OpenWSN

Important!

Implementing the Internet of Things!

Cedric Adjih
Inria

Thomas Watteyne
Research Lead, OpenWSN project, UC Berkeley

Workshop Internet Of Things / Equipex FIT IoT-LAB
6 November 2014, Montbonnot, France

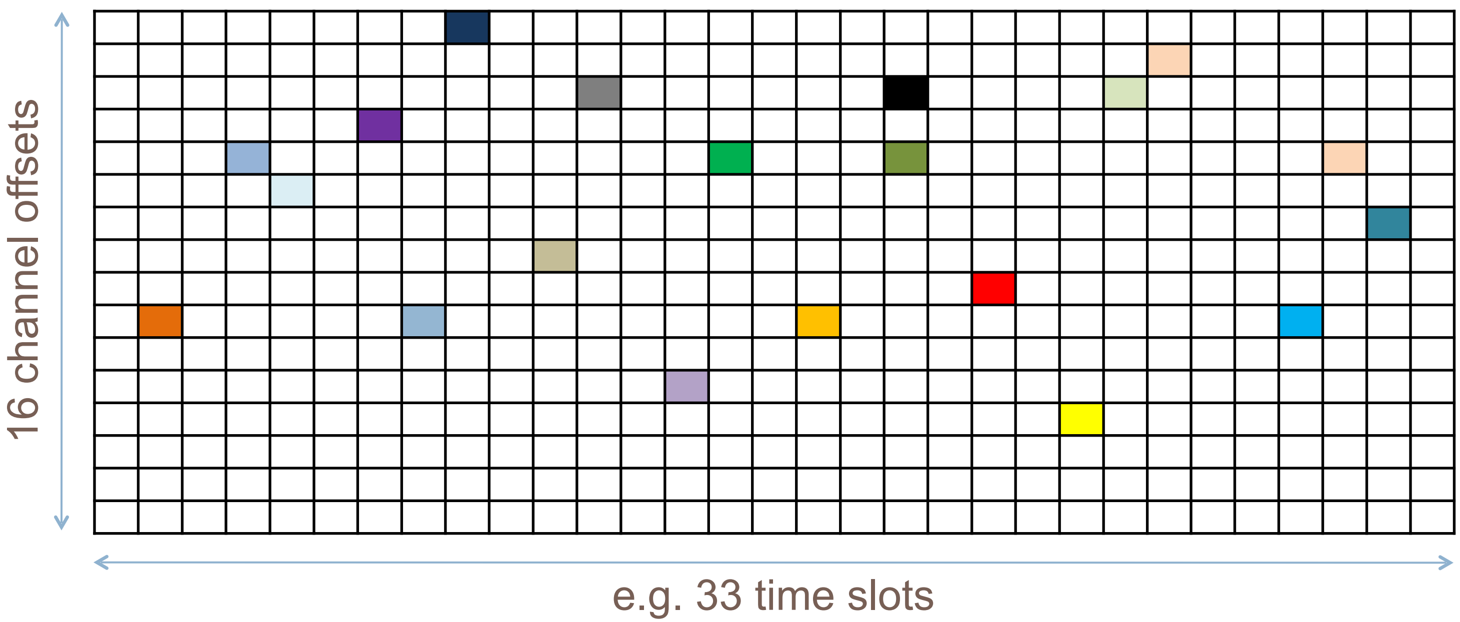
Outline

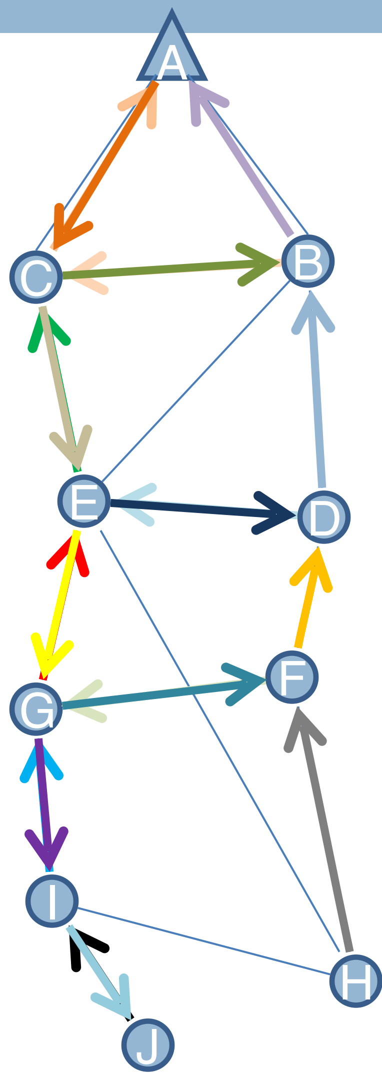
- Why OpenWSN?
- Technical Overview
- Example Projects
- Opportunities and Road Ahead

Time Synchronized Channel Hopping

- Motes are synchronized
- Communication follows a schedule
- Schedule gives tunable trade-off between
  - packets/second
  - latency
  - robustness

  } …and energy consumption





Time Synchronized Channel Hopping

2006

Time
Synchronized
Mesh
Protocol

DUST NETWORKS

2007

Wireless
Extension to
HART

de-facto standard for process control

WirelessHART

2009

Integrates IPv6 technology

ISA100 Wireless COMPLIANT

2012

IEEE802.15.4e
Time
Slotted
Channel
Hopping

IEEE

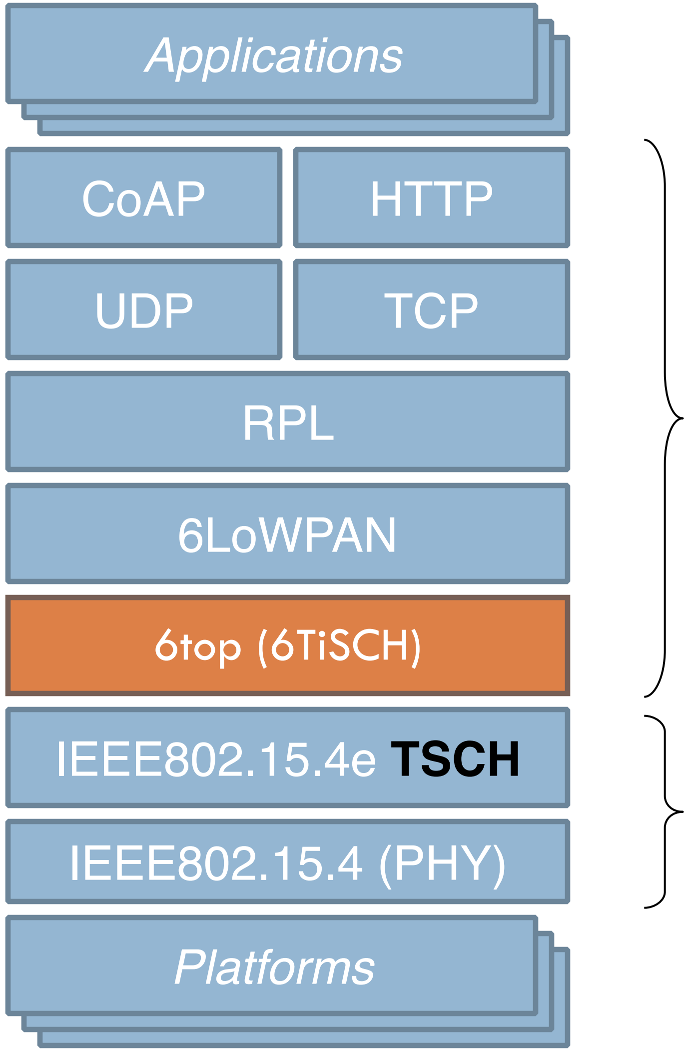
tens of thousands of networks → proven technology

- Clean layer 2 technology: can combine with IoT upper stack
- Enables different scheduling approaches, identify **limits**

Open Source Implementation — OPENWSN (California, BERKELEY)

