



VERT.X

pour l'IoT dans l'embarqué (avec aussi du Pi4J)

Laurent HUET @lhuet35

Qui suis-je ?

- > 1 an dans la robotique industrielle
- 15+ ans en SSII
- 4 ans de bâtiment !



BreizhJug



Breizh C@mp

La conférence à l'Ouest

Sommaire

- Vert.x overview
- Aspects “Hardware”
- Code & Démonos



VERT.X

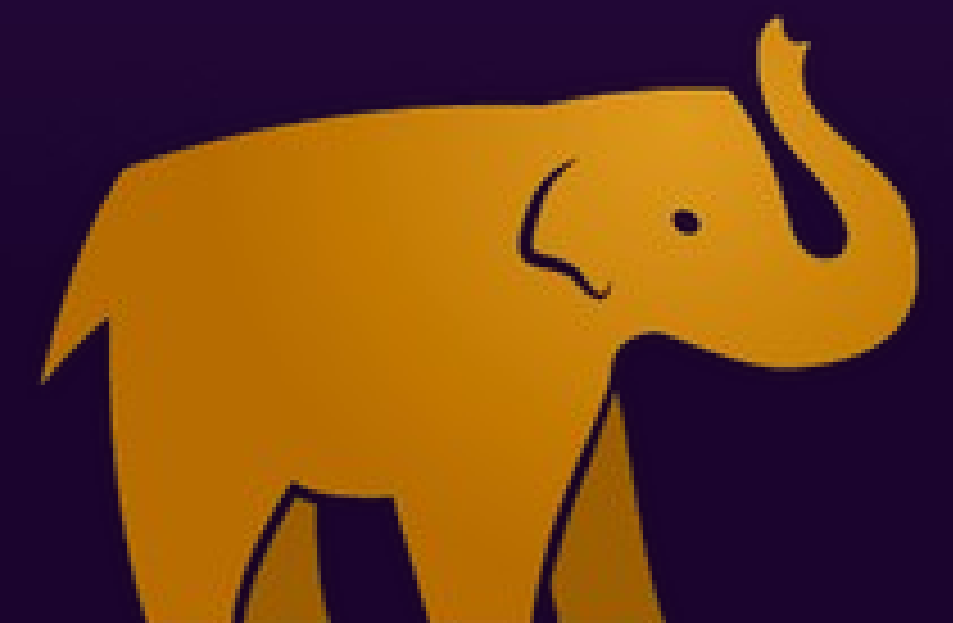
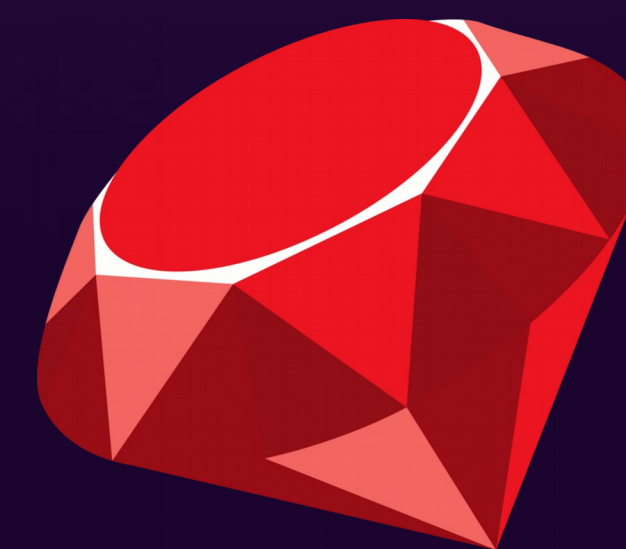
overview

VERT.X

Vert.x is a tool-kit for
building reactive
applications on the JVM.

“Qualités”

