

Вещи в Интернете: затаскиваем облачные сервисы на «безмозглые железяки»

Александр Белокрылов
Александр МIRONENKO



BY THE YEAR 2020, THERE WILL BE

50,000,000,000 connected devices,
creating and sharing

40,000,000,000,000 GB

worth of data across the Internet of Things.

Отслеживание домашних животных



Телемедицина



Умный дом



Умный дом



Интернетизация автомобилей



Мир меняется прямо сейчас...

Интеллектуальные устройства

+ вездесущий Internet

+ интеграция в IT инфраструктуру

+ Распределенная Аналитика

...меняют

Индустрию встроенных устройств

Тренд №1 в Embedded мире

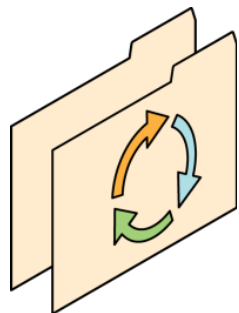
Программное обеспечение

— главная ценность встроенных систем

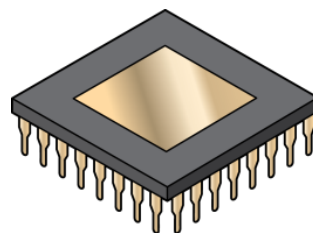
Какие требования к оконечным устройствам в IOT?

- Работа с периферией (I2C, UART, USB, CAN, SPI ...)
- Возможность иметь обновляемую бизнес-логику
- Безопасная работа с сетью
- Модель безопасности
- Быстрый процесс вывода на рынок конечного продукта

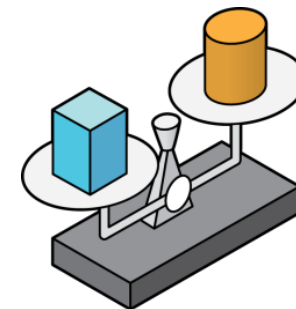
Вызовы в Embedded мире сегодня



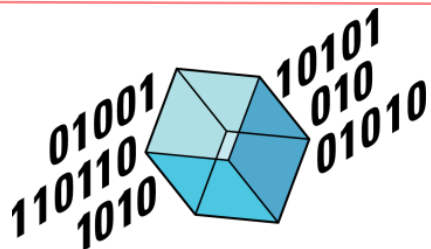
Большое разнообразие
вариантов
использования и железа



Возможности железа и
сетевых подключений
эволюционируют



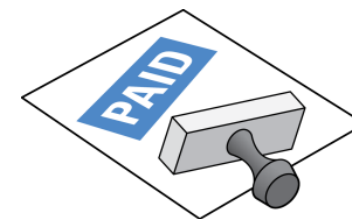
Soft главная ценность,
но встроенный софт
сложно разрабатывать



Приходится постоянно
изобретать самое
необходимое



Машинный интеллект,
как бизнес драйвер



Время выхода на рынок и
гибкость решений –
ключевые качества

Что делать?

