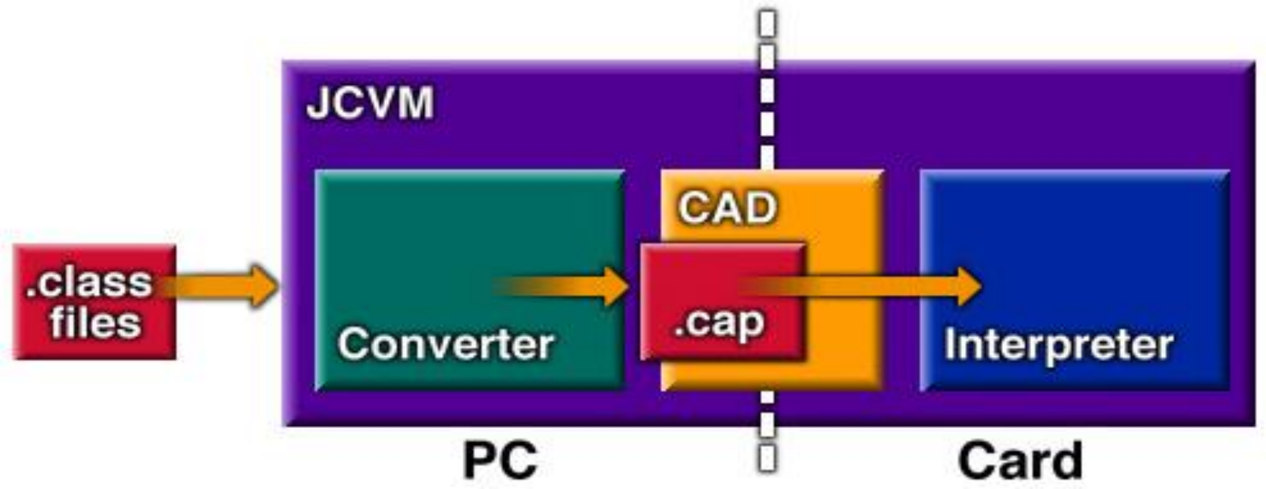
- Loads code
- Executes code
- Controls the access to resources


Application
Application
Application
Application Management Security Policy
Vertical Market API
Java Card™ Runtime Environment
Java Card™ API
Java Card™ Virtual Machine
OS / BSP
Hardware Platform

JC Application Programming Interface

- Defines Application framework
- Provide I/O communication means
- Cryptography and security framework
  - Symmetric and asymmetric crypto
  - Keys, PIN codes, certificates, …
  - Biometry

Java Card VM Technology



- Off-card
  - Class loading, linking and name resolution
  - Bytecode verification, optimization and conversion
- On-card
  - Bytecode execution and security enforcement

- CAP (.cap) File
  - Contain executable code and can be downloaded and installed onto a Java Card enabled device
  - Output of the Converter tool
  - Verified off-card by the off-card verifier tool
- Export (.exp) File
  - Public façade of a package in a CAP file
    - Contains public API information
  - Used by the converter tool for linking
  - Used by the verifier tool for verification

Java Card VM and Applet Lifetime

- The Java Card VM runs forever.
- VM only stops temporarily when power is removed
- VM starts again and recovers its previous object heap from persistent storage when the card is next powered up.
- Applet’s life starts when it is properly installed and registered with the system registry
- Applet’s life ends when it is removed from the system registry

Memory Model

- ROM
- Persistent Memory
  - Flash/EEPROM
  - All objects are by default created in persistent memory
  - All persistent objects persist across CAD sessions
  - Transaction mechanism for atomicity
- Transient Memory RAM
  - Partitioned into Clear-On-Deselect, Clear-On-Reset and Stack space.
  - Transient objects have their headers in persistent memory and data in RAM
  - Not transacted

