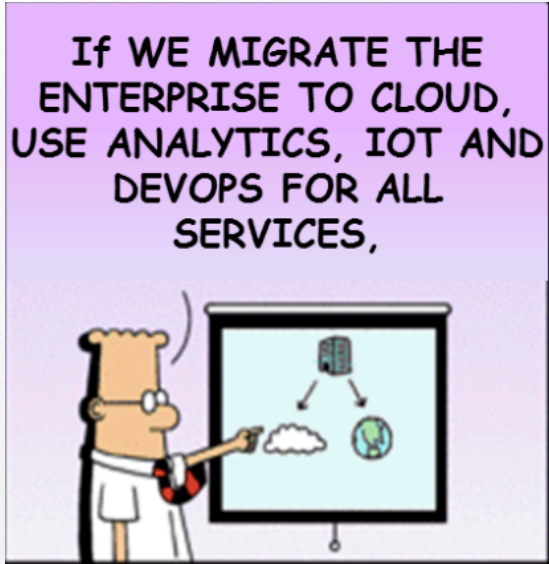
A person with blonde hair is shown in a meditative pose (lotus position) underwater. They are wearing a dark wetsuit and a snorkel mask. The background is a deep blue with many small bubbles rising from the person's mask, creating a serene and focused atmosphere.

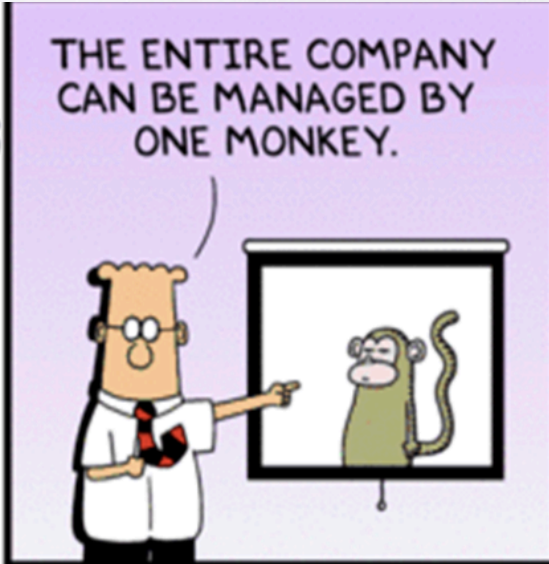
# Погружение в Интернет Вещей с Java 9

[WWW.BELL-SW.COM](http://WWW.BELL-SW.COM)

2018



Dilbert.com DilbertCartoonist@gmail.com



1-18-10 ©2010 Scott Adams, Inc./Dist. by UFS, Inc



....

# Кто здесь?

## Александр Белокрылов

 @gigabel

 BELLSOFT

Liberica – Java 10 для Raspberry Pi

<http://bell-sw.com>

Бывшие работодатели:

 ORACLE®

 Sun  
microsystems

 BELLSOFT



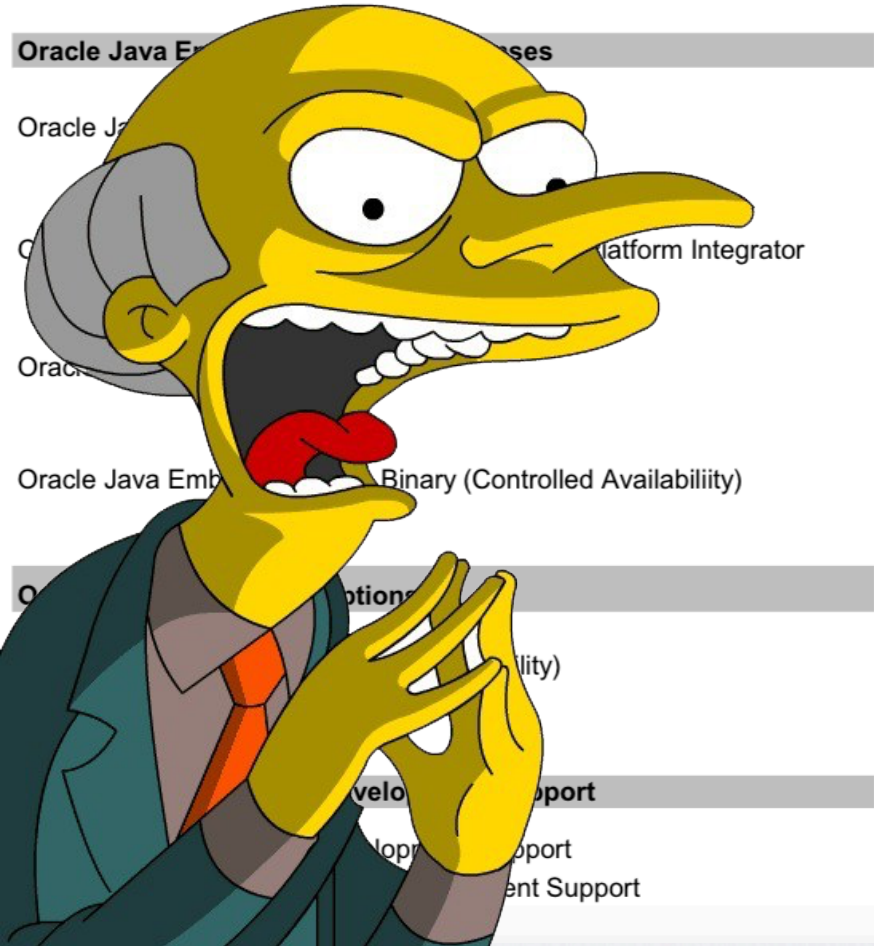
...  
2012 год



Java SE Embedded 7



No, it is not for free



License Price	Software Update License & Support	Metric	Notes
300.00	57.00	Java Embedded Processor	1, 3
300.00	57.00	Java Embedded Processor	1, 5
300.00	57.00	Java Embedded Processor	1, 4
450.00	85.50	Java Embedded Processor	1, 7
1,000.00	190.00	Java Embedded Processor	1, 7
<b>Annual Fee</b>			
50,000.00		Standard Binary	2
75,000.00		Standard Binary	2, 7

....

