



ORACLE

JAVA CARD

IoT Connectivity & Security

Nicolas Ponsini
Security Solutions Architect
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.



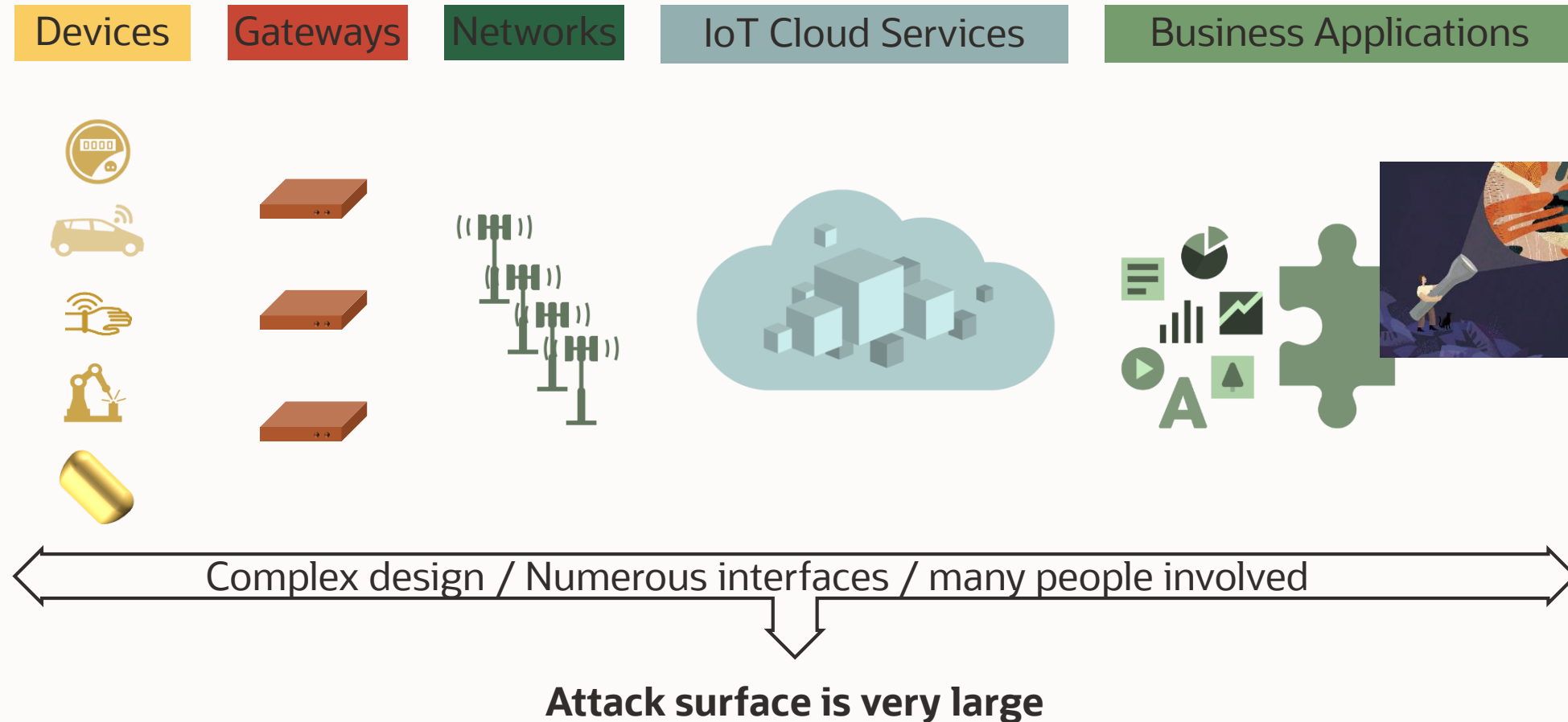
Program Agenda

- 1 Taking Risks out of IoT Security with Java Card
- 2 Java Card and Flexible IoT security



Taking Risks out of IoT Security with Java Card

IoT Architecture



Security at the Device Edge

A complex Risk Analysis

Vulnerabilities – Threats - Asset values - Exposure



Mitigation: Security / Cost Equation ?