Java Card Platform Java Language Subset

Supported

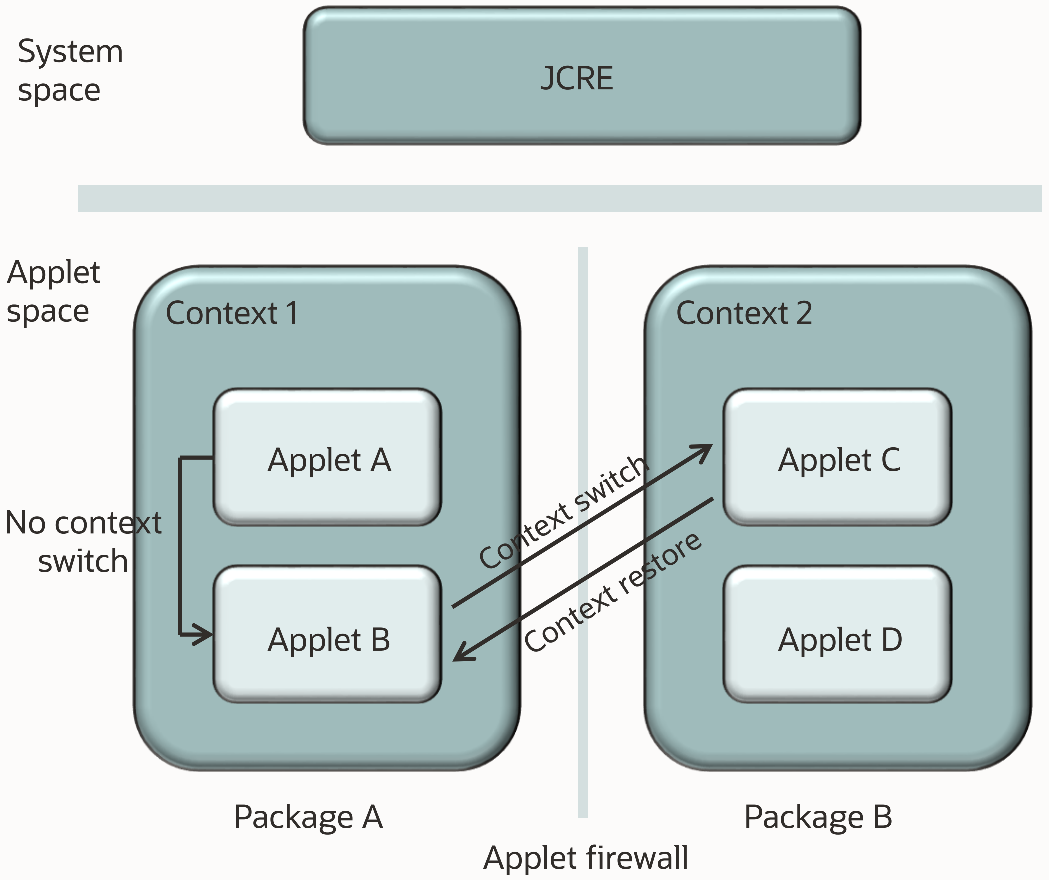
- Small primitive data types: boolean, byte, short, int (optional)
- One-dimensional arrays
- Objects
- Packages, classes, interfaces, and exceptions
- Java object-oriented features such as inheritance, virtual methods etc.
- Optional object deletion

Unsupported

- Large primitive data types: long, double, float
- Characters and strings
- Multidimensional arrays
- Dynamic Class Loading
- Security Manager
- Finalization
- Threads
- Cloning
- Varargs
- Keywords (native, synchronized, transient, volatile, strictfp, enum, assert)

Application Isolation (Firewall)

- Firewalls partition the Java Card platform's object system into separate protected object spaces called contexts
- Object access only allowed within the same context
- Access across firewall (different context) allowed through special mechanisms.



Comprehensive API

Application framework

`java.lang, java.io, javacard.framework`

- Application lifecycle
- I/O protocols - ISO 7816 based protocols
- Memory and transaction management, Sharing

Cryptographic framework

`javacard.security, javacardx.crypto`

- Random number generation
- Message Digest
- Symmetric & Asymmetric cryptography for Encryption, Decryption, Signature, Verification
  - AES, SM4, HMAC, multiple modes (ECB, CBC, CFB, CTR, XTS) and multiple paddings
  - RSA, DSA, Elliptic Curves (Brainpool, SECP, curve25519, curve448, FRP256v1, SM2)
- Key Agreement (DH, XDH) and Key Generation (RSA, DSA, ECC)

Security framework

`javacard.security, javacardx.security`

- Keys and PIN codes management
- Integrity and CRC
- Security assertions

Biometry

`javacardx.biometry, javacardx.biometry1toN`

- Enrollment of biometric templates and verification of biometric data

Big numbers operations

`javacardx.framework.math`

- Arithmetic operations on big numbers

ASN.1 TLV structures handling

`javacardx.framework.tlv`

- Parsing of BER TLV structures

System Time management

`javacardx.framework.time`

- Manage system uptime and perform operations on time durations

Certificate management

`javacardx.security.cert`

- Parsing and storage of X.509 certificates

Pseudo Random Functions and Key Derivation Functions

`javacardx.security.derivation`

- KDF schemes (NIST SP800-108, ANSI X9.63, ICAO, IEEE1363) and PRF (TLS 1.1 and 1.2)