Standardization — 6TiSCH

OpenWSN.berkeley.edu



Open Source Collaboration Tools

documentation

ticketing

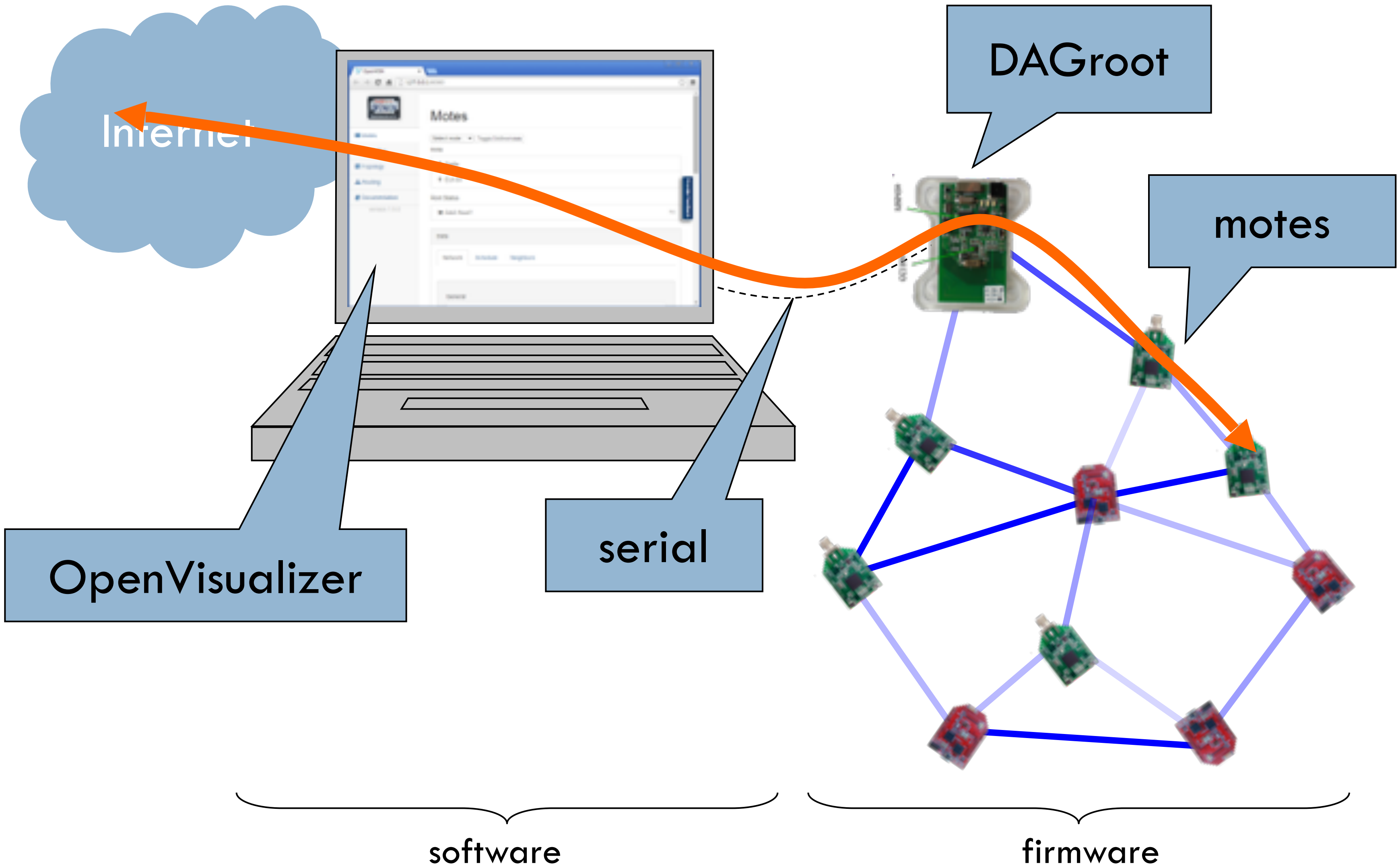
source code
(GitHub)

openwsn.berkeley.edu

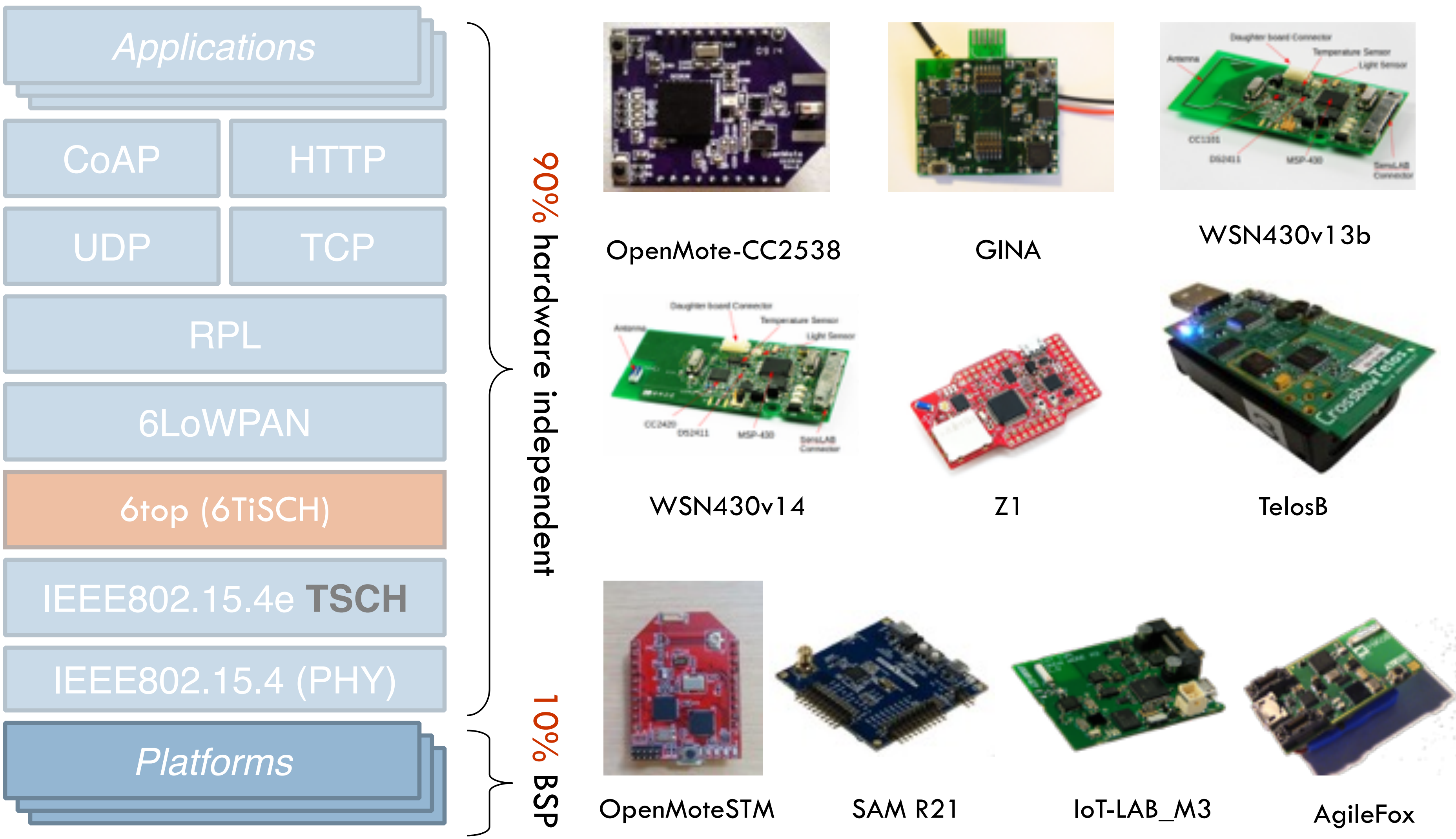
continuous integration
(Travis-CI and Jenkins-CI)