Решения сегодня

mbed



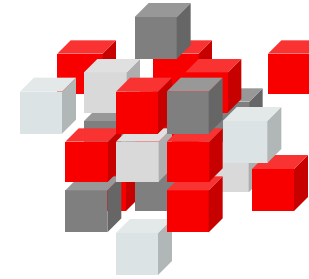
Introducing Java Embedded



Абстракция HW



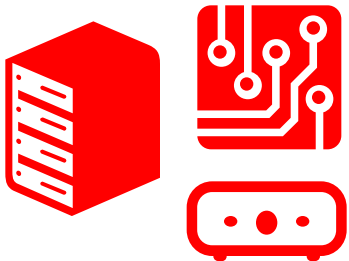
Write-once



Модулярность



Продвинутая модель безопасности



Распространенная
Модель разработки



Удаленная разработка и
управление



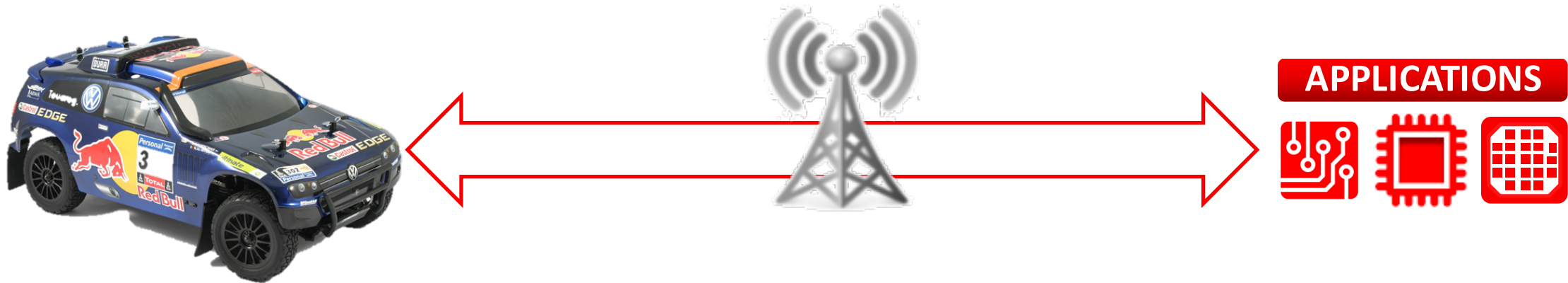
Невысокие затраты на
разработку и
тестирование