Monotonic Counter

`javacardx.security.util`

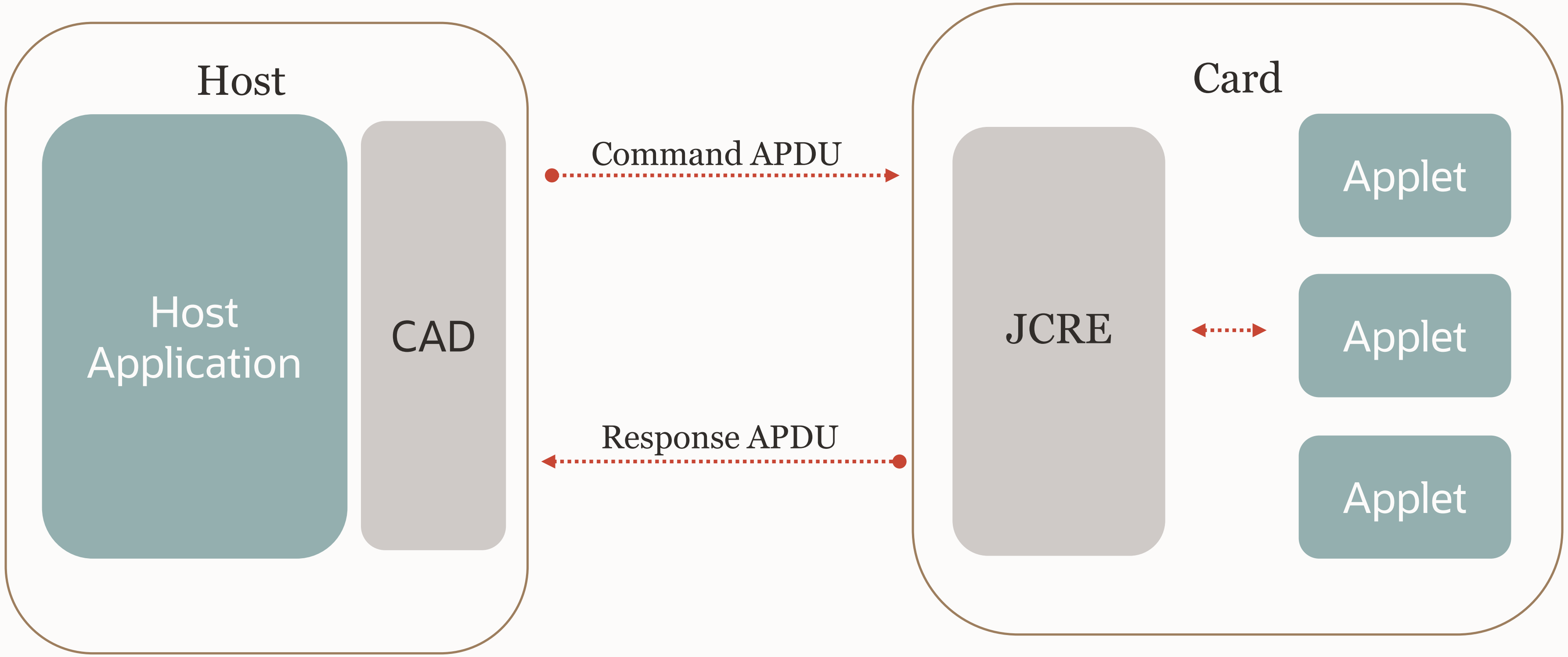
- Secure implementation of monotonic counters for anti-replay

Extended I/O

`javacardx.framework.nio, javacardx.framework.event, javacardx.external`

- Event framework and I/O buffer management

APDU Based Communication



Jan 2019 Release

Java Card 3.1 – Key Features

Java Card Platform

- Large Application support *
- Simplified Library Deployment *
- Extensible I/O Model *
- Security Services APIs *
- New Crypto Extensions *
- Static Resources for Applications
- Optimized Array Management
- Improved API Upgrade Mechanism
- Backward Compatibility

** Optional Features*

Java Card Development Kit

- Implements 3.1 Specification
- New tools / emulator packaging
- Documentation Improvements

Java Card TCK

- Compliance with 3.1 Specification
- Continued test coverage enhancements

Building a Java Card Solution

JAVA CARD **PLATFORM**

Oracle delivers Java Card Platform specifications, test suites and reference implementation code for the Java Card runtime and APIs.