source code documentation

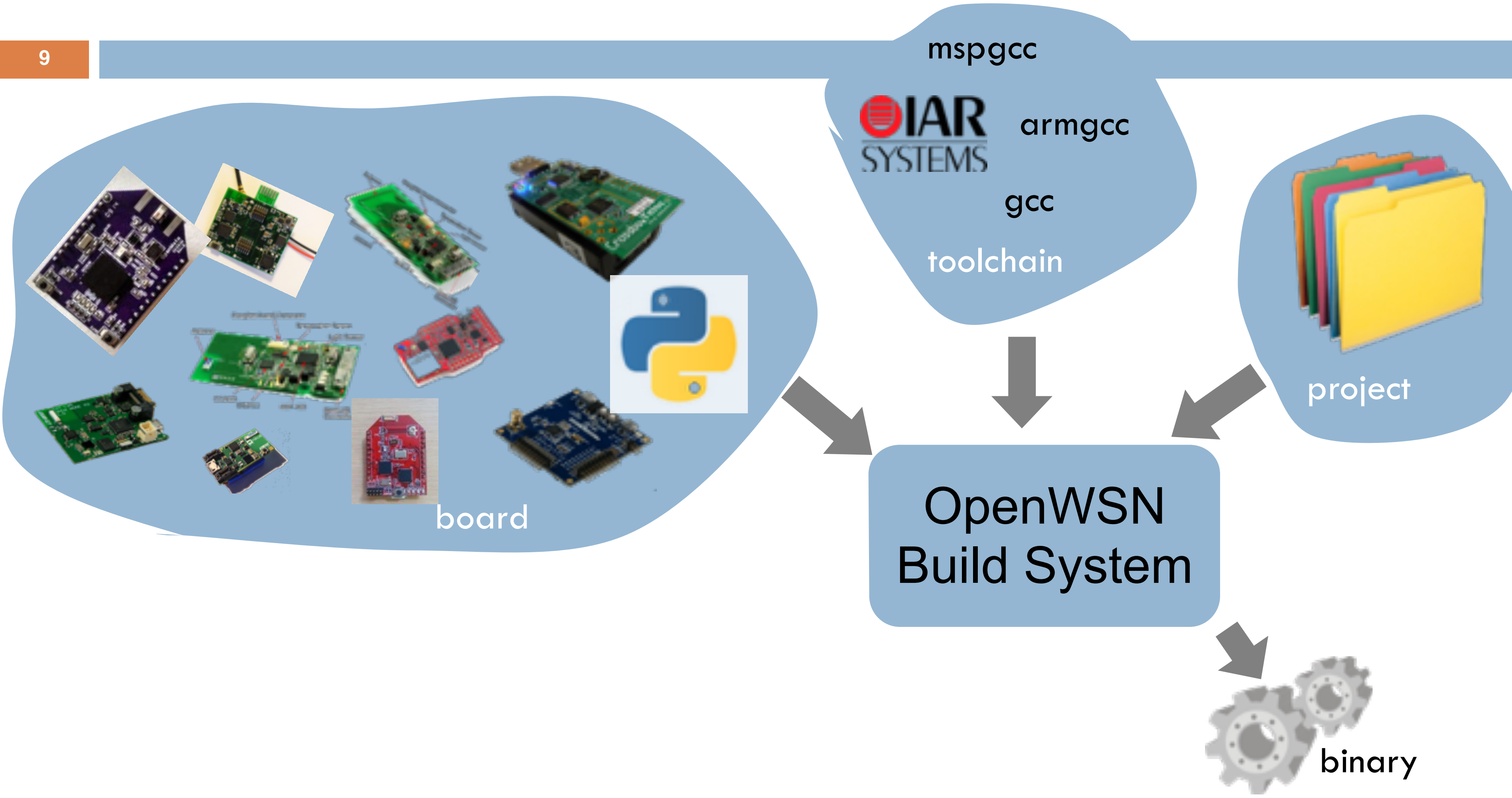
Architecture



Porting OpenWSN

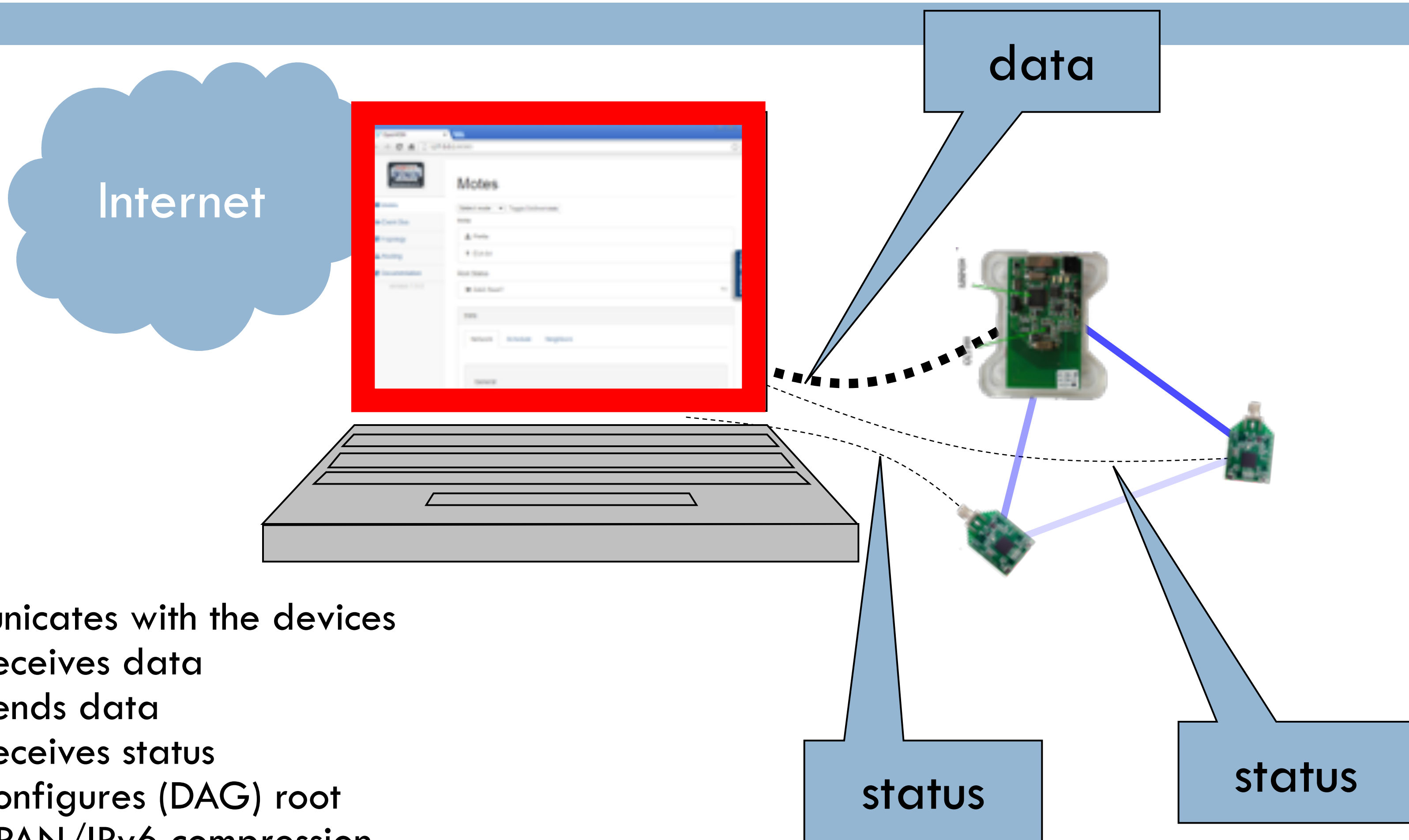


Build System

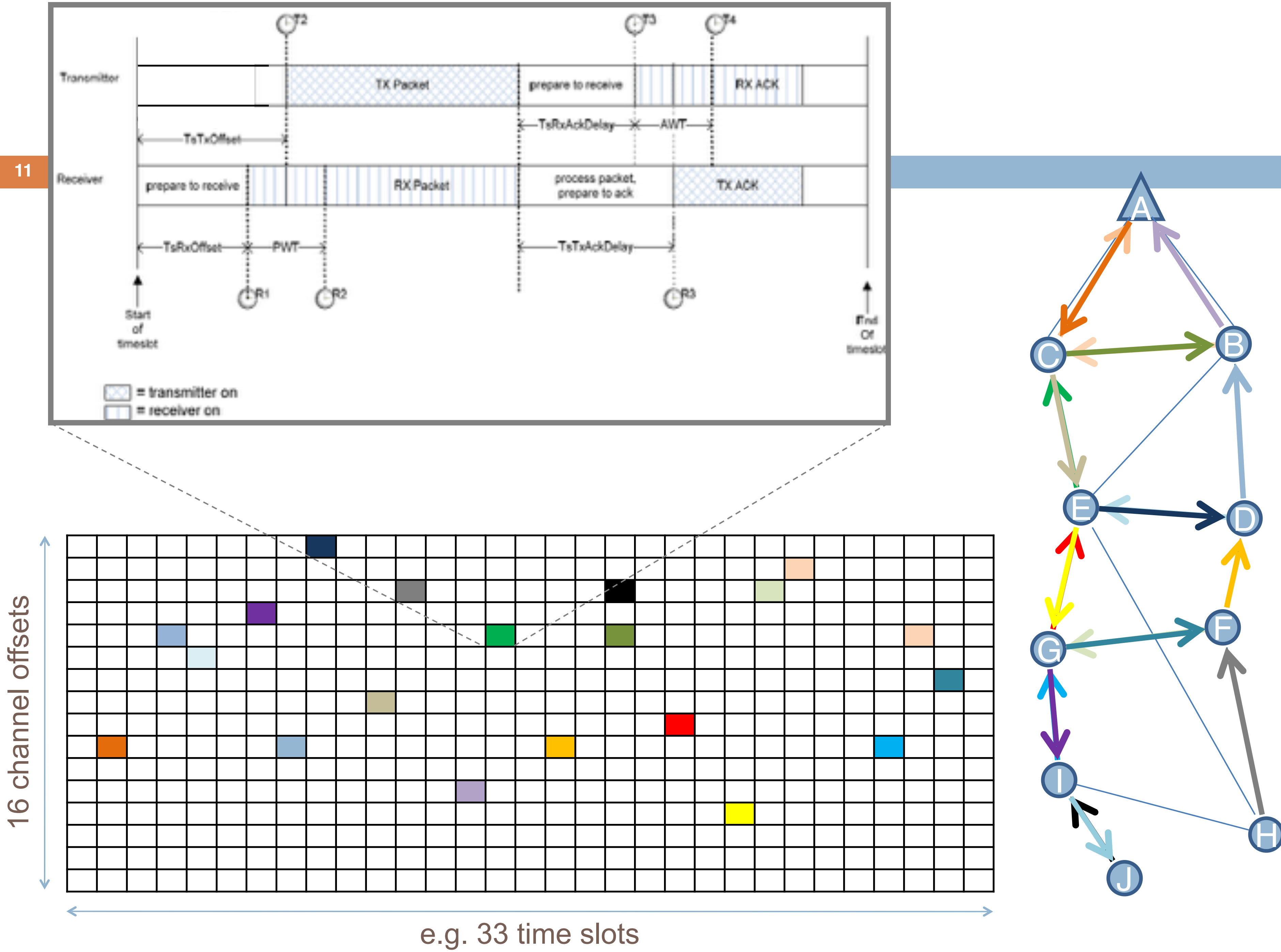



```
scons board=<board> toolchain=<toolchain> <project>
```

OpenVisualizer



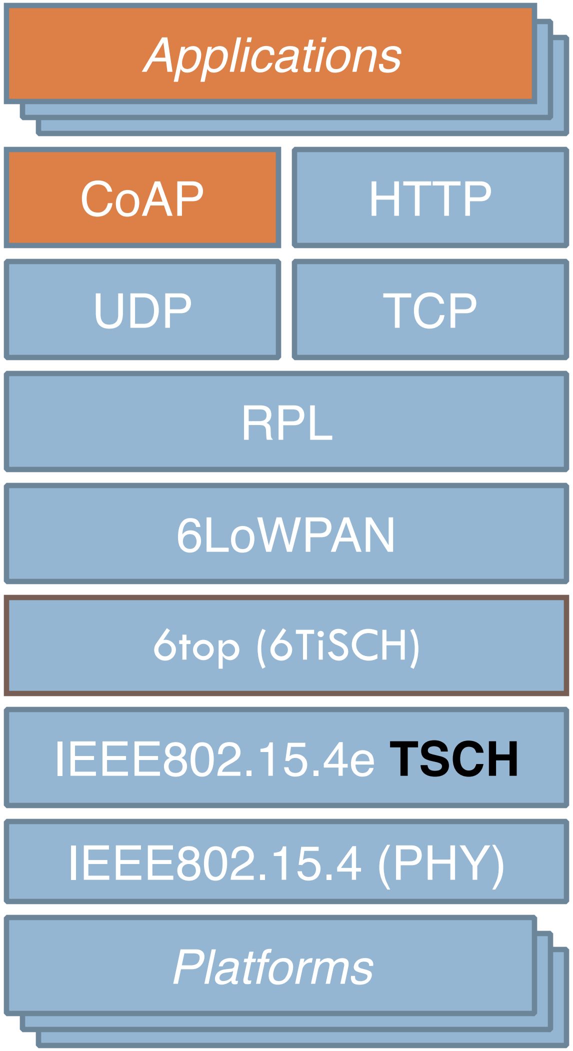
- communicates with the devices
  - receives data
  - sends data
  - receives status
  - configures (DAG) root