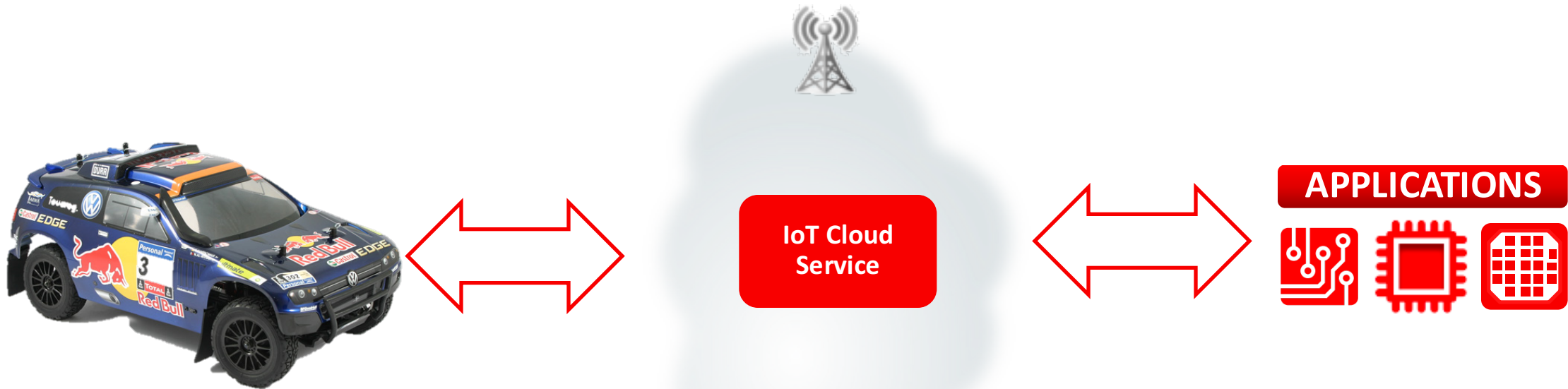
Растущая экосистема

Пример использования

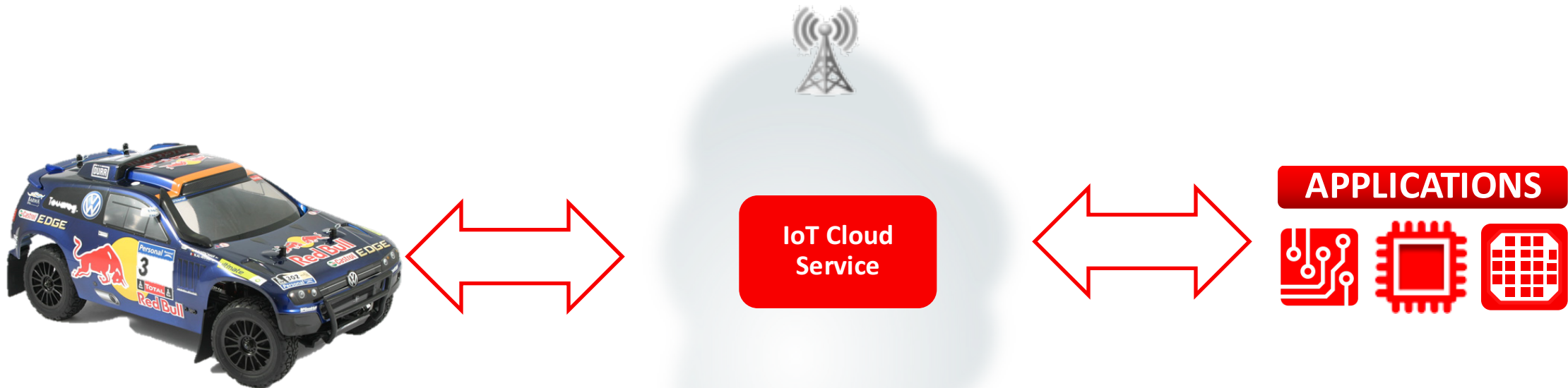
Use Case – продвинутое страхование автотранспорта



Use Case – продвинутое страхование автотранспорта



Use Case – продвинутое страхование автотранспорта



Alert types

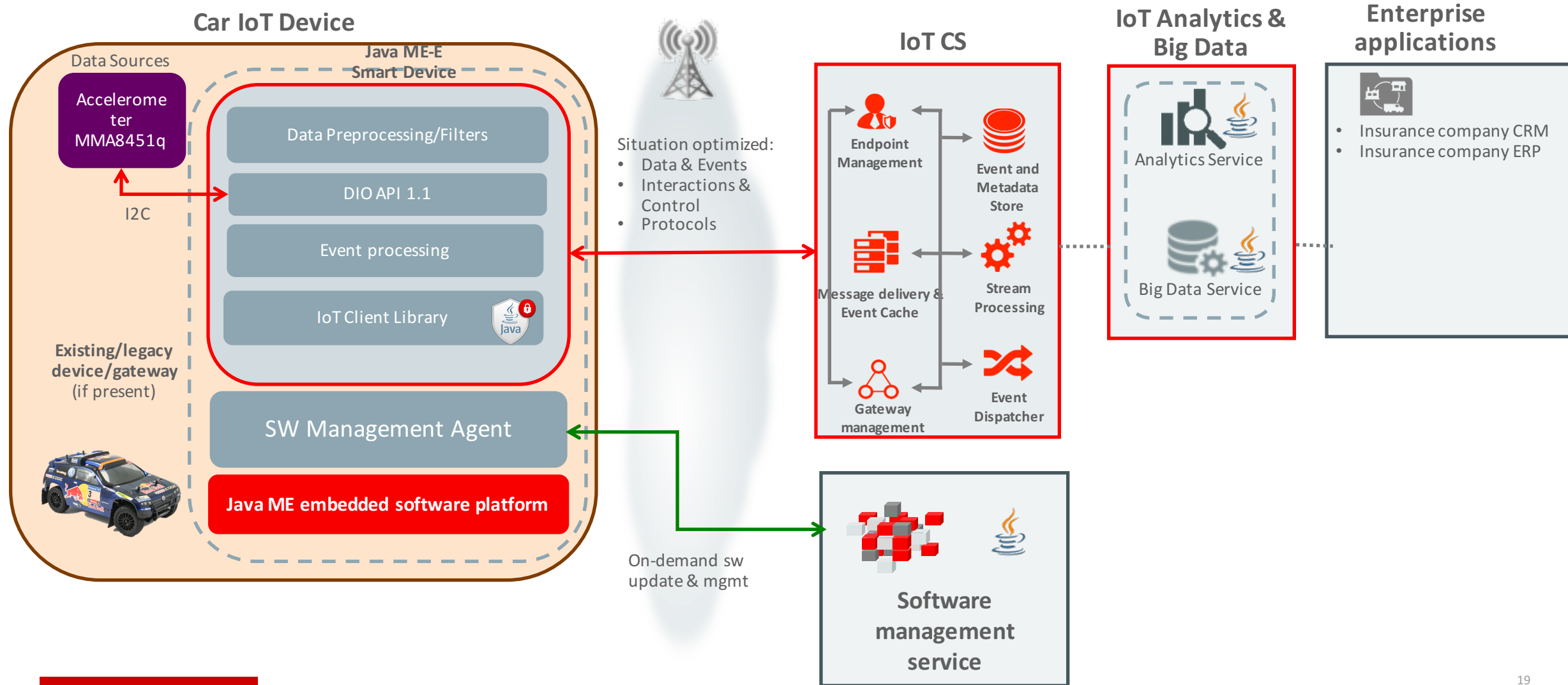
Bad Driving
Dangerous Driving
Collision
Roll Over

