

## JavaScripting STM32

### Agile End to End Programming with Hop.js

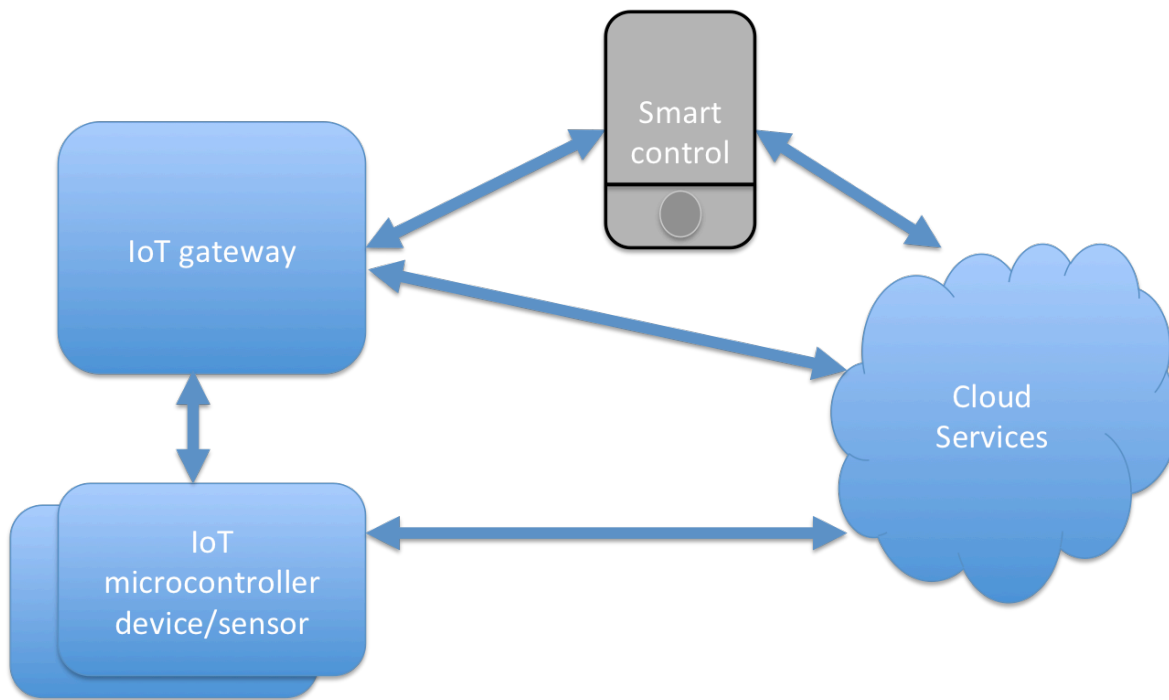
Vincent Prunet

Hop.js Founder and CEO



# Application Software Challenges 2

## Long and Costly Development Cycles



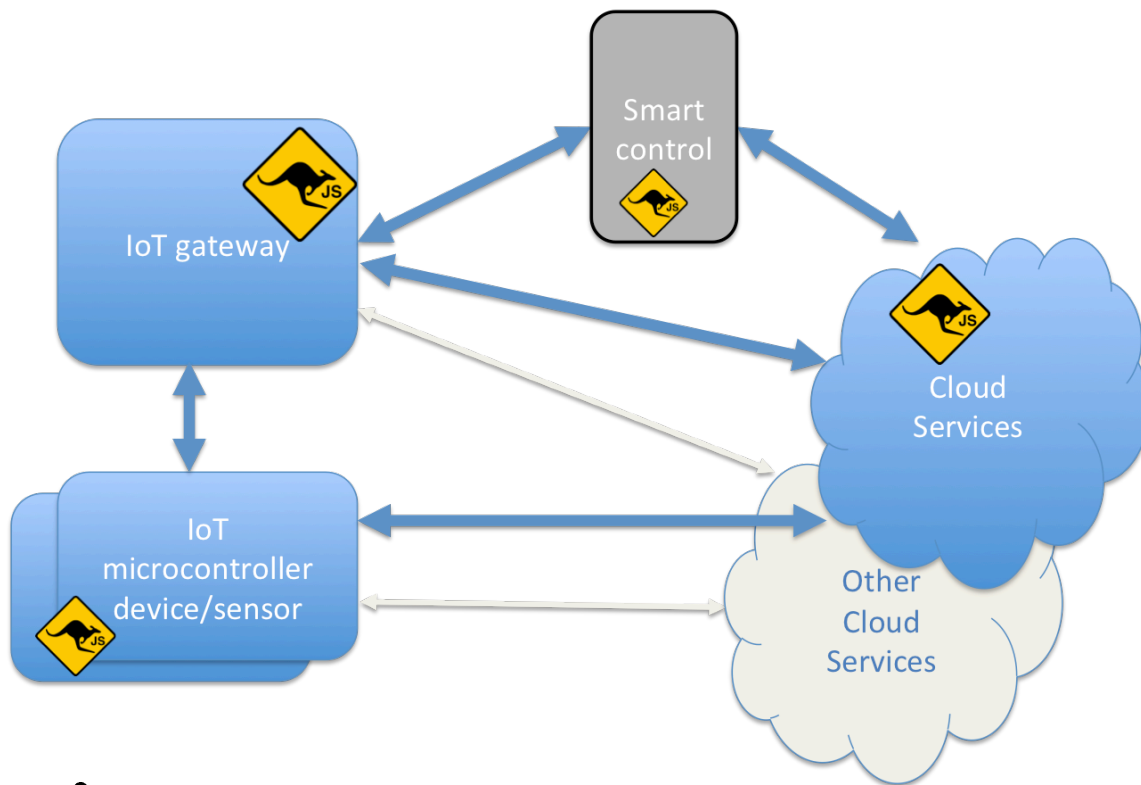
Heterogeneous Platforms

Multiple Programming Languages and Tools

Distributed Architecture / Protocols  
Add a new feature is hard



## Distributed JavaScript Framework for IoT Software Applications



Cross Platform Distributed Framework

Ready to Use

For Agile Teams



# Hop.js Features

## Cross Platform and Ready to Use

4

### Programming Language Concepts

Multi-tier JavaScript  
Remote Services  
Server Events

Same Language/API on  
Server and Client

### Runtime Environment

**Server:** multi threaded  
JavaScript engine, Hop.js  
compiler, builtin secure http  
server. Runs on Linux/Posix