## Java SE Embedded 8 – последний релиз



# OpenJDK™ 9

- Все фичи Java Embedded влиты в OpenJDK 9
- ARM32 & ARM64 порты в OpenJDK 9
- Использование бесплатно



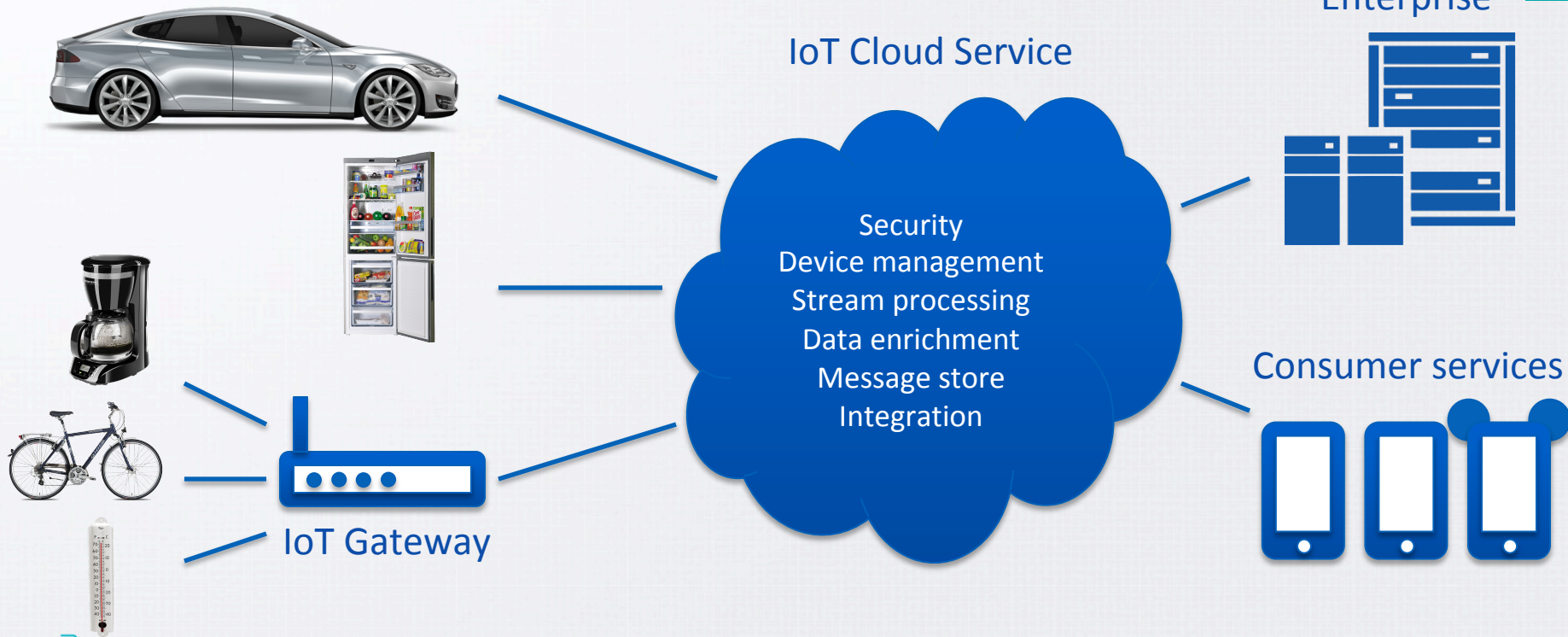


# Интернет Вещей





# IoT Architecture





# IoT устройства: Маленькие

- ARM Cortex M0 – M7
- MIPS32
  - RAM 4 – 512KB
  - ROM 4 – 2MB



Electric Imp [www.electricimp.com](http://www.electricimp.com)



# Electric Imp



electric imp Code Documentation Forums

device name or id

Unassigned Devices 1

235e1b3643fc42ee

Active Models

- Serial Pipeline
- Shield Red
- Serial Pipeline 2
- Breakout

Inactive Models

- blinker
- Example
- Hannah-TempLog
- Hello, blink
- Imp Web Interface
- I/O Control
- LED-Trigger-Wiper
- LED Web Control
- LED-Web-Control
- Web Response
- Example

### Serial Pipeline 2 Build 6

Breakout | device online | agent running | agent link

Check Build and Run

```
Agent - https://agent.electricimp.com/lyy9rPLn807m running
1 // Imp Serial Pipeline Agent
2 by: Jim Lindblom
3 SparkFun Electronics
4 date: March 24, 2014
5 License: Beerware, Use, reuse, and modify this code however you see fit.
6 If you find it useful, buy me a beer some day!
7
8 The Serial Pipeline model is designed to pass serial data from one Imp to
9 another. A second Serial Pipeline model should be treated to be a near exact
10 copy of this, except the agentURL variable should be modified to the URL of
11 this agent.
12
13 The agent must accomplish two tasks:
14 1. On data in from the Imp, post it to another agent. This is the
15 agentURL variable at the top of the code.
16 2. On data from another agent, send that through to the Imp.
17
18 Resources:
19 http://electricimp.com/docs/api/hardware/uart/configure/
20 http://mactecbean.com/2014/03/imp-to-imp-communication/
21 //
22
23 // The URL of the other Imp's agent:
24 const agentURL = "https://agent.electricimp.com/EPTAHdSMeth";
25
26 // When the Imp sends data to the agent, that data needs to be relayed to the
27 // other agent. We need to construct a simple URL with a parameter to send
28 // the data.
29 device.on("ImpSerialIn", function(char)
30 {
31 // Construct the URL. The other agent will be looking for the key "data".
32 // Value will be the byte-voise of our serial data.
33 local uriSend = format("%s?data=%s", agentURL, char);
34 // If we get the constructed URL.
35 http.get(uriSend).sendasyn(function(resp){});
36 });
37
Device log
2014-03-25 14:02:22 UTC+6: [Status] Device booting; 4.13% program storage used
2014-03-25 14:02:22 UTC+6: [Device] Serial Pipeline open!
```

```
Device - 23086d4cced3dbee on106
1 // Imp Serial Pipeline Device
2 by: Jim Lindblom
3 SparkFun Electronics
4 date: March 24, 2014
5 License: Beerware, Use, reuse, and modify this code however you see fit.
6 If you find it useful, buy me a beer some day!
7
8 The Serial Pipeline model is designed to pass serial data from one Imp to
9 another. Data transfers Look like this:
10
11 Arduino 1 Serial Out to Imp 1 -> Imp 1 passes serial data to Agent 1 ->
12 Agent 1 passes data to Agent 2 -> Agent 2 passes data to Imp 2 -> Imp 2
13 passes data to Arduino 2 over serial. Whew.
14
15 A second Serial Pipeline model should be created to be a near exact
16 copy of this, except the agentURL variable (on the agent code) should be
17 modified to the URL of this device's agent.
18
19 The device must accomplish two tasks:
20 1. On serial data in, send it off to the agent. The agent will send that
21 data off to the other agent.
22 2. On data coming in from the agent, send it through the serial port.
23
24 Resources:
25 http://electricimp.com/docs/api/hardware/uart/configure/
26 http://mactecbean.com/2014/03/imp-to-imp-communication/
27 //
28
29 // Global Variables
30 //
31 ////////////////////////////////////////////////////////////////////
32 local txLEDToggle = 1; // These variables keep track of rx/tx LED state
33 local txLEDToggle = 1;
34 arduino <- hardware.pin57;
35 rxLED <- hardware.pin8;
36 txLED <- hardware.pin9;
37
```