- 6LoWPAN/IPv6 compression
- RPL source route calculation
- written in Python



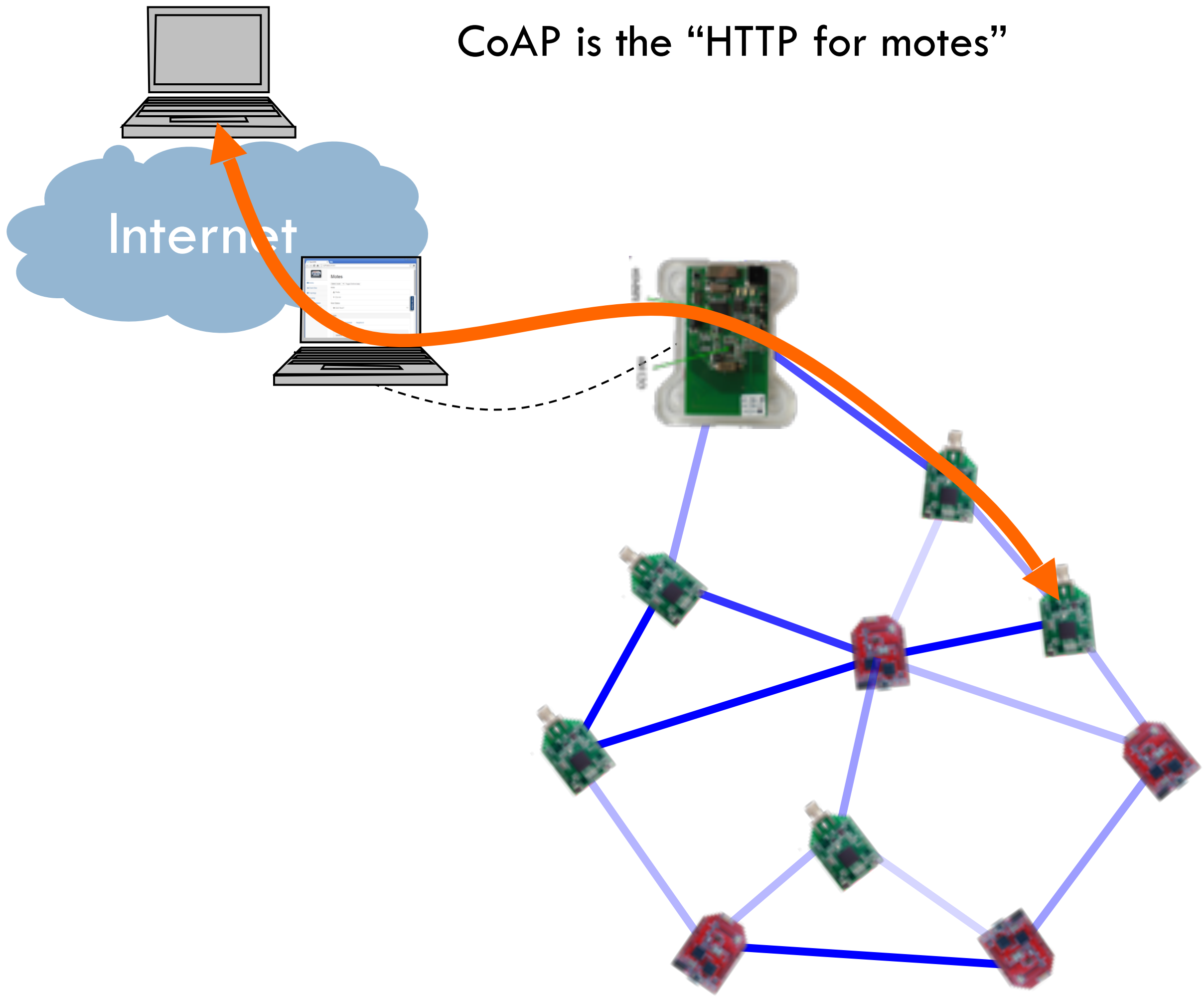
Transmitter
TX Packet
prepare to receive
RX ACK
TsTxOffset
TsRxAckDelay
AWT
Receiver
prepare to receive
RX Packet
process packet, prepare to ack
TX ACK
TsRxOffset
PWT
TsTxAckDelay
Start of timeslot
End Of timeslot
= transmitter on
= receiver on
16 channel offsets
e.g. 33 time slots
A
B
C
D
E
F
G
H
I
J