Java ME Embedded Demo

Лучше один раз увидеть, чем сто раз услышать

Архитектура

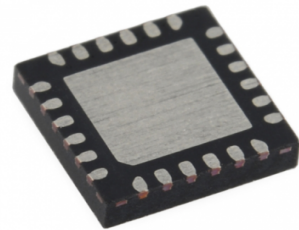
Пример: продвинутое страхование



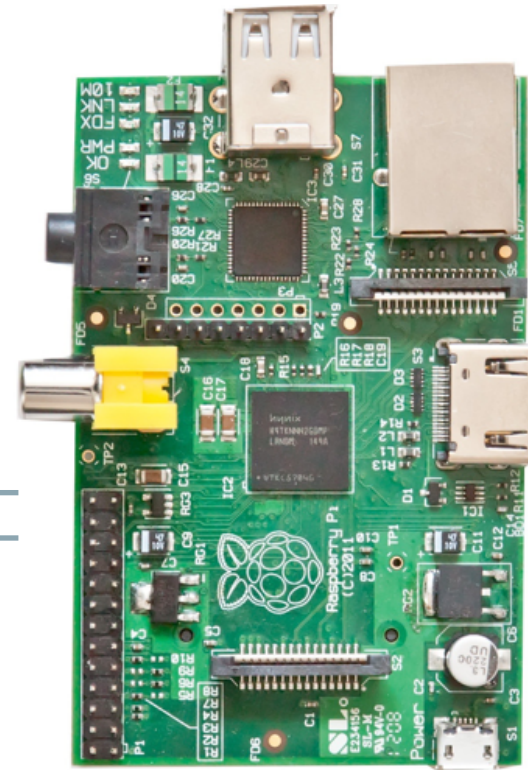
Challenge 1

Работа с периферией для получения данных

Java и низкоуровневая периферия???!!!



I2C Bus



Solution 1

DIO I2C API

```
// Open I2C Slave
I2CDevice sensor = DeviceManager.open(I2CDeviceConfig_instance);

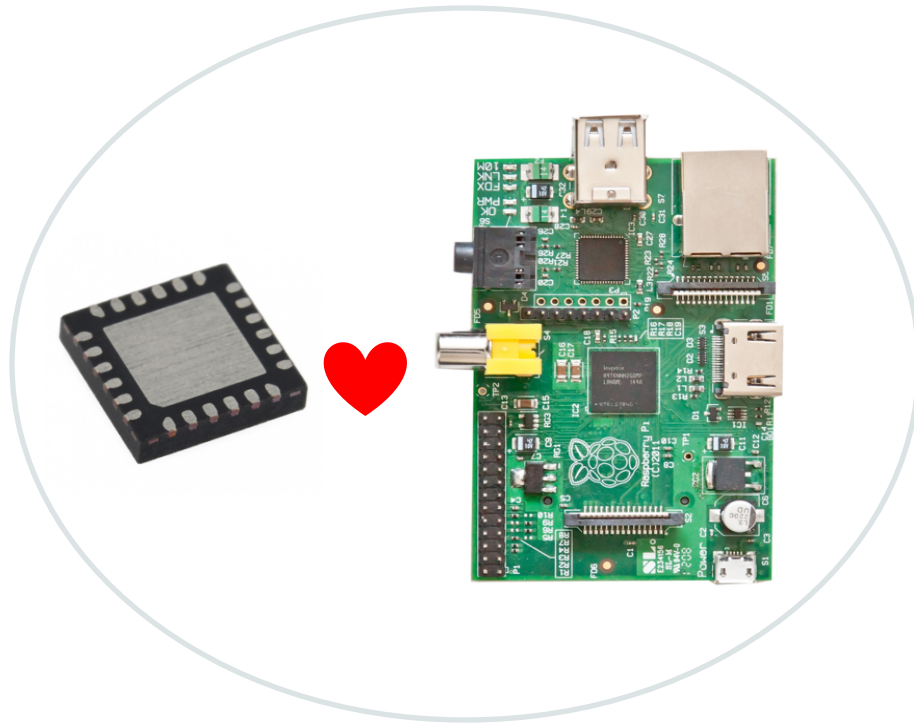
// Write data to sensor
sensor.write(ByteBuffer_instance);

// Read data(for instance, temperature)
sensor.read(ByteBuffer_instance);

sensor.close();
```