# IoT устройства: Большие

- ARM A5 – A9
- Intel Atom
  - RAM 256 MB+
  - ROM 256 MB+





# IoT Gateway

- Большие устройства



SuperMicro  
Intel Core i7  
RAM 64GB



Dell  
Intel Atom  
RAM 4GB



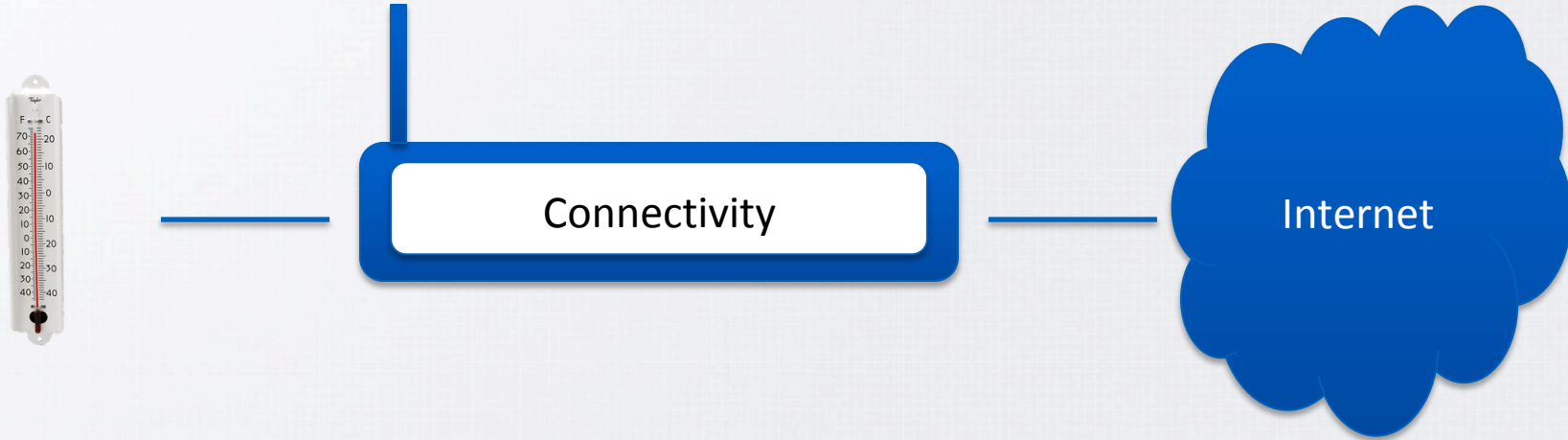
Eurotech  
ARM Cortex A8  
RAM 512 MB



Advantech  
ARM Cortex A9  
RAM 1GB

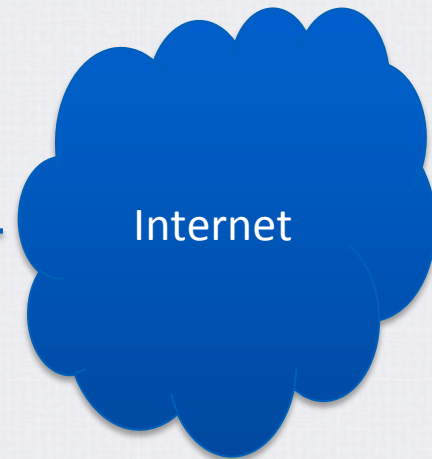
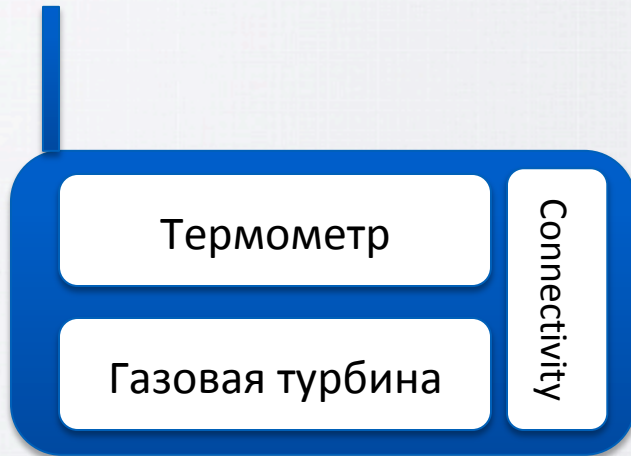
....

# IoT gateway: Connectivity for Edge Devices



....