TX
8/19/2011 6:38 PM
state
TXDATA OFFSET
TXDATA PREPARE
TXDATA READY
TXDATA DELAY
TXDATA
RXACK OFFSET
RXACK PREPARE
RXACK READY
RXACK LISTEN
RXACK
TX PROC
SLEEP
variables
*dataToSend
*dataReceived
*ackToSend
*ackReceived
lastCapturedTime
syncCapturedTime
timing & events
tt1
tt2
tt3
tt4
tt5
tt6
tt7
tt8
delay Tx
wdDataDuration
maxRxAckPrepare
delay Rx
wdAckDuration
wdRadioTx
maxTxDataPrepare
pins
fsm
task
isr
radio
waste!
timeslot template
TsTxOffset
TsShortGT
TsSlotDuration
TsLongGT
TsTxAckDelay
RX
RXDATA OFFSET
RXDATA PREPARE
RXDATA READY
RXDATA LISTEN
RXDATA
TXACK OFFSET
TXACK PREPARE
TXACK READY
TXACK DELAY
TXACK
RX PROC
rt1
rt2
rt3
rt4
rt5
rt6
rt7
rt8
maxRxDataPrepare
maxTxAckPrepare
wdRadioTx

Interacting with motes



CoAP is the “HTTP for motes”



OpenWSN on IoT-LAB

- WSN430 port
  https://openwsn.atlassian.net/wiki/display/OW/WSN430
- IoT-LAB_M3 Port (+AgileFox)
  https://openwsn.atlassian.net/wiki/display/OW/IoT-LAB_M3
- Automated exp. on IoT-LAB
  https://github.com/adjih/exp-iotlab
- IoT-LAB tutorial
  (w/ architecture, example of cstorm.c, tests, fixes, and a lot more)

  https://openwsn.atlassian.net/wiki/display/OW/Running+a+Network

  https://openwsn.atlassian.net/wiki/display/~gaillard/Running+OpenWSN+experiments+on+IoT-Lab+M3+nodes