Challenge 2

Secured connection with a cloud



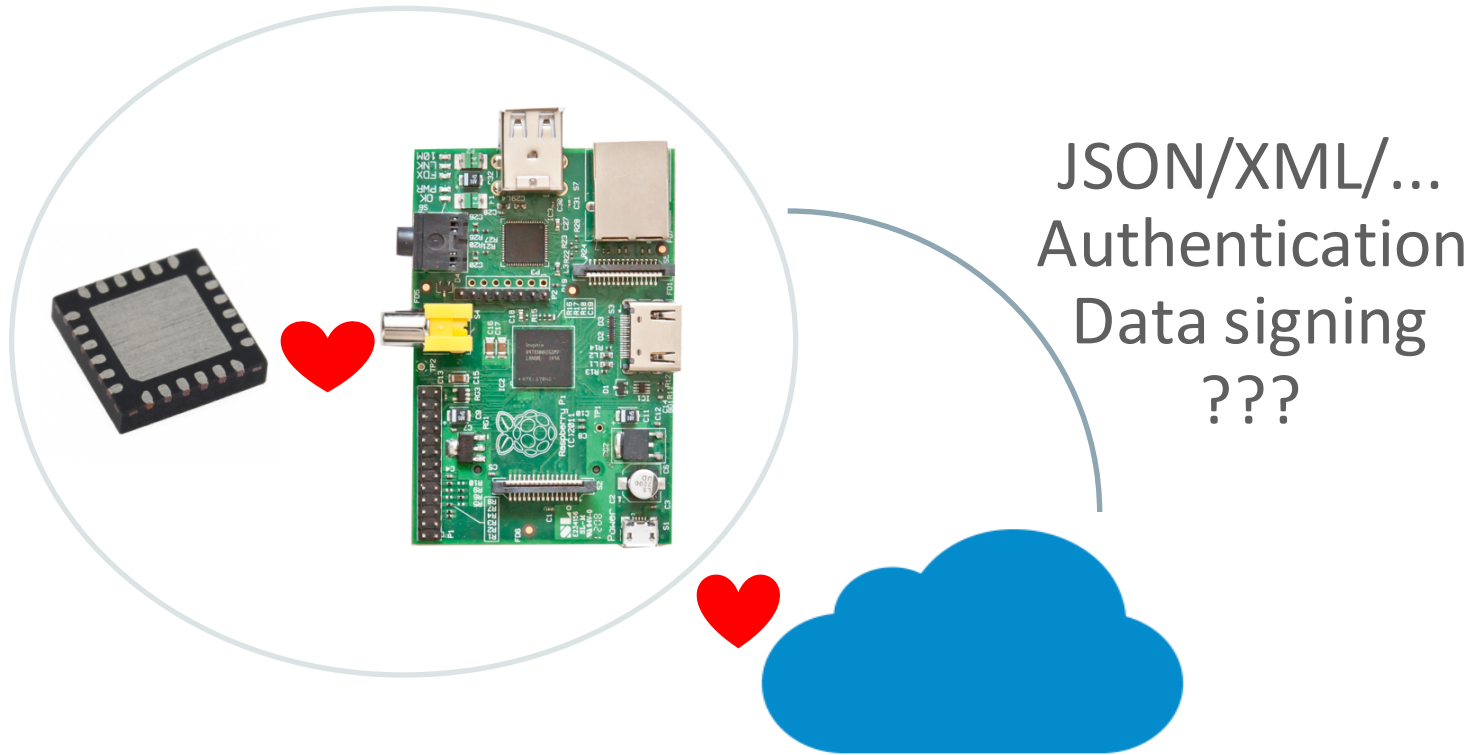
Solution 2

Java ME CLDC Generic Connection Framework

```
HttpsConnection conn =  
    (HttpsConnection)Connector.open(HTTPS_URL_HOST);  
  
DataOutputStream dataOutputStream =  
    conn.openDataOutputStream();  
dataOutputStream.writeUTF("...");
```