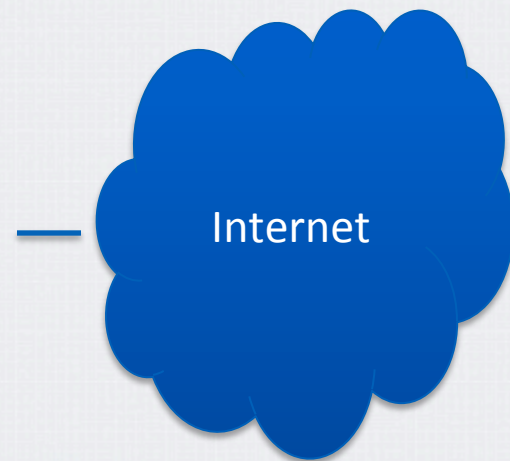
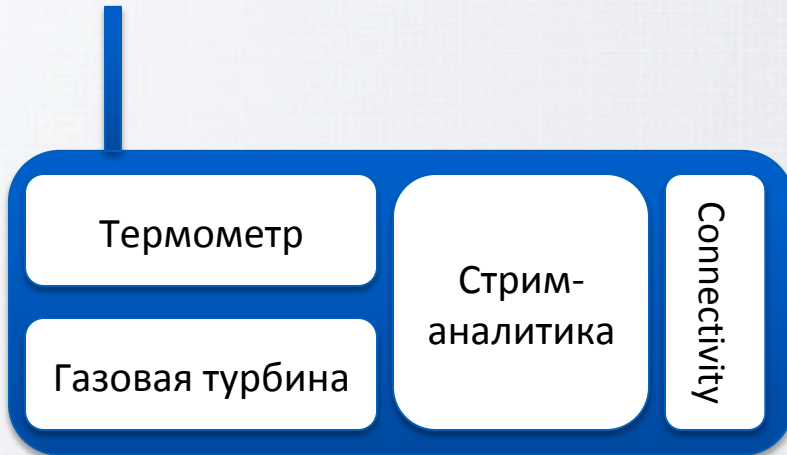
# IoT gateway: Modular Software





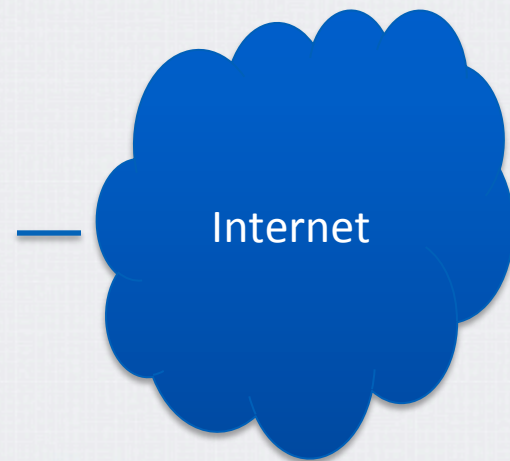
...

# IoT gateway: Stream processing



...

# IoT gateway: Дополнительные функции



...

# ИОТА – криптовалюта IoT

- Расширяемая
- Децентрализованная
- Цена транзакции – бесплатно
- Клиент на Java



# Java в IoT

- Eclipse Kura
  - OSS
  - Используется несколькими производителями Eurotech, Sierra Wireless
- Bosch IoT Gateway Software
- Oracle IoT Gateway
- GE Predix Edge





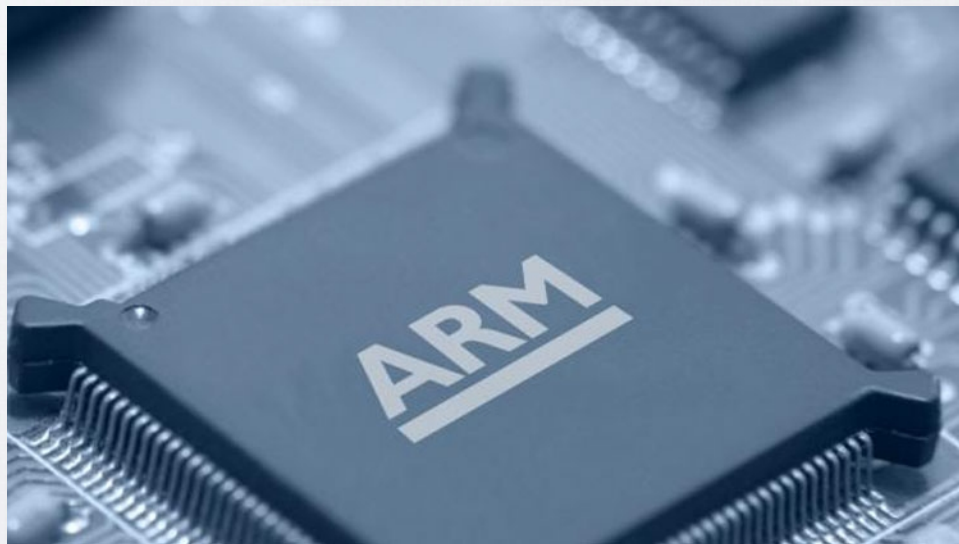
## Java 9 дистрибутивы для IoT Gateway

- Oracle Java SE Embedded 8
  - No Java SE Embedded 9
  - No Binary for ARM in JDK 9
- Azul Systems, Zulu Embedded 9
- BellSoft, Liberica 9



# Open JDK ARM ports

- ARM
  - 32 bit / “64 bit”
    - ARM v6
    - ARM v7
    - ARM v8
- AARCH64
  - 64 bit only





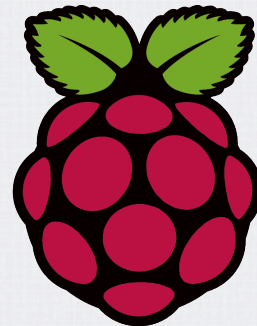
# Raspberry Pi 3

SoC Broadcom 2837  
4x ARM Cortex-A53 64-bit



# Linux on Raspberry Pi 3

- Raspbian **32 Bit**
  - All Raspberry Pi versions supported
- Альтернативы:
  - openSUSE Linux 64 Bit





# Java 9/10 фичи для IoT

- Модуляризация
- AppCDS
- http/2
- Process API



....

# Модуляризация



....