Massive - Long Lifecycle – Exposed Environment – Physical Access

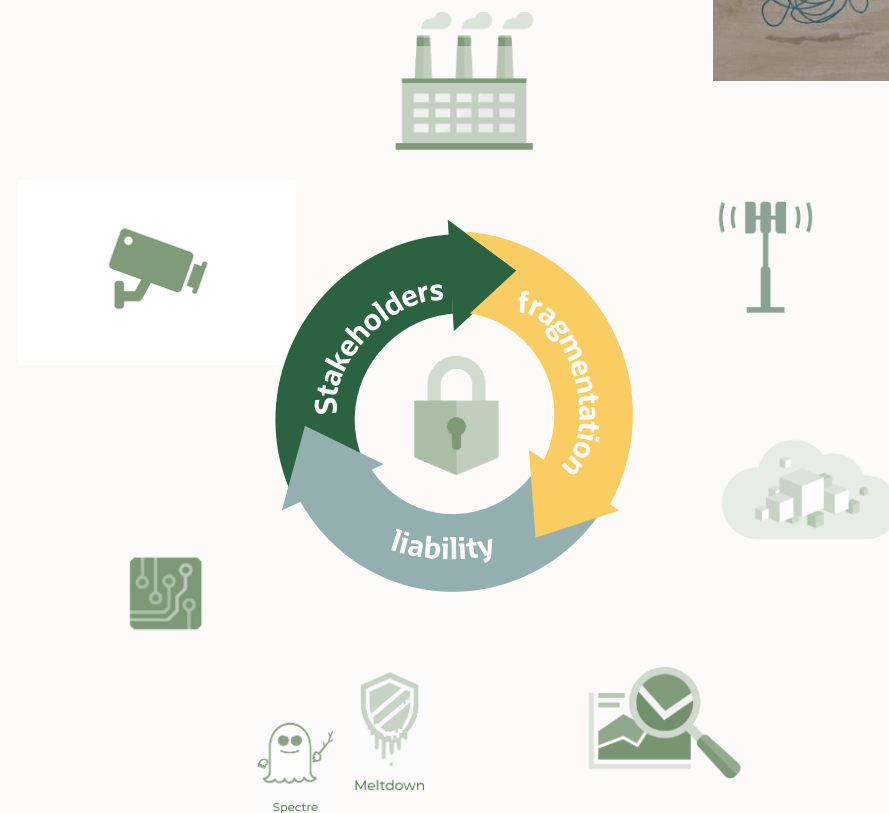
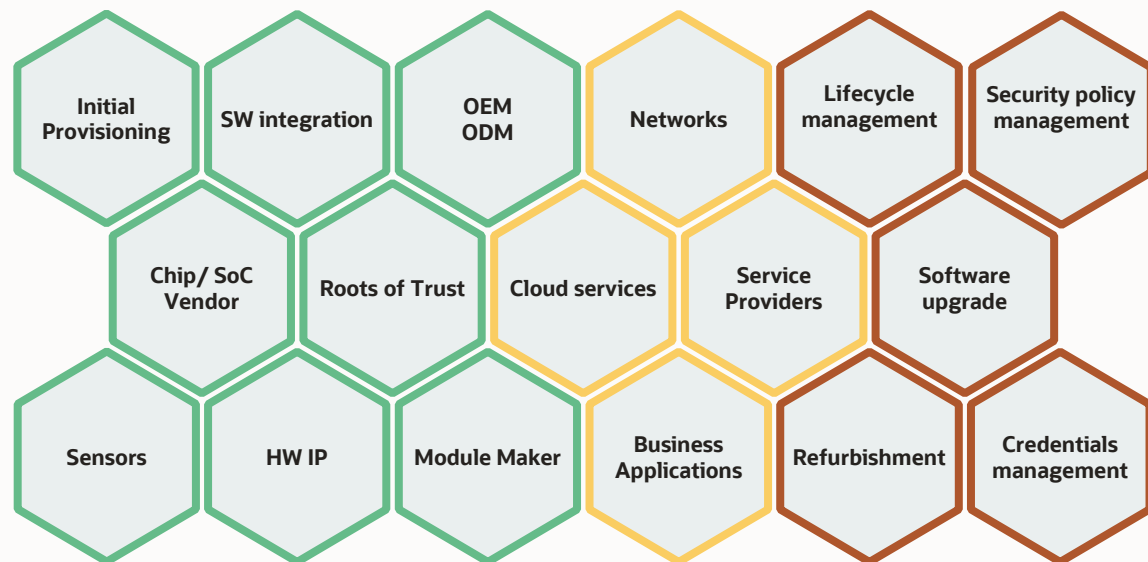
Security at the Device Edge