**Client:** JavaScript enabled  
MCU [STM32F407], any  
smartphone, PC

### Interoperability

JavaScript server and  
frontend modules

Third Party Cloud/Web  
Services



# Hop.js Benefits

5

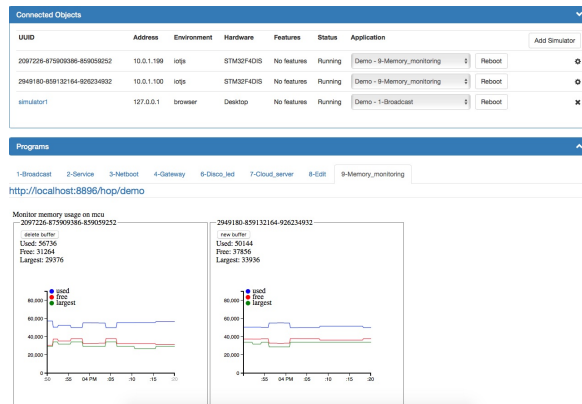
Lower the barrier to IoT development

Embed the business logic

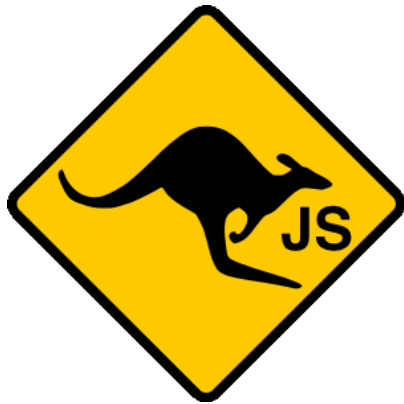
Reduce time and costs of application development and upgrades

Improve the user experience

Secure the solution



# Hop.js Roadmap



- Evaluation kits for linux SBC, PC and STM32F4. Just ask now
- Developer Studio - S2 2017
- Runtimes for MCU - 2017-2018
- Software Components for Business Applications - general availability in 2018

Business Information – IoT Product line, Licencing - Mail to  
[Vincent.Prunet@gmail.com](mailto:Vincent.Prunet@gmail.com)

Technical Information on Hop.js language and free download of the  
server application at <http://hop.inria.fr>