## Построим граф зависимостей

```
jdeps -R -s ThermoApp/  
ThermoApp -> java.base  
ThermoApp -> java.logging  
ThermoApp -> jdk.dio
```

## Сделаем под себя

```
jlink \  
-p /usr/lib/jvm/jdk-9.0.4-bellsoft-arm32-vfp-hflt/jmods/ \  
--add-modules java.base,java.logging,jdk.dio \  
--output customjdk
```



## Избавимся от лишнего

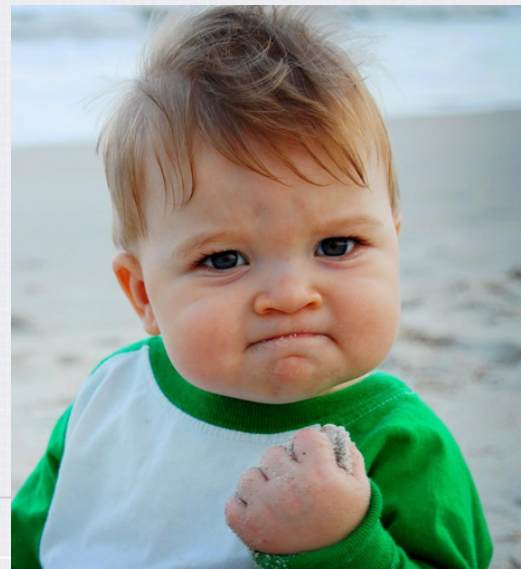
```
cp customjdk minjdk
```

```
rm -rf minjdk/lib/server/ minjdk/lib/client/
```

....

## Впечатляемся результатом

```
du -sh /usr/lib/jvm/jdk-9.0.4-bellsoft-arm32-vfp-hflt customjdk/ minjdk/  
379M /usr/lib/jvm/jdk-9.0.4-bellsoft-arm32-vfp-hflt  
49M customjdk  
28M minjdk Java9 Platform Module System
```



...

# Class Data Sharing



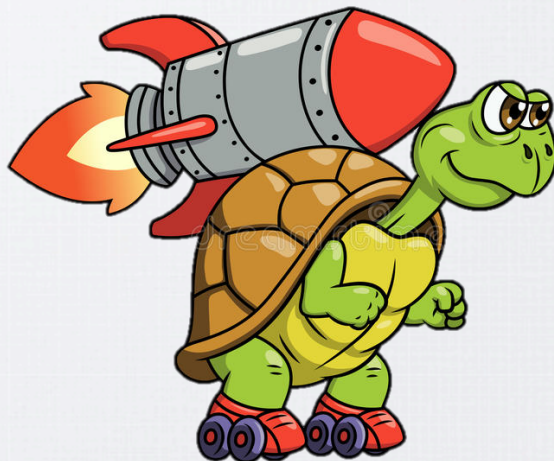
# Class Data Sharing





# AppCDS

- Цели
  - Сократить время старта
  - Уменьшить использование RAM за счет разделения мета-данных классов
- Oracle WebLogic стартует на 30% быстрее с AppCDS



# AppCDS on Raspberry Pi

**Apache Flink 1.4**

На Raspberry Pi 3 стартует на 15% быстрее

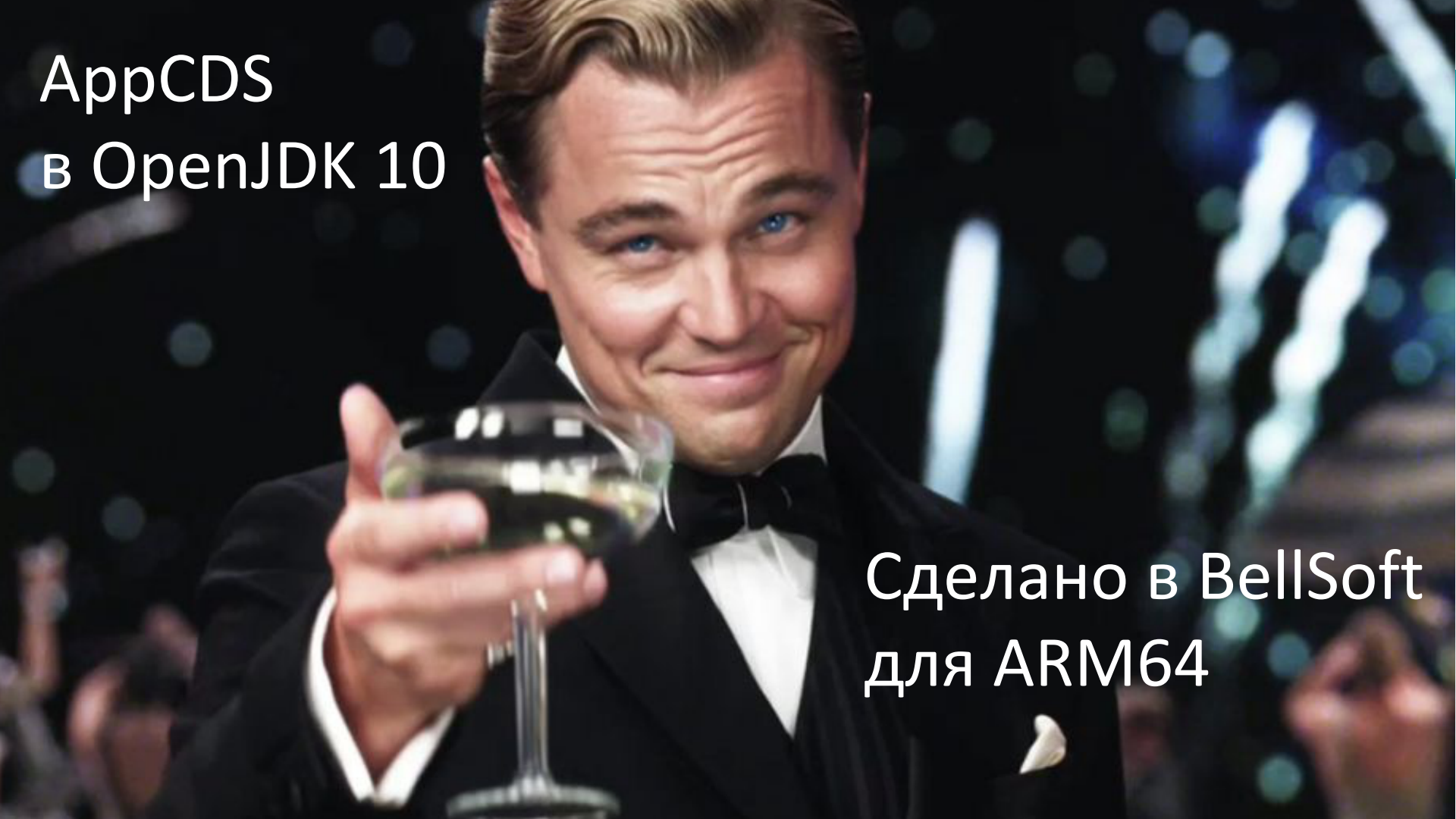




## AppCDS: доступность

- Oracle Java 8.40+
- Oracle Java 9.x

**-XX:+UnlockCommercialFeatures**

A close-up photograph of Leonardo DiCaprio from the movie 'Inception'. He is wearing a black tuxedo with a white shirt and a black bow tie. He has a slight, knowing smile and is looking directly at the camera. He is holding a martini glass filled with a green liquid, likely gin and tonic, in his right hand. The background is dark and out of focus, with some blue and white light streaks, suggesting a party or a club setting.