Isabel Vergara


Franck Rousseau


Pedro Helou


Oana Iova


Adilla Susungi

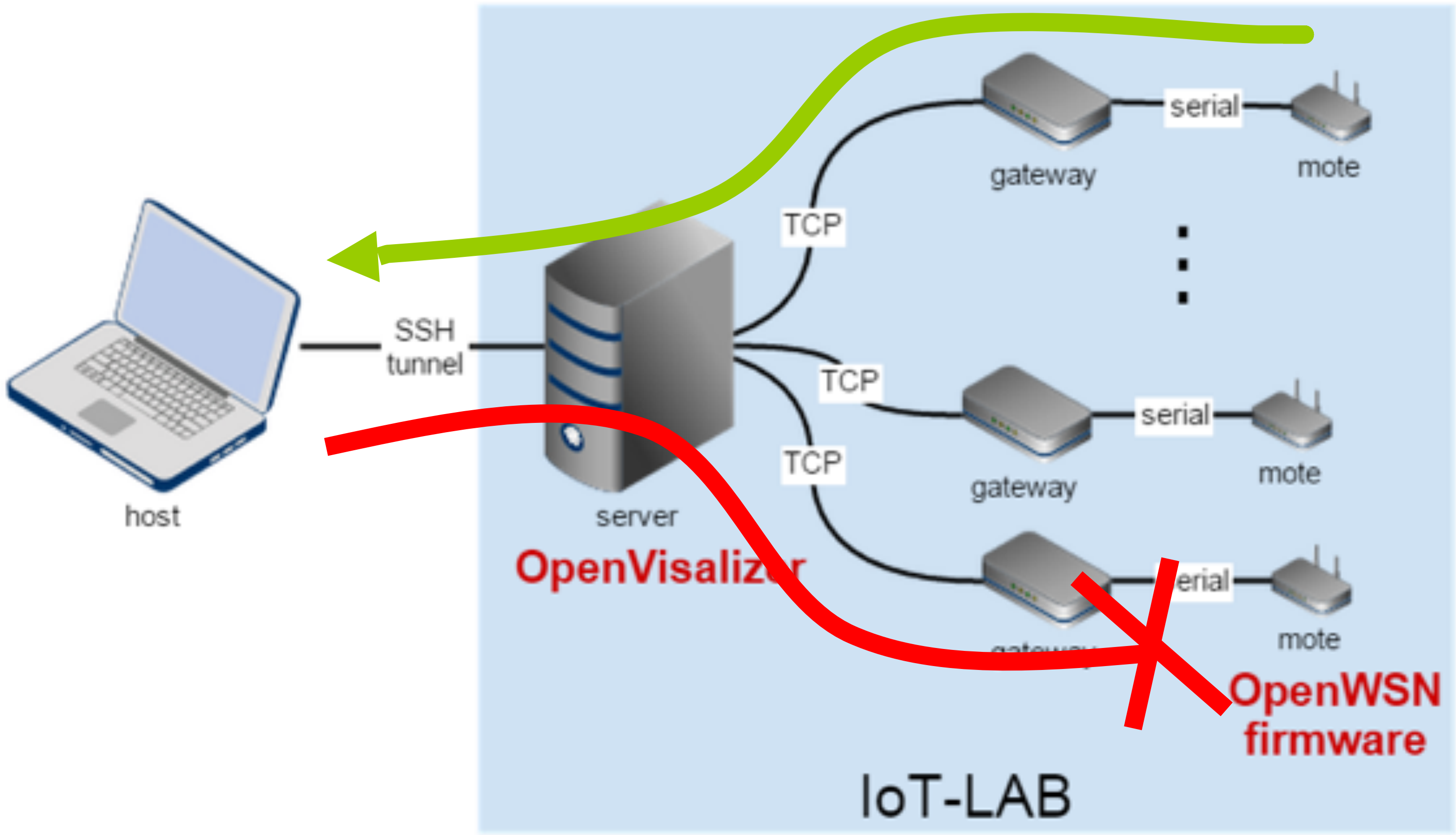

Fabrice Theoleyre


Alaeddine Weslati

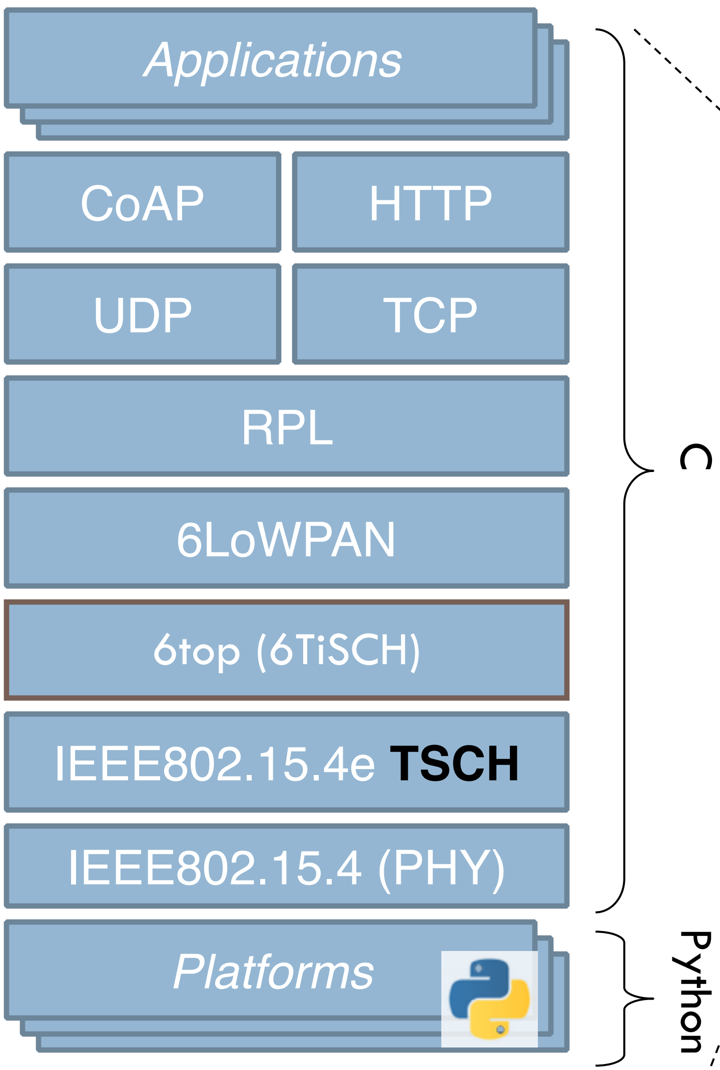

Cedric Adjih

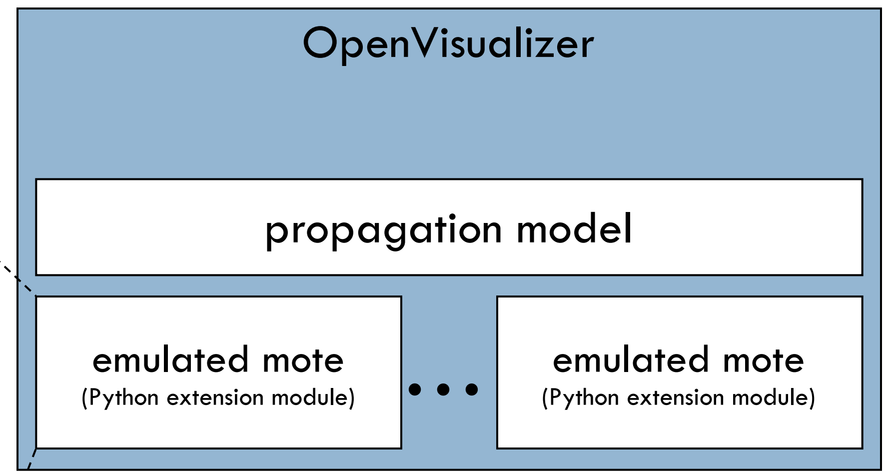

Guillaume Gaillard

OpenWSN on IoT-LAB



OpenSim Emulator



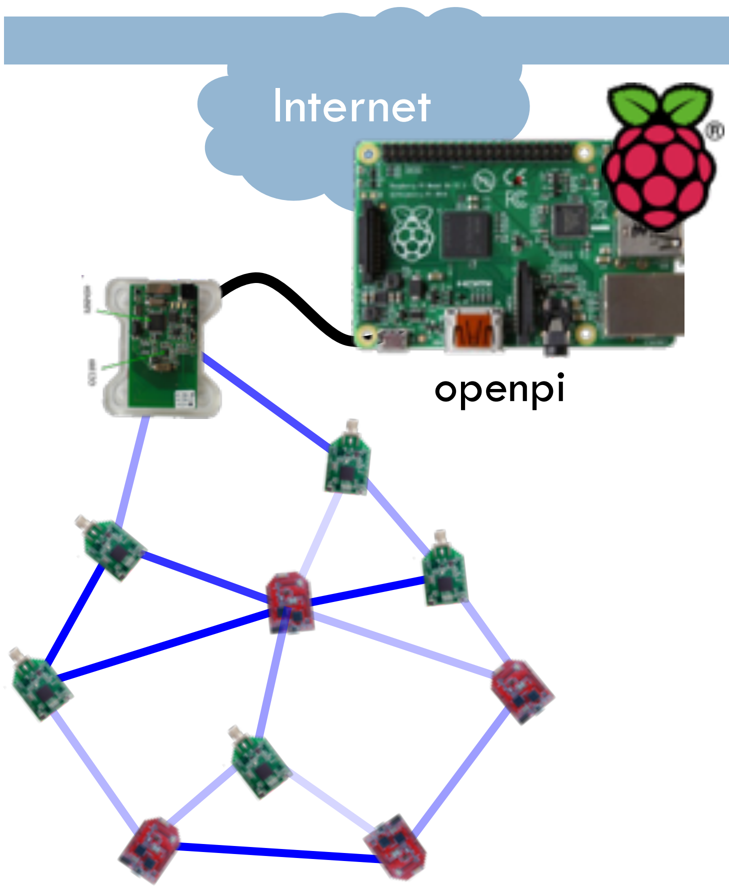


- same interface as real motes
- omniscient debugging:
  - see all packets (Wireshark)
  - see all pin activity (logic analyzer)
- models:
  - clock
  - propagation/topology

```
scons board=python toolchain=gcc oos_openwsn
```

```
openVisualizerWeb.py --sim --simCount=3
```

OpenPi



• OpenVisualizer pre-installed on Raspberry Pi image
• built nightly with latest OpenVisualizer

openpi.openwsn.org

OpenVM

• run with (free) vmware Player
• all toolchains pre-installed

openvm.openwsn.org

{Open}Projects

some examples

OpenMote

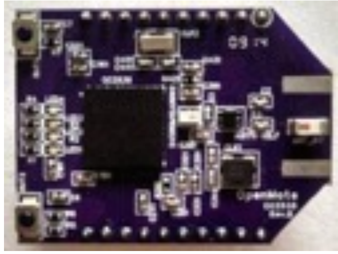
Open hardware for the Internet of Things.

Xavi Vilajosana

Pere Tuset

- An easy-to-use ecosystem of IoT hardware
- Centered about the OpenMote-CC2538

OpenMote	**OpenBase**	**OpenBattery**
TI CC2538 SoC (Cortex M3 + radio) 4 LEDs, 2 Buttons 2 antennas	Ethernet PHY+MAC USB-to-UART port 10-pin ARM JTAG	Temp./Humd. Acceleration Light 2xAAA batteries



www.openmote.com

Mercator

Constanza Perez Garcia

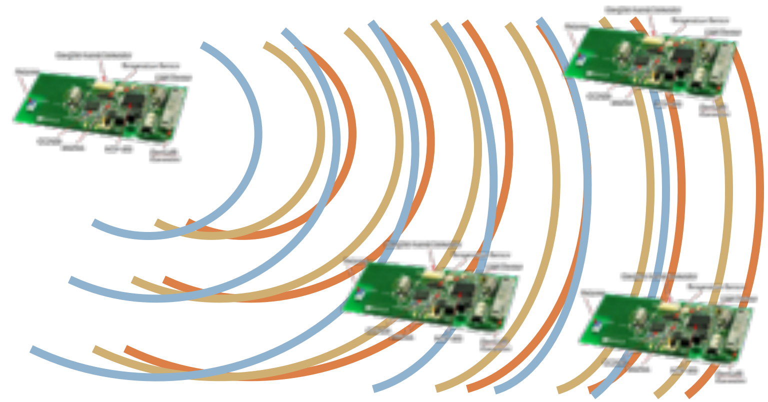
Diego Dujovne

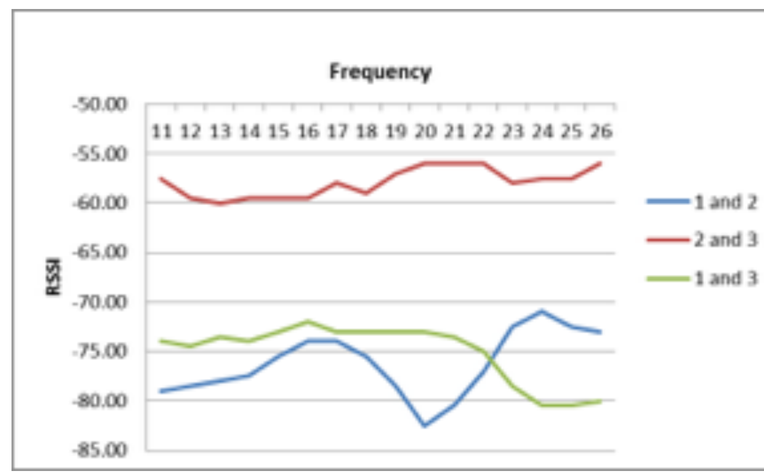
Universidad Diego Portales, Chile

Dense Wireless Connectivity Datasets for the IoT

Mercator is a collection of connectivity datasets gathered on the IoT-LAB sites. These datasets are:

- **dense in time**, variation of connectivity over time.
- **dense in space**, how connectivity is affected by the location of transmitter and receivers.
- **dense in frequency**, how connectivity is affected by the communication frequency.





https://github.com/openwsn-berkeley/mercator

RIOT integration

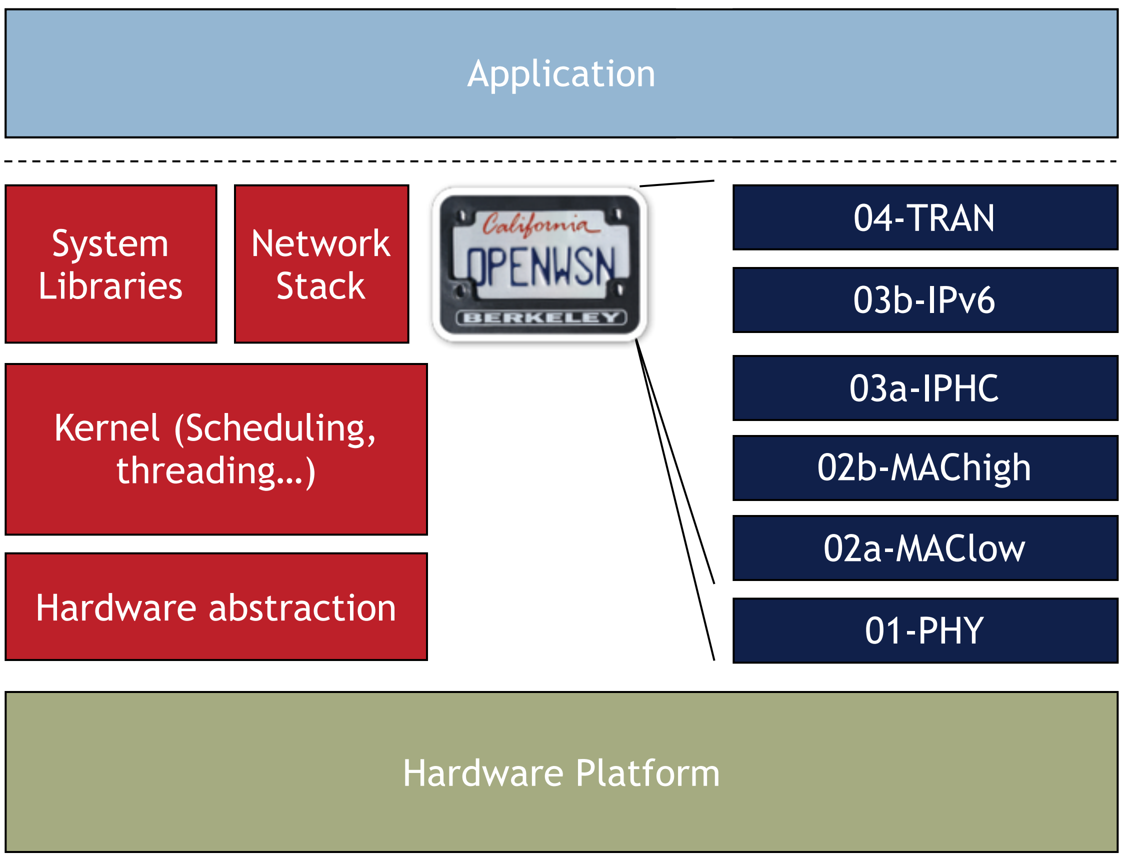

Thomas Eichinger


Oliver Hahm


Emmanuel Baccelli

Inria/Freie Universitat Berlin, Germany

- Goal: combine the RIOT preemptive scheduler with the OpenWSN protocol stack



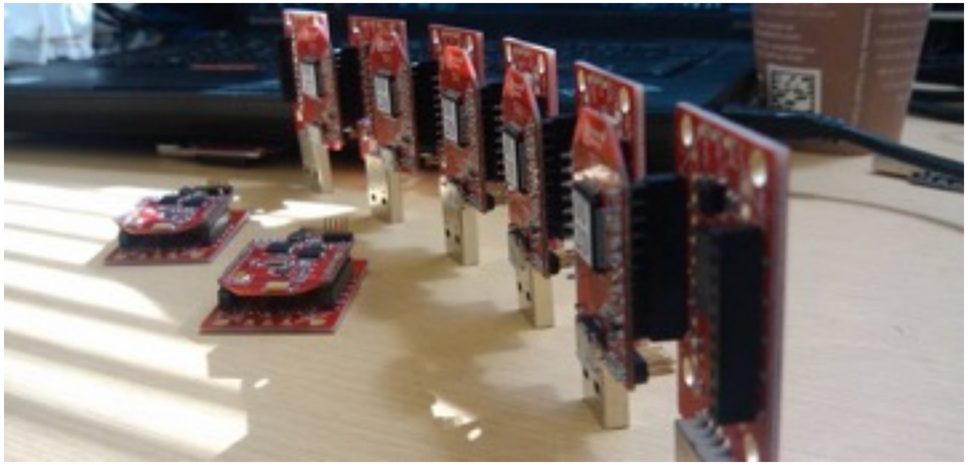
Demonstrated during 6TiSCH plugfest at IETF90, Toronto, July 2014.