Java Card Specification

Test Suites & Reference Implementation

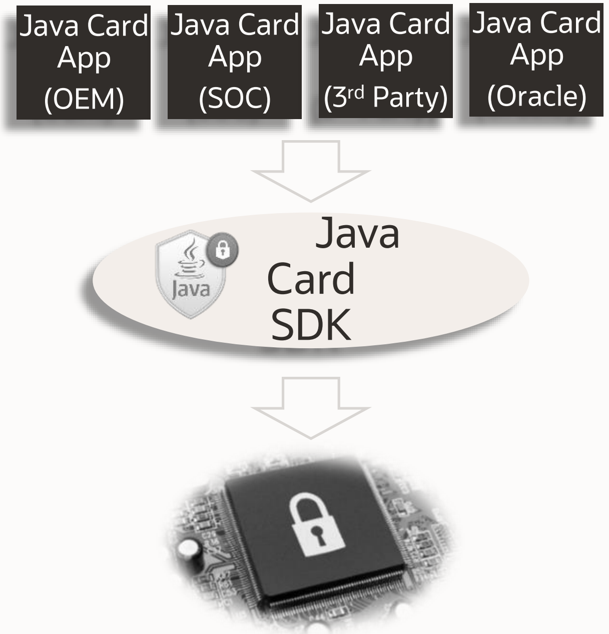
Protection Profile

JAVA CARD **PRODUCTS**

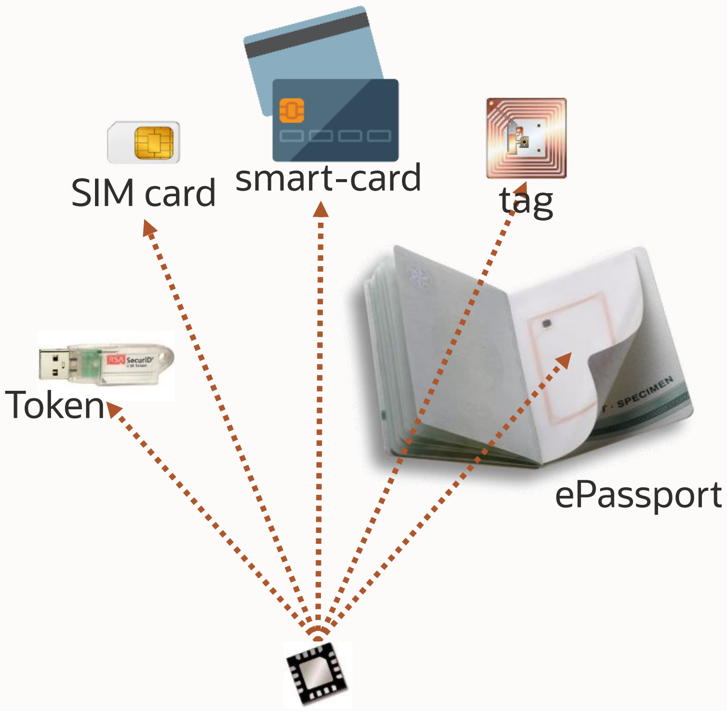
Java Card licensees implement the specifications in software or hardware products, certify and sell to end users eg device OEMs, IoT solutions vendors, MNOs.

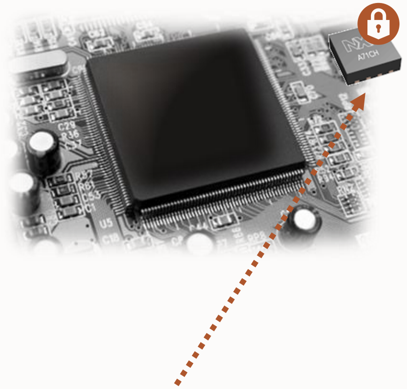
JAVA CARD **CONTENT**

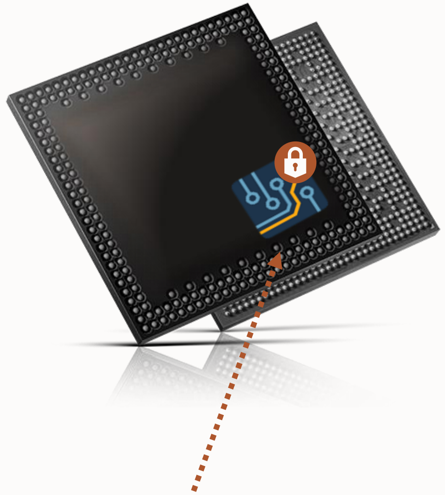
Security Service Providers develop applications against the specifications, using the Oracle SDK or 3rd party tools, and deploy on Java Card Products.