AppCDS  
в OpenJDK 10

Сделано в BellSoft  
для ARM64

....

## Запустить приложение и создать список классов

```
-XX:+UseAppCDS -Xshare:off \  
-XX:DumpLoadedClassList=classlist.clj
```



## Создать кеш

```
-XX:+UseAppCDS -Xshare:dump \  
-XX:SharedClassListFile=sample.classlist \  
-XX:SharedArchiveFile=sample.jsa
```



## Получаем удовольствие

```
-XX:+UseAppCDS -Xshare:on \  
-XX:SharedArchiveFile=sample.jsa
```

....  
http/2

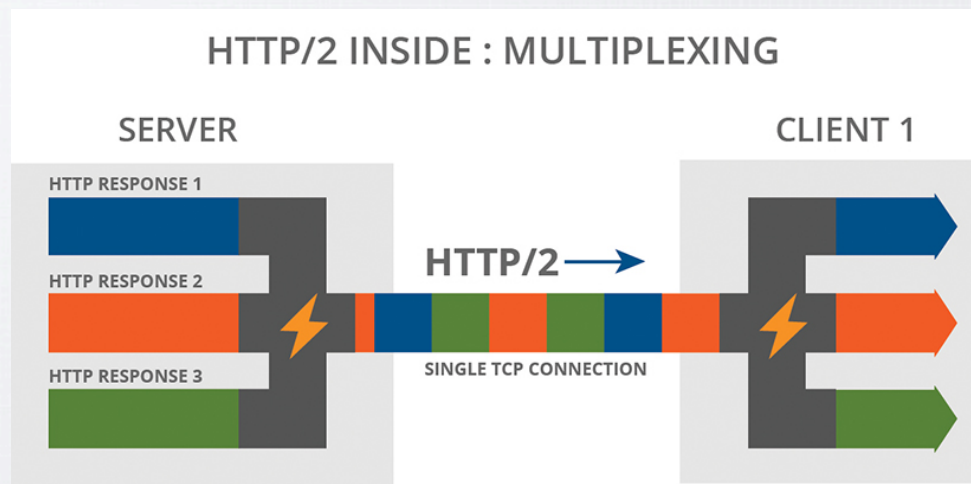
- Multiplexing
- Сжатие заголовков HPAC
- Server Push
- Бинарный протокол





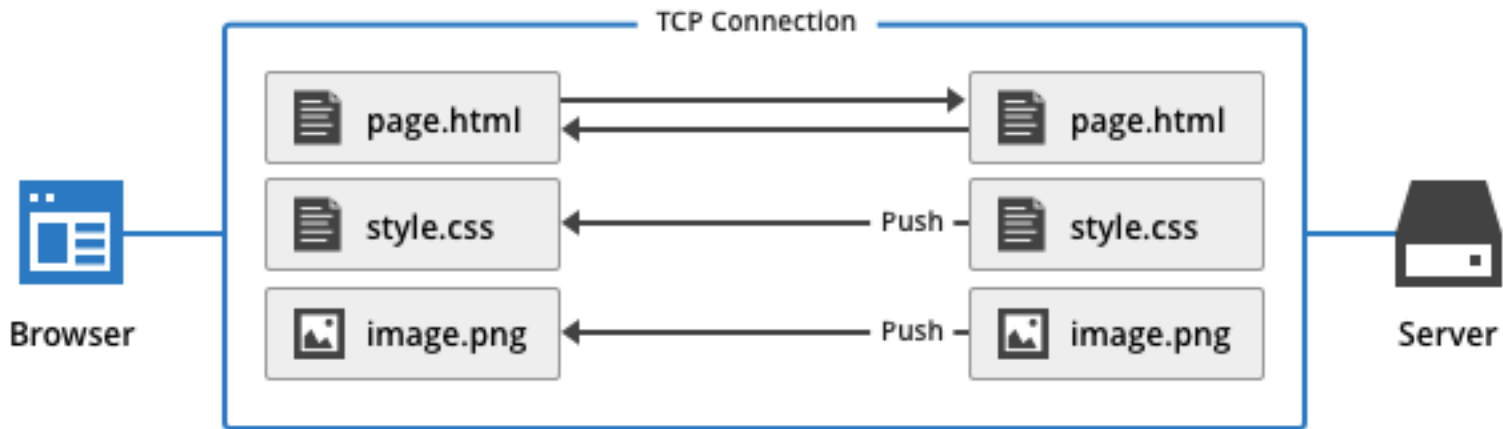
# Multiplexing

- Коннекция может иметь несколько стримов
- Стрим разделен на фреймы
- Порядок Request/Response может быть произвольным



# Server push

- Сервер может посылать данные до запроса
- Клиент хранит дополнительные данные в кэш
- Клиент может отказаться от получения данных



...

## Сжатие заголовков НРАС



....

## Бинарный протокол

There are only 10 types  
of people in the world:  
Those who understand binary  
and those who don't.

....

Publisher/Subscriber или Request/Response?