A complex ecosystem

Build

Integrate

Sustain

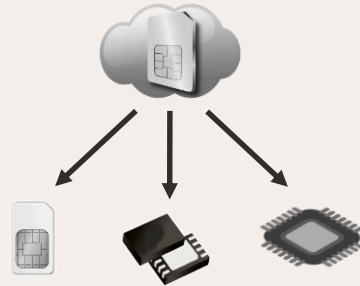


Use-cases in IoT security market



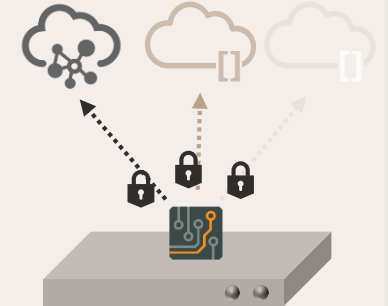
DIGITIZED SIM

Abstract the underlying hardware for portability of the SIM applications across multiple hardware at lower cost.



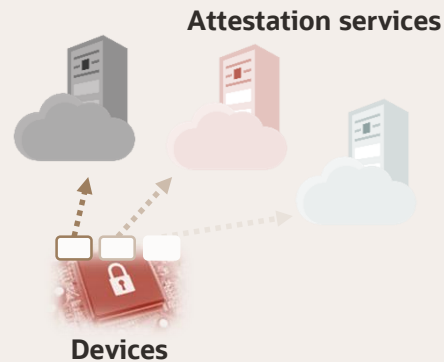
MULTI-CLOUD AUTHENTICATION

Provide device security across multiple IoT Solution Vendors and authentication schemes.



ADAPTABLE ATTESTATION

Support multiple proprietary or standard Device Attestation schemes.