Challenge 3

Data preparing and processing



Solution 3

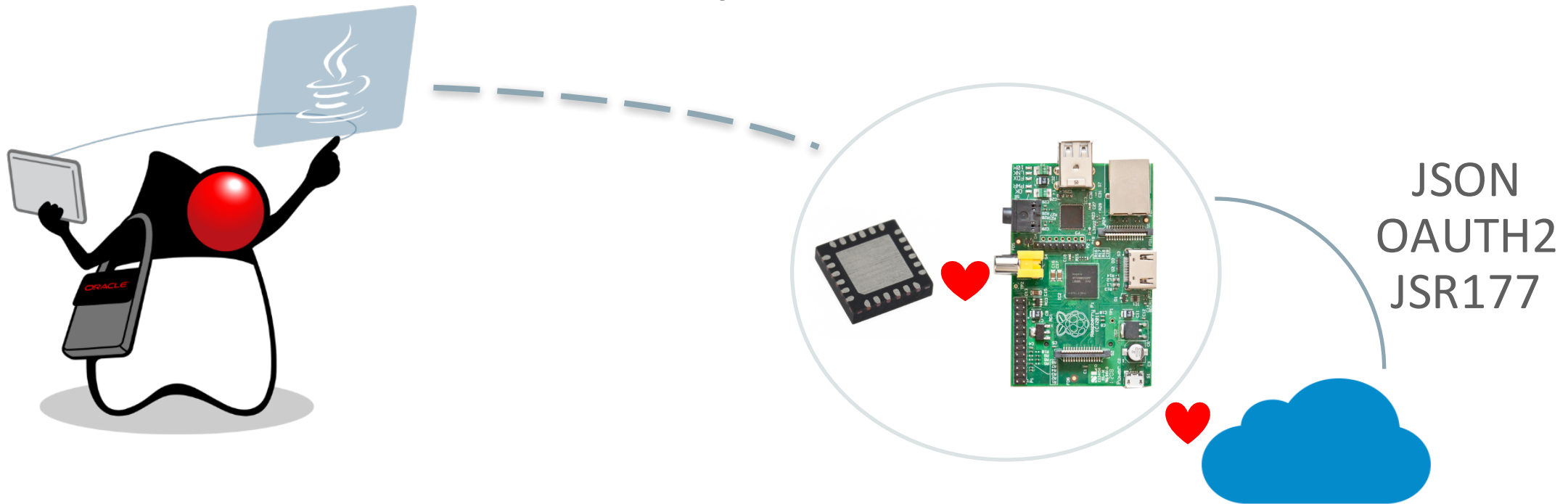
Standard components of Java ME Embedded stack

- OAUTH2 for device authentication
- JSON API for JSON message making and parsing
- JSR177: Security and Trust Services API for security and key management

Challenge... Not for cloud, FOR US!

Software management for application provisioning

Soft update over the air



Solution... Easy solution!

Java ME Software management API

```
SuiteManager sManager = ManagerFactory.getSuiteManager();  
SuiteInstaller sInstaller = sManager.getSuiteInstaller(  
    URL_OF_MY_APPLICATION,  
    false);  
sInstaller.addInstallationListener(myInstallationListener);  
  
sInstaller.start();
```

Кроссплатформенность

Кроссплатформенность

- Write once run anywhere с небольшими оговорками
- Перенос между устройствами не требует изменений кода
- Демо платформы
 - Raspberry Pi model B+
 - Intel Edison
 - STM32429I-EVAL

Java ME Embedded – эффективный инструмент IoT



The preceding 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, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

ORACLE®

BackUp Slides

Java ME Embedded система безопасности

Java ME 8 Security Overview

Проверенная годами система безопасности

- Безопасность платформы
- Контроль доступа на уровне платформы
- Криптография
- Защищенные каналы коммуникации