OpenMote-STM

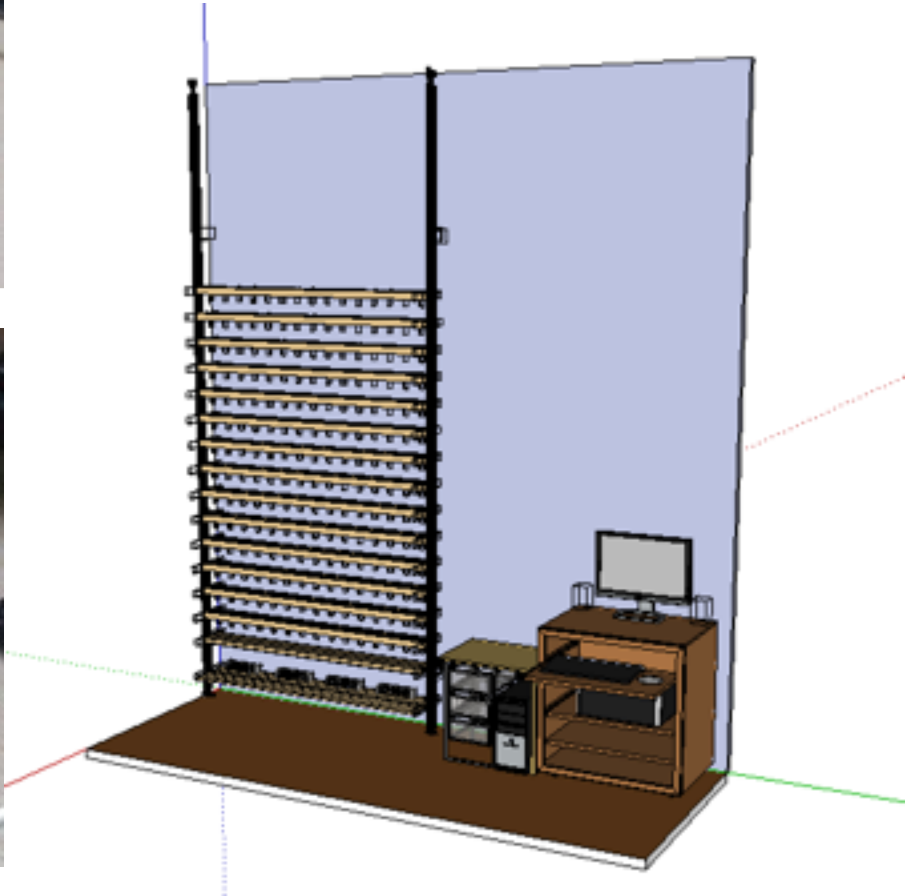

Tengfei Chang

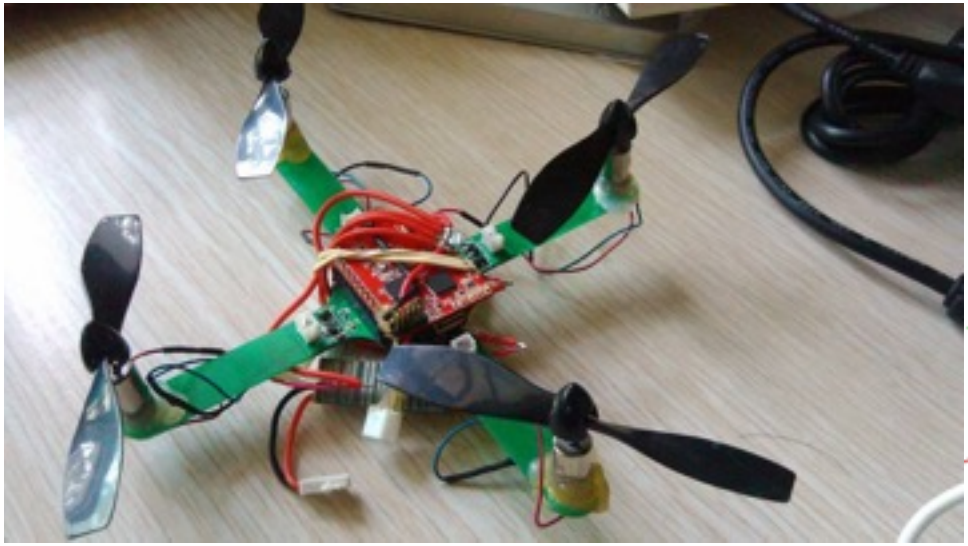

Qin Wang

University of Science & Technology Beijing, China



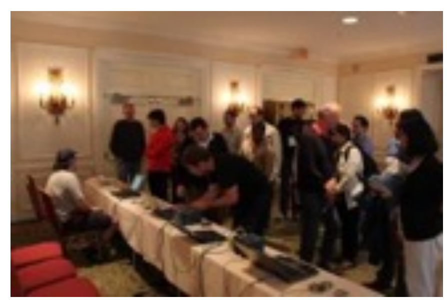
- STM32 micro-controller, Atmel AT86RF231 radio
- XBee-compliant pin-outs





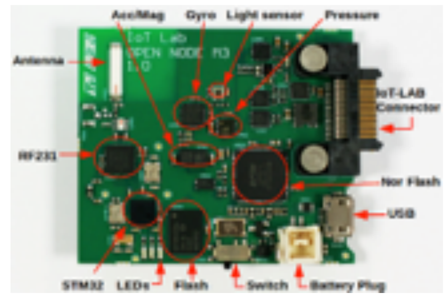
6TiSCH plugfests

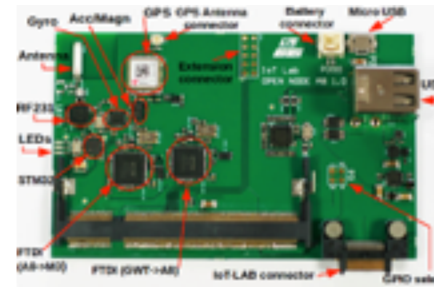
IETF89, London, March 2014



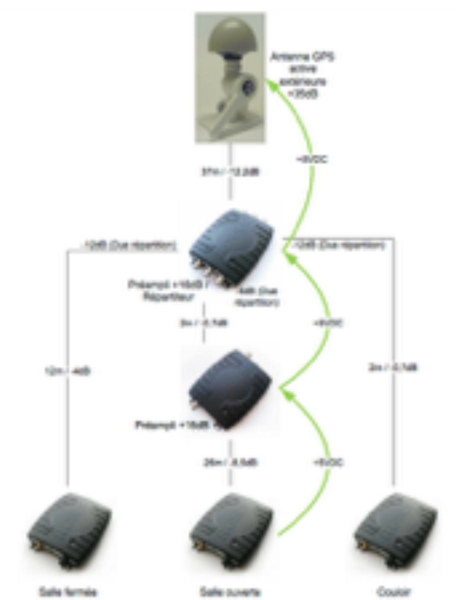
IETF90, Toronto, July 2014

Next step for IoT-LAB





- M3 != M3-A8
  - "Port" on M3-A8
  - Update of moteProbe to run on open-A8
- IoT-LAB tools
  - Sniffers with timestamps
  - Wireshark 6TiSCH dissector
- Advanced 6TiSCH architectures
  - GPS time sync.



Road Ahead

- [Q4 2014] FreeRTOS support (RIOT?)
- [Q1 2015] 6TiSCH Minimal security
- [Q2 2015] 6TiSCH interop
  - By ETSI, IETF93 Prague, July 2015
- [Q3 2015] Cloud-based (re)programming
- [Q4 2015] distributed scheduling

- Open for porting opportunities
- Open for collaboration

Contributors

- https://openwsn.atlassian.net/wiki/display/OW/Team
- Adilla Susungi, Ahmad Dehwah, Alaeddine Weslati, Andrew Pullin, Ankur Mehta, Anita Flynn, Boyang Zhang, Branko Kerkez, Constanza Pérez García, Cedric Adjih, Charles McParland, Chol Su Kang, Christopher Snyder, Chuang Qian, David Burnett, David Stanislowski, Diego Dujovne, Edmund Ye, Emmanuel Baccellli, Emily Chen, Fabien Chraim, Fabrice Theoleyre, Franck Rousseau, Giuseppe Piro, Giuseppe Ribezzo, Guillaume Gaillard, Hilfi Alkaff, Isabel Vergara, Jonathan Simon, Kazushi Muraoka, Ken Bannister, Kevin Weekly, Kris Pister, Laura Keys, Leonid Keselman, Marcelo Barros de Almeida, Mathivanan, Michael Lin, Min Ting, Nahir Sarmicanic, Nicola Accettura, Oleksiy Budilovsky, Oliver Hahm, Pascal Thubert, Pedro Issa Helou, Peng Du, Pere Tuset, Qin Wang, Ricardo Cervera-Navarro, Russ Tremain, Sahar Mesri, Tengfei Chang, Thomas Eichinger, Thomas Watteyne, Vinoth Kumar, Vincent Ladeveze, Vitor Garbellini, Xavi Vilajosana.
- *if I left you off the list, send me an email! It wasn't intentional.*


California
OPENWSN
BERKELEY
California
OPENWSN
BERKELEY
California
OPENWSN
BERKELEY
California
OPENWSN
BERKELEY
California
OPENWSN
BERKELEY
California
OPENWSN
BERKELEY
California
OPENWSN
BERKELEY
California
OPENWSN
BERKELEY
California
OPENWSN
BERKELEY

Cedric Adjih

Cedric Adjih is currently a researcher in the Infine team of Inria Saclay-Île-de-France. He received his PhD diploma in 2001 from Versailles University (France). He graduated from ECP (Ecole Centrale de Paris) in 1994.

His research interests have focused on wireless multihop networks (ad-hoc networks, mesh networks, sensor networks). He has studied various aspects related to performance evaluation, algorithm and protocol design, and practical experimentation, on different subjects, including: efficient routing, reliable routing, broadcasting and multicasting ad-hoc military networks, energy-efficiency, quality of service, queueing theory, autoconfiguration, security issues. His current research interests include modern Wireless Sensor Networks, Internet of Things, network coding, Information Centric Networking (ICN), caching, …

Cedric follows and participates continuously to standardization efforts, with contributions to within 802.11 task group "s", OLSR protocol (RFC 3626), IEEE 802.14 (cable), network coding research group at IRTF, …

He is co-managing the IoT-LAB site located at Rocquencourt (with Emmanuel Baccelli).

Thomas Watteyne

Thomas Watteyne (http://eecs.berkeley.edu/~watteyne/) is an insatiable enthusiast of low-power wireless mesh technologies. He is a Senior Networking Design Engineer at Linear Technology, in the **Dust Networks** product group, the undisputed leader in supplying low power wireless mesh networks for demanding industrial process automation applications. He designs networking solutions based on a variety of Internet-of-Things (IoT) standards. Since 2013, he co-chairs the **IETF** 6TiSCH working group, which standardizes how to use IEEE802.15.4e TSCH in IPv6-enabled mesh networks, and recently joined the IETF Internet-of-Things Directorate. Prior to that, Thomas was a postdoctoral research lead in Prof. Kristofer Pister's team at the University of California, **Berkeley**. He founded and is the coordinator of Berkeley's OpenWSN project, an open-source initiative to promote the use of fully standards-based protocol stacks for the IoT. Between 2005 and 2008, he was a research engineer at France Telecom, Orange Labs. He holds a PhD in Computer Science (2008), an MSc in Networking (2005) and an MEng in Telecommunications (2005) from INSA Lyon, France. He is fluent in 4 languages.