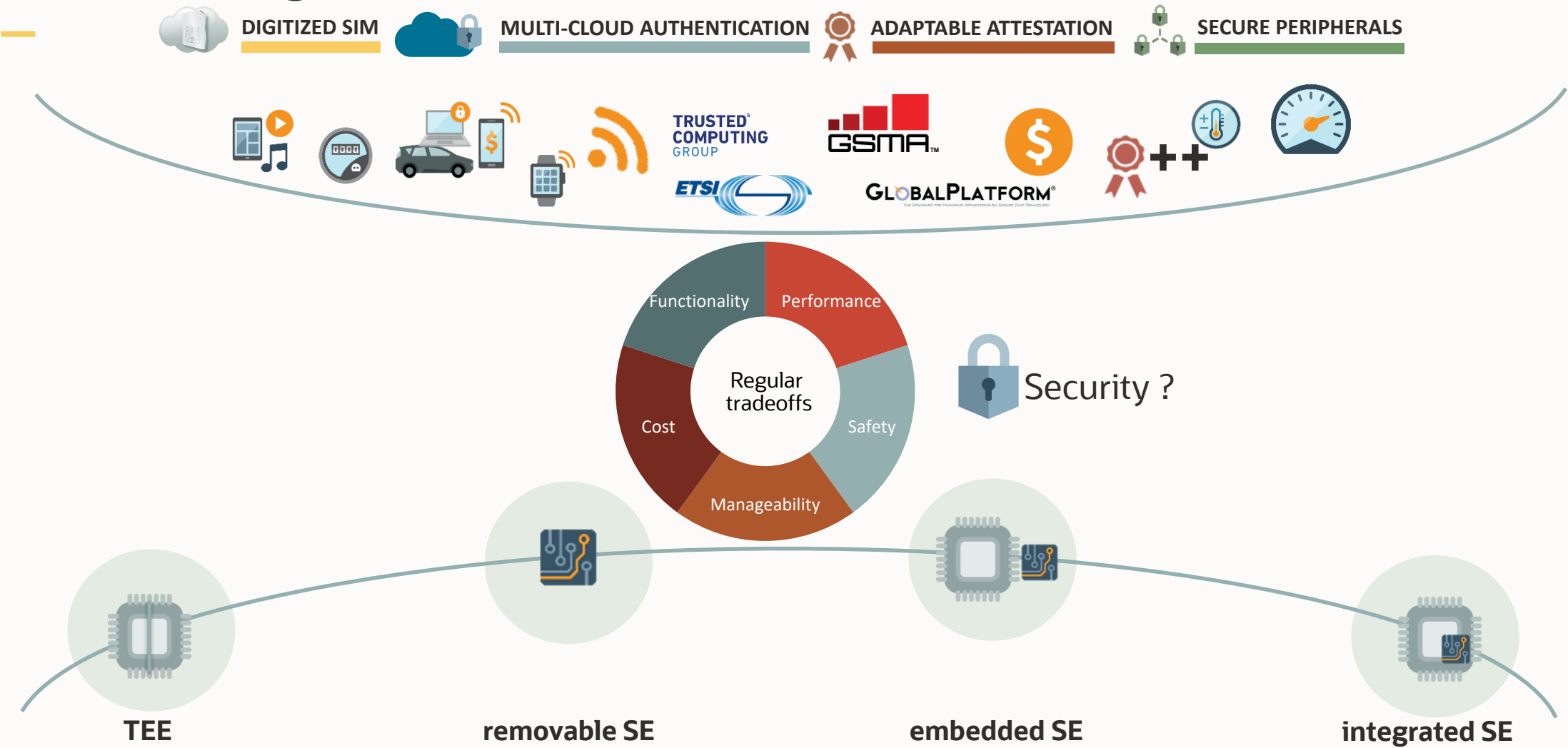


SECURE PERIPHERALS

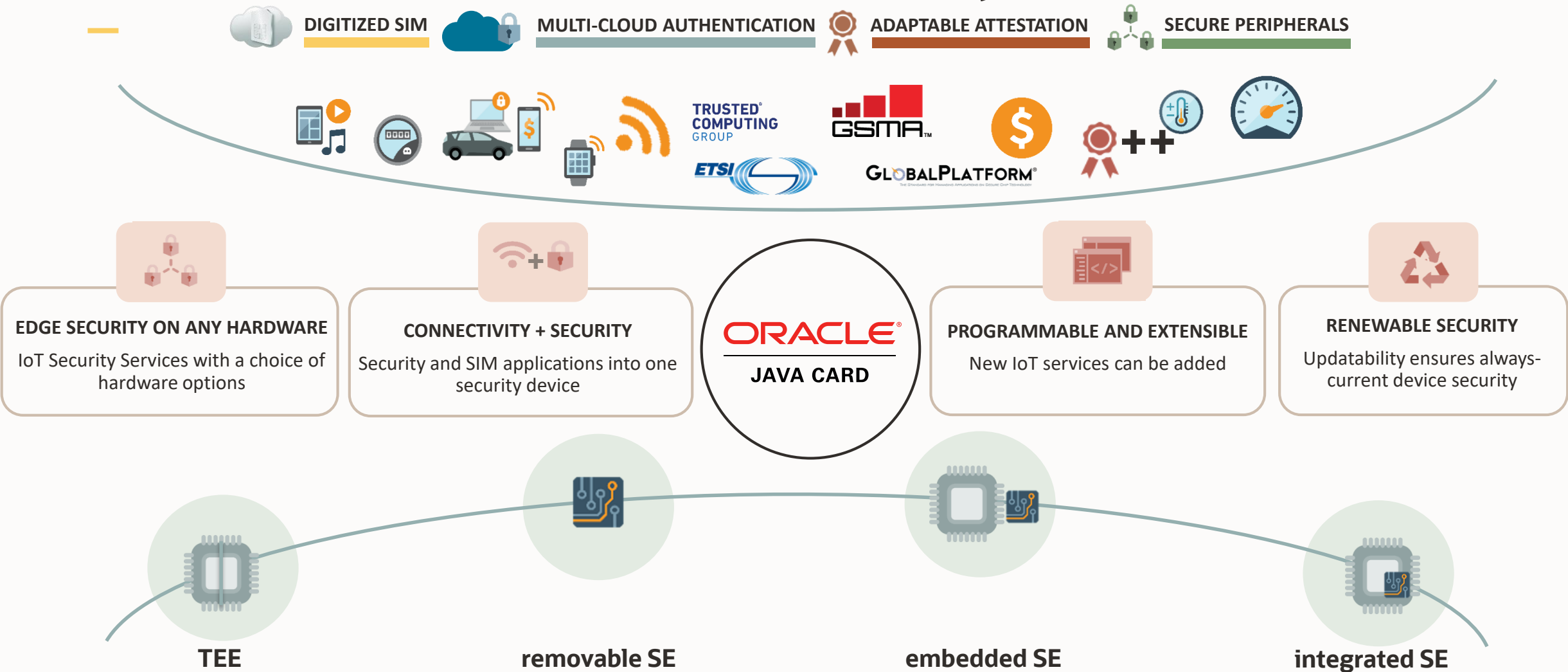
Securely access and control peripherals, enabling trust and exchange of sensitive data at the very edge.