M

# HTTP/2

For Faster and Safer Internet



# http/2 в Java

- JEP 110
  - Java 9/10
  - `jdk.incubator.httpclient`
- JEP 321
  - Java 11
  - `java.net.httpclient`

# HTTP/2, Java 9

Синхронный запрос

```
HttpClient client = HttpClient  
    .newBuilder()  
    .build();
```

```
HttpRequest request = HttpRequest  
    .newBuilder(uriIp)  
    .GET()  
    .build();
```

```
HttpResponse<String> response = client  
    .send(request, HttpResponse.BodyHandler.asString());
```

# HTTP/2, Java 9

Асинхронный запрос

```
HttpClient client = HttpClient  
    .newBuilder()  
    .build();
```

```
HttpRequest asyncRequest = HttpRequest  
    .newBuilder(uriIp)  
    .GET()  
    .build();
```

```
CompletableFuture<HttpResponse<String>> asyncResponse = client  
    .sendAsync(asyncRequest, HttpResponse.BodyHandler.asString());
```



...

# Process API





# OS Process access

## JEP 102: Process API Updates

Расширение возможностей управления процессами операционной системы



# Java 9: Process API

- **JEP 102: Process API Updates**
  - Расширен класс `java.lang.Process`
  - Новый интерфейс `ProcessHandle`

# Java 9: Process API

```
static Stream<ProcessHandle> allProcesses()  
Stream<ProcessHandle> children()  
int compareTo(ProcessHandle other)  
static ProcessHandle current()  
Stream<ProcessHandle> descendants()  
boolean destroy()  
boolean destroyForcibly()  
Long pid()  
ProcessHandle.Info info()  
boolean isAlive()  
static Optional<ProcessHandle> of(long pid)  
CompletableFuture<ProcessHandle> onExit()  
Optional<ProcessHandle> parent()  
boolean supportsNormalTermination()
```



## Process API: allProcesses()

```
ProcessHandle.allProcesses().filter(ProcessHandle::isAlive)
    .filter(process -> process.info().command().isPresent())
    .forEach(process -> printProcessInfo(process));
}
```

## PROCESS INFORMATION

=====

Process id: 27557

Command: /Library/Java/JavaVirtualMachines/jdk-9.0.4.jdk/Contents/Home/bin/java

Arguments: ProcessAPISample

Command line: /Library/Java/JavaVirtualMachines/jdk-9.0.4.jdk/Contents/Home/bin/jo

Start time: 2018-03-04T21:17:21.963Z

Run time duration: 410ms

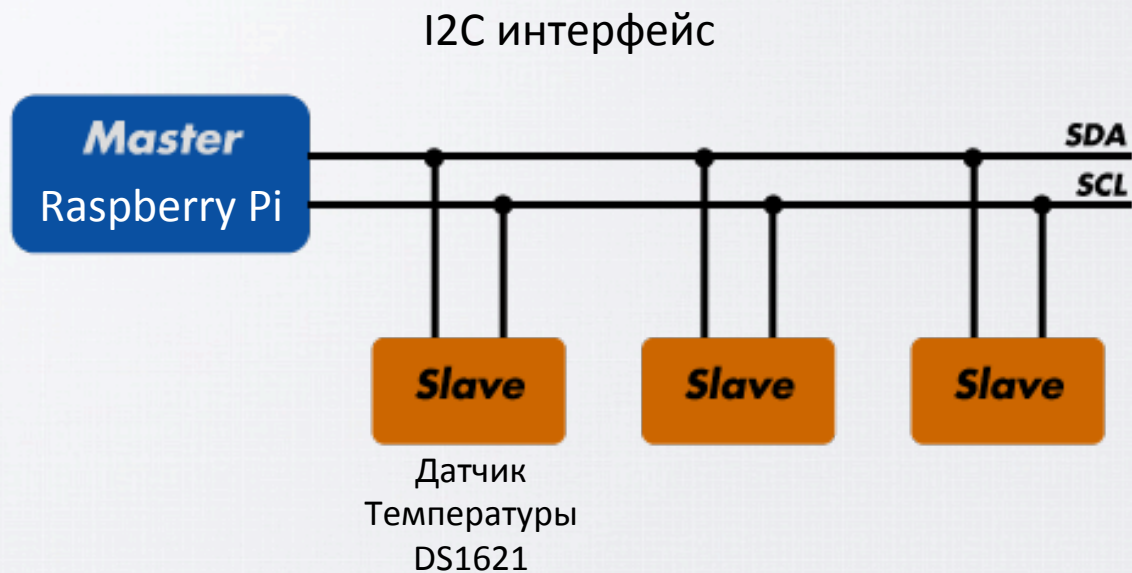
Owner: alexanderbelokrylov

## Process API: Триггер на завершение процесса

```
Process p = new ProcessBuilder("/usr/bin/java", "-version").start();
ProcessHandle process = p.toHandle();
CompletableFuture<ProcessHandle> onExit = process.onExit();

onExit.thenAccept(ph -> System.out.printf("PID %d terminated%n", ph.pid()));
```