- “Event driven”
- “Non blocking”
- “Event loop”
- Polyglotte



Asynchrone vs Synchrone



Event loop

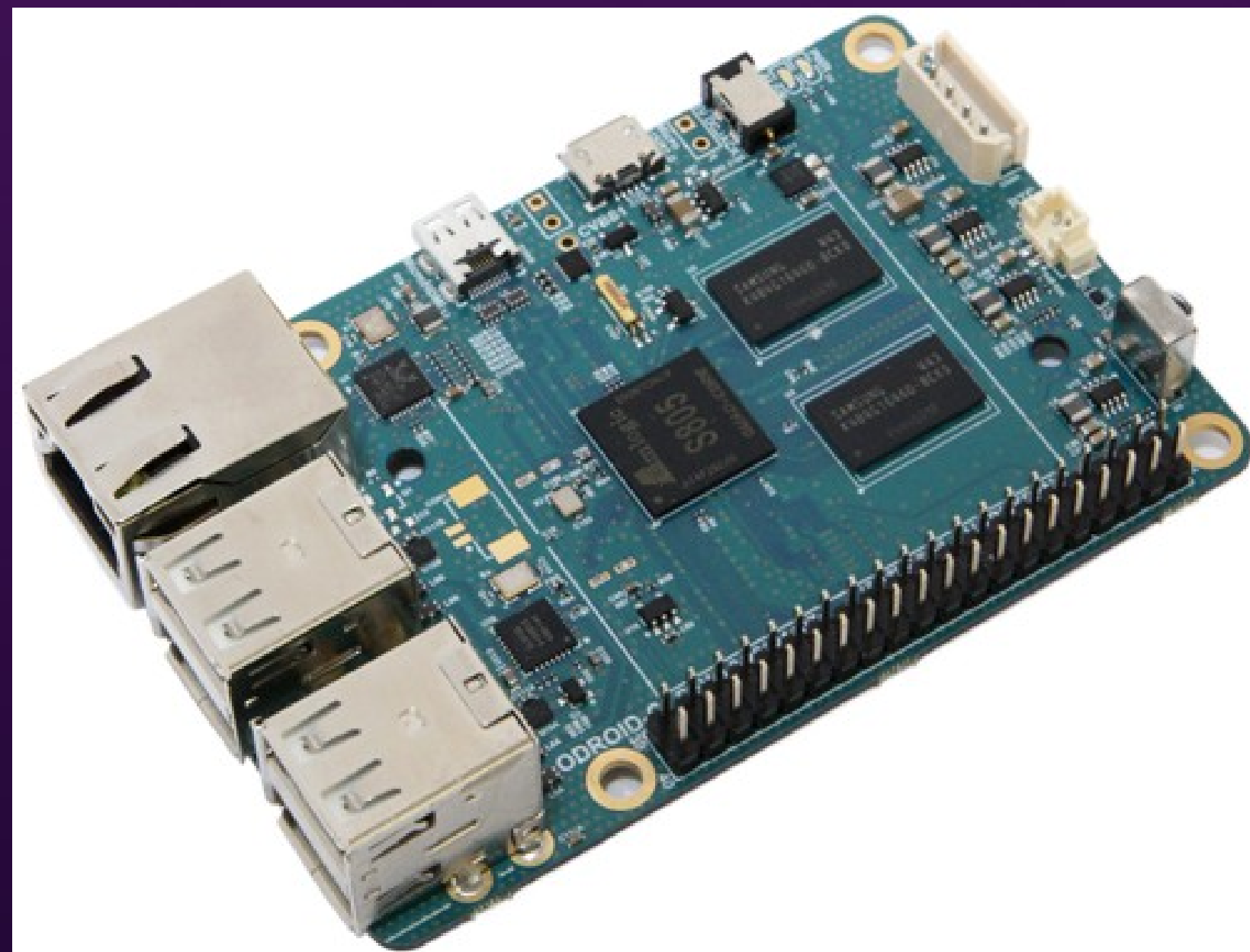
- Programmation mono-threadé
- Règle d'or : **NE JAMAIS BLOQUER** l'event loop
- API Bloquante
 - “Worker thread pool”

Event bus

- Système de messagerie de Vert.x
- Distribué
- 3 types de messages
 - Publish / subscribe
 - Point-to-point
 - Request-response

Hardware

Cartes ARM



Odroid C1

Quad Core ARMv7 @1.5 GHz

1 Go RAM

Gigabit Ethernet



Raspberry Pi2

Quad Core ARMv7 @900MHz

1 Go RAM

10/100 Mbps Ethernet

Systeme

- Linux (Ubuntu)
- Modules linux (i2c, 1-wire, GPIO)
- Librairie Pi4j
 - Version 1.1.0-SNAPSHOT !
- WiringPi
 - Portage sur Odroid dispo

GPIO

- Entrées / Sorties “digitales”



Bus

- UART
- 1-Wire
- I2C



Demo 1

“Hello World”

En résumé

- Code asynchrone
- Dépendances très limitées
- Accès aux GPIO via Pi4J

Demo 2

Verticle “Hello World”

En résumé

- Verticle
 - Brique de base pour “microservices”
- Cycle de de vie (start, stop)
- Démarrage via la classe Launcher
- ... et plein d'autres fonctionnalités

Demo 3

Event Bus

En résumé

- Event bus pour communiquer entre verticles
- Messages en mode « point-to-point »
- ... et toujours asynchrone

Demo 4

Bus 1-wire & Web

En résumé

- Appel d'API bloquante
- Composition de "Future"
- Serveur HTTP intégré à Vert.x
- Communication via l'Event bus
 - en mode "request/response"

Demo 5

Bus 1-wire & I2C

En résumé

- Bus I2C (Ecran LCD)
- Fichier de configuration JSON
- Message JSON en mode "publish/subscribe"
- Event bus distribué

Demo 6

Routeur Web

En résumé

- API REST avec un module Web complet
 - ~ Express
 - Performant

Demo 7

Web & Websocket

En résumé

- Event Bus distribué (dans le browser)
- Application Web « complète »



Ce que vous n'avez pas vu

Event Bus bridge

- Multiples technos clientes
 - C++
 - Android
 - JavaScript (Node par ex.)
 - TCP

Fonctionnalités avancées

- HA
- “Worker verticle”
- Mix de langages
- CLI
- “Shared data”
- Vertx-unit

Ecosystème de modules

- Accès base de données (JDBC, Mongo, Redis, ...)
- Vert.x RX
- Vert.x Shell
 - SSH / Telnet dans votre appli
- Vert.x awesome
 - <https://github.com/vert-x3/vertx-awesome>

Code des démos

<https://github.com/lhuet/vertxOnRpiAndOdroid>



Merci
de votre attention

Q / A