OEM Challenges



Framework to take the Risk out of IoT Security





Java Card and Flexible IoT security

Features & Ecosystem

JC3.1 New API for IoT security services

Certificate API to optimize storage and certificate handling

- Parse, build and verify certificates
- Support for X.509, can be extended with other formats

Key derivation API for secure communication

- Pseudo Random Functions (including TLS)
- Key Derivation Functions (NIST SP 800-108, RFC 5869, ICAO, ANSI X9.63...)

Monotonic Counter API for anti-replay functions

- Support for persistent and transient counters
- Secure encapsulation of hardware functions

System Time API for timestamps, timer or watchdogs

- Get the system uptime and perform time duration operations



DIGITIZED SIM

Abstract the SIM from the underlying hardware



MULTI-CLOUD AUTHENTICATION

Device Security across multiple IoT Solution Vendors and authentication schemes



ADAPTABLE ATTESTATION

Multiple attestation schemes and standards on one chip



SECURE PERIPHERALS

Sensitive and Trusted Data at the very Edge

JC3.1 Cryptography API extended with new algorithms

Enhanced Elliptic Curves Cryptography

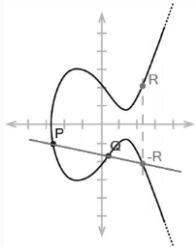
Named curves to simplify configuration and optimize use of ECC

X25519 and *X448* Key Agreements schemes

Ed25519 and *Ed448* digital signature (EdDSA)

FRP256v1 digital signature and key agreement

SM2 digital signature and key agreement



Configurable Key Pair generation

Ability to configure some parameters like primality test or random number generator

Allows an application to select its own parameters for key generation



Extended set of cryptographic algorithms

Additional AES encryption modes

- AES-CFB for stream ciphering
- AES-XTS for external storage encryption

SM3 hashing algorithm

SM4 block cipher algorithm



JC3.1 Extensible I/O framework for new hardware platforms

Platform abstractions for new I/O interfaces

Event framework to implement different and specialized communication models

New API to efficiently access and manipulate data in I/O buffers

Benefits

Support new integration models, using different physical I/O interfaces (UART, SPI, I2C, GPIO, ...)

Support different messages structures to integrate with external applications (sockets, TLS, CBOR, UBJSON, ...)

Support new use-cases, with secure element directly accessing and controlling peripherals and sensors



