**Edje Project**

The Software Foundation for IoT Devices

https://projects.eclipse.org/projects/iot.edje

© IS2T S.A. 2016. All rights reserved.
PRESENTER

Jérôme Leroux
Development and Innovation Manager at MicroEJ®
Edje Project Leader
10 years expertise in Java Embedded Systems

jerome.leroux@microej.com

The information contained herein is not warranted to be error-free.
IS2T® and MicroEJ® and all relative logos are trademarks or registered trademarks of IS2T S.A. in France and other Countries.
Java™ is Sun Microsystems’ trademark for a technology for developing application software and deploying it in cross-platform, networked environments. When it is used in this site without adding the “™” symbol, it includes implementations of the technology by companies other than Sun. Java™, all Java-based marks and all related logos are trademarks or registered trademarks of Sun Microsystems Inc, in the United States and other Countries.
Other trademarks are proprietary of their respective owners.
EDJE AND IOT INTRODUCTION
EDJE PROJECT

The Hardware Abstraction Java API for the IoT embedded systems

- Peripheral management
- Controller Communication Interfaces (Serial connection, CAN, SPI, I²C)
- Digital and Analog I/O (GPIO, ADC, DAC)

Scope

- Target resource-constrained micro-controller
- Provide ready-to-use software packages for target hardware
- Define a modular and easy to port framework

Project Status

- Eclipse IoT project
- Incubation
IOT TOPOLOGY

IoT DEVICES

GATEWAY

SERVER

8/16/32-bit MCU Mono-Core
Frequency: < 200 MHz
Flash: < 1 MB
RAM: < 512 KB

32/64-bit MPU Multi-Core
Frequency: in GHz
Flash: in GB
RAM: in GB

freeRTOS
IOT HARDWARE

STM32F4 Discovery

- Processor: 32-bits Cortex-M4 (STM32F407VGT6)
- Frequency: 168 MHz
- RAM: 192 KB
- Flash: 1 MB

Raspberry PI 2

- Processor: 32-bits quad-core Cortex-A7 (BCM2836)
- Frequency: 900 MHz
- RAM: 1 GB
- Flash: SD Card
EDJE APPLICATION FIELD

A library at the edge of the IoT for sensor hubs and devices
EDJE REQUIREMENTS
EDJE DEVICE CONFIGURATION

List of Java API

• The minimum execution environment provided by an Edje compatible device
• Intersection between Java SE, Java SE Embedded, MicroEJ and Android
• Includes java.lang, java.util, java.io, ...

![Diagram showing the EDJE DEVICE CONFIGURATION with intersections between J2SE, J2SE Embedded, MicroEJ, Edje, and Android]
EDJE DEVICE CONFIGURATION

List of Java API

- The minimum execution environment provided by an Edje compatible device
- Intersection between Java SE, Java SE Embedded, MicroEJ and Android
- Includes java.lang, java.util, java.io, ...

Diagram:
- Edje Application
- Edje API
- Edje Device Configuration
- Library
- Execution Environment
HARDWARE REQUIREMENTS

Minimal targeted Hardware

- Processor: 32-bits (e.g. Cortex-M0)
- Frequency: 16 MHz
- RAM: 32 KB
- Flash: 128 KB
EDJE API
EDJE API

Edje comes with the following services:

- Controller Communication Interfaces
- Serial Port
- CAN
- Analog Input
- Digital and Analog I/O
- GPIO
- USB Devices
- Peripheral Management
- LCD
- Analog Input
EDJE API

Peripheral Management

- List the peripherals of a platform
  - MCU peripherals (UART, timer, USB controller, ...)
  - Board peripherals (screen, button, LED, sensors, ...)
  - External peripherals (USB devices, bluetooth devices, ...)
- Peripheral plug/unplug notification system

```
PeripheralManager

list(peripheralType:Class): List<Peripheral>
register(peripheralType:Class, peripheral:Peripheral)
addRegistrationListener(listener:RegistrationListener, peripheralType:Class)
...
```

```
Peripheral

getName(): String
getDescription(): HardwareDescriptor
...

children *

HardwareDescriptor

getName(): String
getProperty(name:String): String
...
```

```
RegistrationEvent

getPeripheral(): Peripheral
...
```

```
RegistrationListener

peripheralRegistered(event:RegistrationEvent)
...
```
EDJE API

Controller Communication Interfaces

• Some peripherals can establish a connection to external devices
  o Serial Connection (UART)
  o Serial Peripheral Interface (SPI)
  o Inter-Integrated Circuit (I²C)
  o Controller Area Network (CAN)
• The peripheral implements Connectable interface
• Connection is described by a String
• Example with serial ports:

```java
List<SerialPort> serialPorts = PeripheralManager.list(SerialPort.class);

for (SerialPort serialPort : serialPorts) {
    Connection connection = serialPort.openConnection("baudrate=115200;bitsperchar=8");
    ...
    connection.close();
}
```
EDJE API

Digital and Analog I/O

• Manage controller pins
• API inspired from Arduino C API

• General Purpose Input/Output (GPIO)
  o LED, Buzzer, Button
• Analog to Digital Converter (ADC)
  o Potentiometer, Temperature Sensor, Light Sensor
• Digital to Analog Converter (DAC)
  o Speaker, Light Dimmer
• Pulse Width Modulation (PWM)
  o Motor
EDJE API

Digital and Analog I/O

- A pin is identified by the port and an ID
- Port name can be
  - MCU specific
EDJE API

Digital and Analog I/O

- A pin is identified by the port and an ID
- Port name can be
  - Board specific
EDJE API

Digital and Analog I/O

- A pin is identified by the port and an ID
- Port name can be
  - Standard

![Digital Arduino](image1)
![Analog Arduino](image2)

Eclipse IoT Day Grenoble 2016
REFERENCE IMPLEMENTATIONS
REFERENCE IMPLEMENTATIONS

Features

• Peripheral Management
• UART, USB CDC, GPIO, DAC, ADC

Hardware

• Raspberry-Pi 2
• Quad-Cortex-A7 @ 900 MHz
• RAM: 1 GB RAM

Platform

• Kura
• OpenJDK
• Linux
REFERENCE IMPLEMENTATIONS

Features

- Peripheral Management
- UART, USB CDC, GPIO, DAC, ADC

Hardware

- STM32F746G-DISCO
- Cortex-M7 @ 200 MHz
- RAM: 8 MB
- Flash: 16 MB

Platform

- MicroEJ OS
- FreeRTOS
- STM32Cube
REFERENCE IMPLEMENTATIONS

Features

• Peripheral Management
• UART, USB CDC, GPIO, DAC, ADC

PC Simulation

• Java® SE
• Hardware In the Loop Simulation (HIL)
EDJE ROADMAP
POTENTIAL ROADMAP

Features

• I²C, SPI
• Controller Area Network (CAN)
• Power Management
• Sensor

Reference Implementations

• MicroEJ Renesas Synergy Cortex-M4
• MicroEJ NXP Kinetis Cortex-M0+
CALL TO ACTION

WE NEED YOU
DEMO

STM32F746G-DISCO + Arduino Multi-function Shield = over MicroEJ

Eclipse IoT Day Grenoble 2016