Typical Secure Hardware





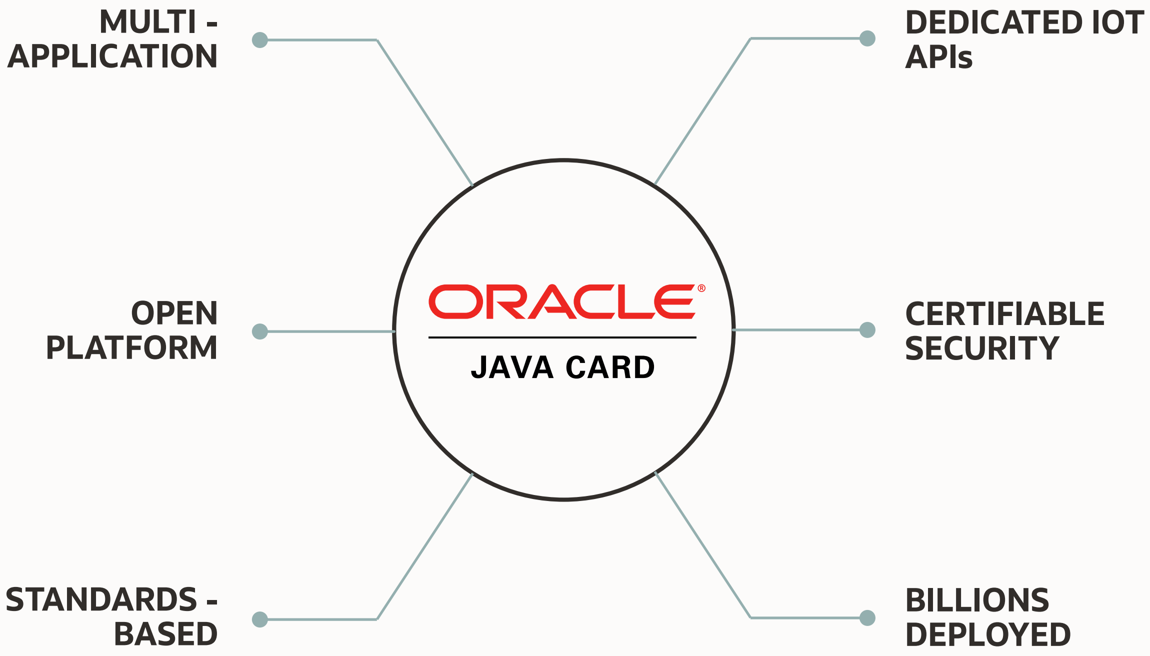


Removable Secure Element
standalone secure microcontroller
plugged into host device

Embedded Secure Element
separate chip
soldered in host device

Integrated Secure Element
part of the design of a chip

Java Card in IoT



MULTI - APPLICATION

DEDICATED IOT APIs

OPEN PLATFORM

ORACLE JAVA CARD

CERTIFIABLE SECURITY

STANDARDS - BASED

BILLIONS DEPLOYED

A Secure Application Platform for IoT Devices and Solutions

Key Value Proposition

EDGE SECURITY ON ANY HARDWARE

IoT Security Services with a choice of hardware options

CONNECTIVITY + SECURITY

Security and SIM applications into one security device

PROGRAMMABLE AND EXTENSIBLE

New IoT services can be added using standard—based Java tools

RENEWABLE SECURITY

Updatability ensures always-current device security

More Information

https://www.oracle.com/technetwork/java/javacard

Java Card Platform Specification 3.1

Latest release of the Java Card specification and the reference for Java Card products.

Java Card Development Kit Tools

The Java Card Development Kit Tools are used to convert and verify Java Card applications. The Tools can be used with products based on version 3.1, 3.0.5 and 3.0.4 of the Java Card Specifications.

Java Card Development Kit Simulator

The Java Card Development Kit Simulator includes a simulation component and Eclipse plug-in.
Combined with the Java Card Development Kit Tools, it provides a complete, stand-alone development environment.

Java Card IoT and Security blog

This Blog covers the latest Java technology for small devices and security in the IoT, mobile, ID and Payment.

Webcast – Secure Business Runs Java Card

Webcast – How to secure IoT Edge with Java Card

Webcast – Oracle Java Card 3.1 Boosts Security for IoT devices at the Edge

Thank You

Q&A

ORACLE