# Process API для IoT





# Опрос I2C шины

- I2c-tools

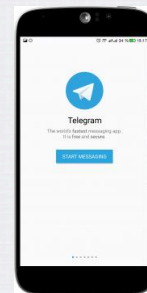
```
pi@raspberrypi:~/ $ i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60: 60  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

DEMO



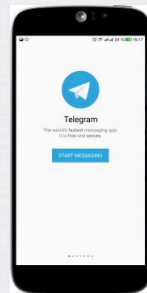


# CoolBeer Notification Demo





# CoolBeer Notification Demo



# CoolBeer Notification Demo

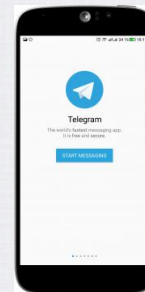


I2C

Raspberry Pi 3

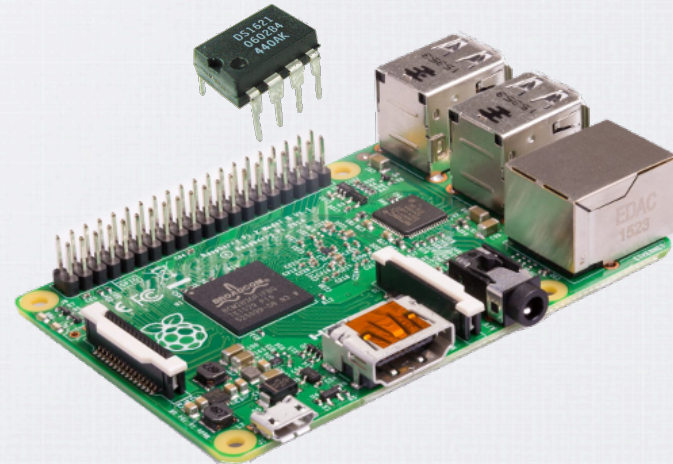


Java 9  
DIO  
Apache Flink  
Telegram Bot



# DIO (Device-IO API) example

- Connect DS1621 thermo sensor to RPi pins:
  - ground, 3.3v, I2C SDA/SCL
  
- Java coding:
  - open I2CDevice
  - write sensor config
  - get current temperature



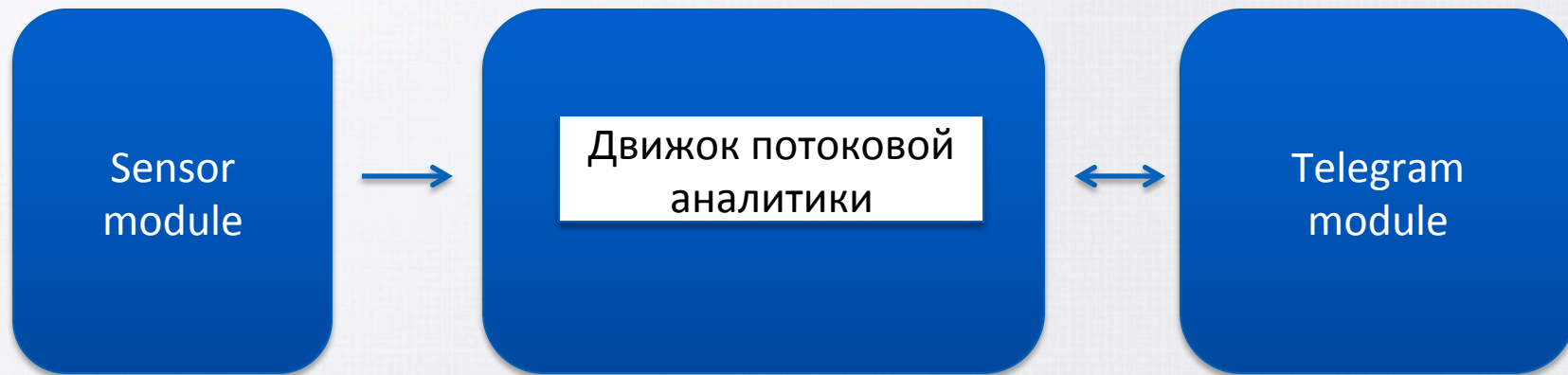
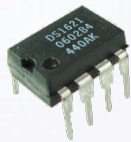
```
class DS1621 extends I2CSensor {
    private final int I2C_DEVICE_ADDRESS = 0x4f; // Sensor address
    // values from DS1621 Datasheet
    private final byte DS1621_SET_ACCESS_CONFIG = (byte)0xAC;
    private final byte DS1621_START_CONVERT = (byte)0xEE;
    private final int DS1621_READ_TEMPERATURE = 0xAA;

    DS1621() throws IOException {
        I2CDeviceConfig config = new I2CDeviceConfig.Builder()
            .setAddress(I2C_DEVICE_ADDRESS, I2CDeviceConfig.ADDR_SIZE_7)
            .build();
        i2cDevice = (I2CDevice)DeviceManager.open(config);
        write(DS1621_SET_ACCESS_CONFIG, (byte)0);
        write(DS1621_START_CONVERT);
    }

    byte getTemperature() throws IOException {
        byte[] data = read(DS1621_READ_TEMPERATURE, 1);
        return data[0];
    }
}
```

....

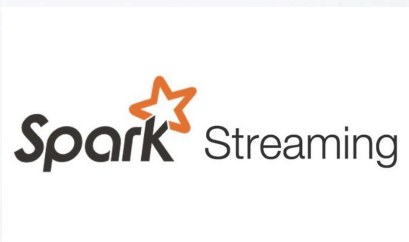
# Temperature stream processing





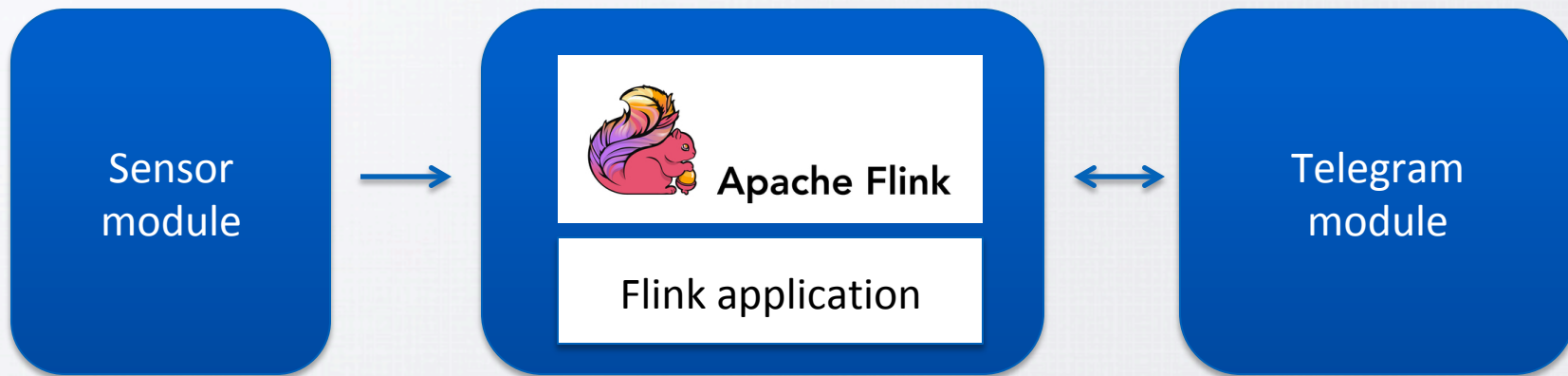
....

## Движки потоковой аналитики



....

# Temperature stream processing





## Выводы

- Java 9/10 позволяет использовать проверенные решения Stream Processing
- Функционал Java 9 отвечает требованиям IoT устройств
- OpenJDK получил код Java SE Embedded

Write once, run anywhere!