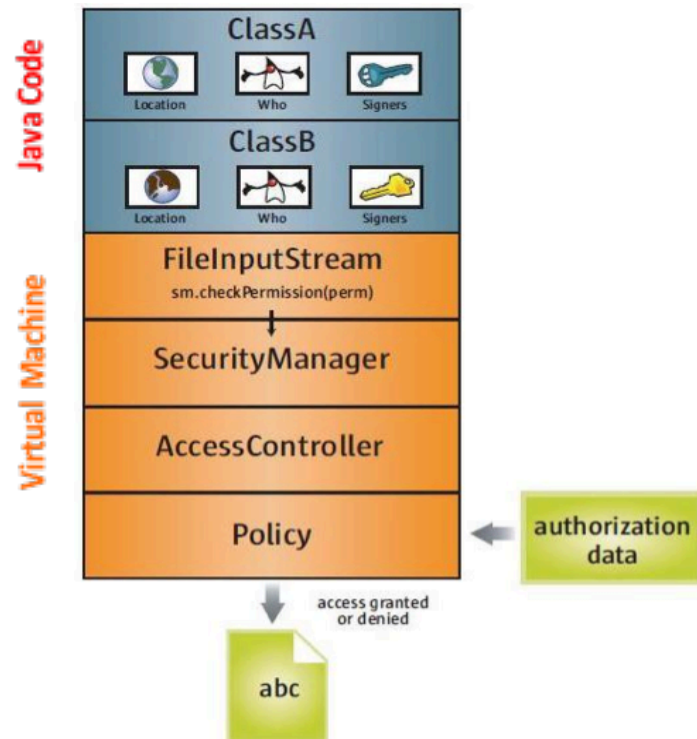
Безопасность платформы

Язык, компилятор, JVM

- Статическое типизирование
- Управление памятью
- Верификация байт-кода
- Любое приложение может быть
 - Подписано
 - Аутентифицировано
 - Авторизовано

Безопасность платформы

Java Runtime контролирует доступ приложения к ресурсам



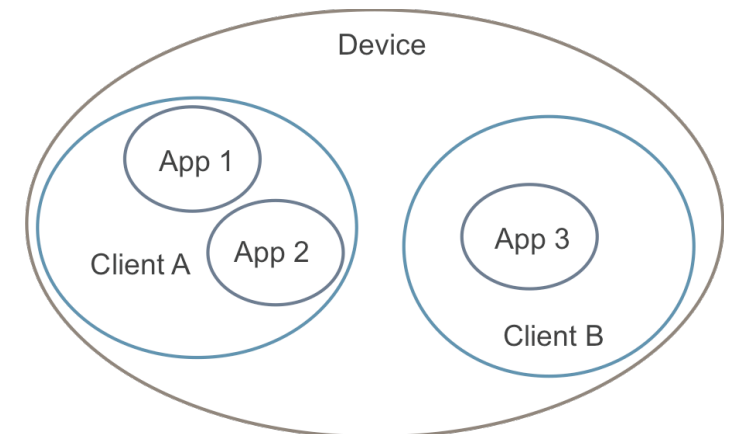
Например: Доступ к файловой системе

- Виртуальная машина контролирует доступ приложений к ресурсам
- Доступ должен быть явно разрешен
- Полномочия проверяются при каждой попытке использования ресурса
- Любой нежелательный или несанкционированный доступ блокируется на уровне Java Runtime

Контроль доступа на уровне платформы

Многопользовательские домены

- Java ME 8 поддерживает множественные Client Security Domains на устройствах
- Примеры клиентов: ODM, сервис провайдеры, системные интеграторы, разработчики
- Каждому клиенту – собственный домен (правила, привилегии, параметры)
- Приложения запускаются в клиентском домене с ограниченными привилегиями и ресурсами
- Ресурсы могут быть общими для нескольких доменов



Криптография

Расширяемый, полнофункциональный API

- Поддерживается широкий спектр алгоритмов шифрования, включая RSA, DSA, AES, Triple DES, SHA, PKCS#5, RC2, and RC4
 - Аппаратное ускорение (зависит от платформы)
- PKCS#11

Защищенные каналы коммуникации

Стандартные и безопасные

- Secure Sockets Layer (SSL)
- Transport Layer Security (TLS/DTLS)
- Поддержка HTTPS через SSL/TLS