Extensible I/O framework to access peripherals

Examples

Smart-metering

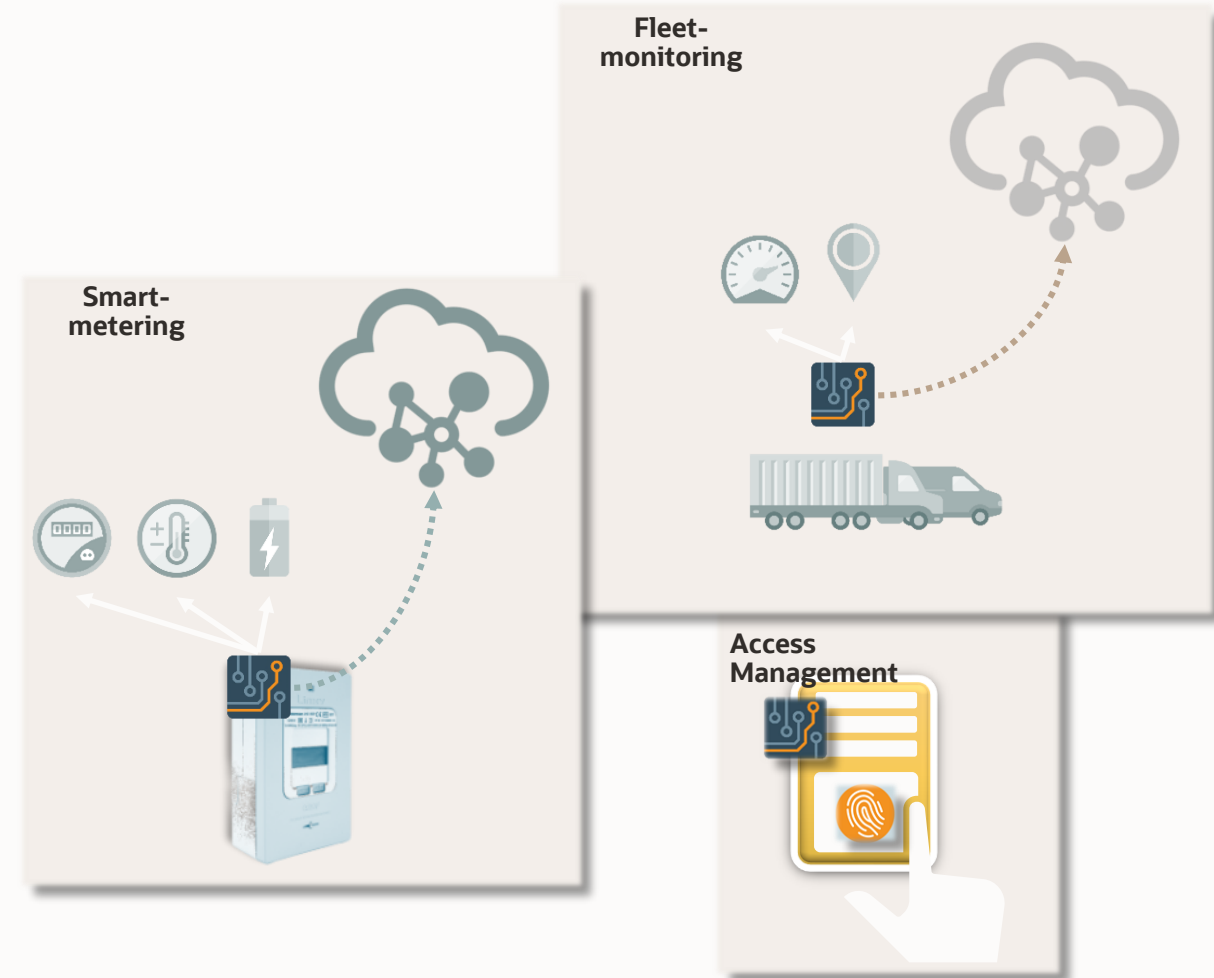
Java Card application uses some sensors to detect tampering and access meter data to enforce measurement integrity.

Fleet monitoring

Java Card application accesses peripherals to securely monitor vehicle parameters and report alerts.


Identity and Access Management

Java Card application uses biometric sensor to securely capture fingerprint and perform matching



Java Card Ecosystem & Interoperability



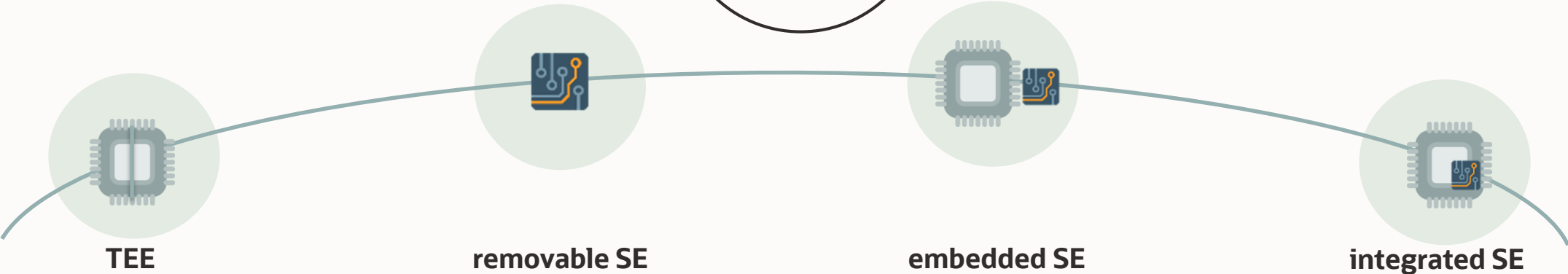

ACCROSS
Runtime Requirements and Test Suites


ACCROSS
Security Labs

ORACLE®
JAVA CARD


ACCROSS
Products Portfolio Compliance


ACCROSS
OEMS



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