openstack summit
devs users

An Edge computing case study for Monasca: Smart City (AI-powered) Surveillance and Monitoring

Giovanni Merlino

University of Messina

VANCOUVER (CANADA)

May 23, 2018

- PostDoc member of the MDSLab research group (mdslab.unime.it)
- Member of the ICT staff @ UniME (www.unime.it/it/centri/ciam)
- Involved in the #SmartME crowdfunding initiative to turn Messina into a Smart city (smartme.unime.it)
- Co-founder of SmartMe.io (smartme.io)

R&D in ICT

Faculty staff (involved in S4T):

Prof. Puliafito (head)
Prof. Bruneo
Prof. Distefano
Prof. Longo

Research activities:

- Cloud Computing
- IoT and Sensor Networks
- Modeling and Performance evaluation
- Software engineering

Services:

- Prototypes development
- Cloud-based services
- Complex computations
- Dependability analysis

Academic (UniMe) spin-off company,
focusing on Smart city solutions,
«on a **shoestring** budget»

Small but **dedicated** team

CINI – Smart Cities Lab

National Inter-university Consortium for Informatics
The Consortium involves 1,300+ professors of both Computer Science (Italian SSD INF/01) and Computer Engineering (Italian SSD ING-INF/05), belonging to 39 public universities.

Italian competence center in the field of Information and Communication Technologies (ICTs) for Smart Cities.
Its main objective is the development of innovative solutions for improving the citizens' quality of life.

Outline

- Motivation
- Cloud and IoT integration
- Enabling technologies
- Stack4Things architecture
- Smart City pilot and case study (+ demo)
- Functionalities and service levels (+ demo)
- Application domains
- Future work

Motivation

- How to manage in a scalable and powerful way the proliferation of (increasingly smarter) mobile and IoT devices?

http://www.ti.com/lsds/media/images/wireless_connectivity/50BillionThings.png

IoT ecosystem:

- Mobiles
- Cyber Physical Systems
- Smart appliances
- Sensors/Actuators
- Wearables
- Vehicles
- …

IoT devices

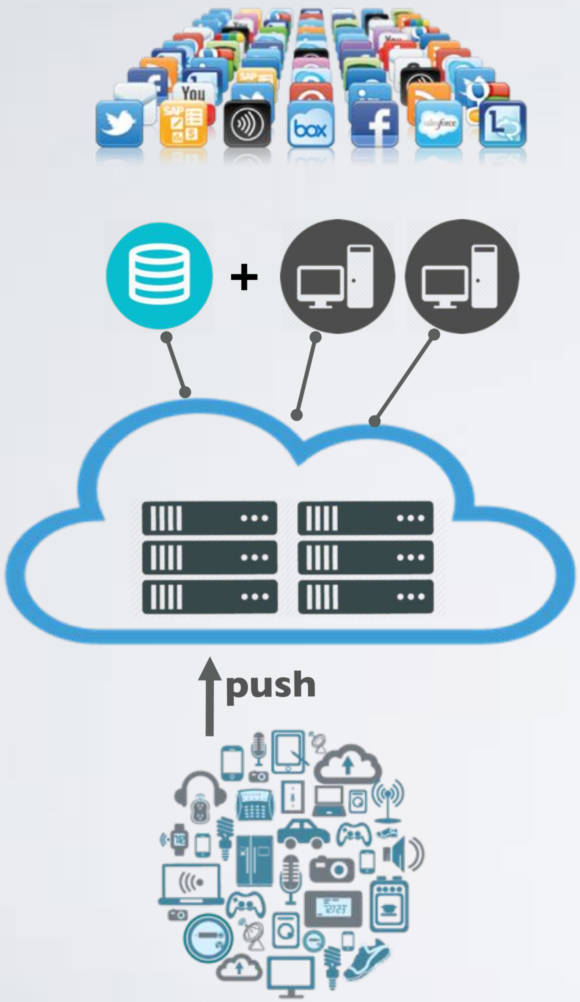
- **Microcontroller boards** or **single board computers** with sensors/actuators attached to (analog/digital) gpio pins or serial bus
- a wide range of interfaces

- **Smart objects** providing interactions with physical world
- wi-fi/bluetooth connectivity

- **Smartphones** with sensors on-board
- wi-fi/bluetooth/3-4G connectivity

Cloud and IoT integration

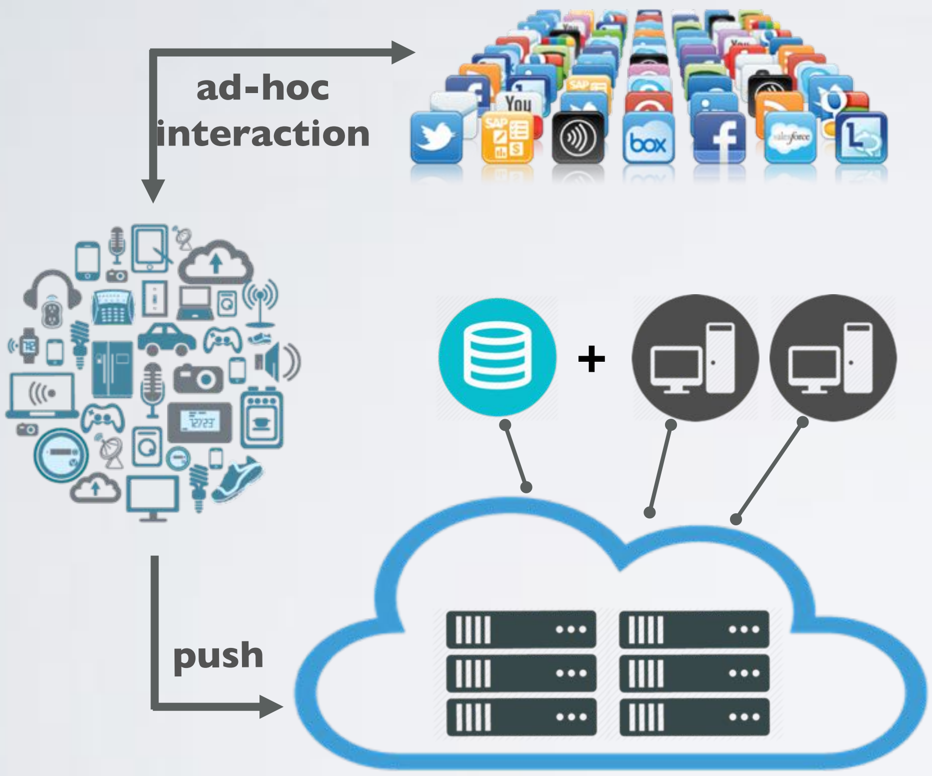
Data-oriented approach



- IoT devices send data to the Cloud
- The application is built on top of **standard cloud facilities** (VMs, storage, network)
- The application makes use of stored (**non-real time**) IoT data
- Indirect, IoT **device-initiated** only, retrieval of actuation commands

Cloud and IoT integration (cont'd.)

Application-specific (vertical) approach



- The application uses **ad-hoc mechanisms** to interact with IoT devices
- **No explicit interactions** between Cloud components and IoT infrastructure
- **Static** infrastructure deployment

Cloud and IoT integration (cont'd.)

Full IoT cloudification (the I/Ocloud)

- We consider the IoT infrastructure as a **natural extension** of a datacenter
- Well-defined Cloud API as a resource management interface
- **Separation of concerns** between infrastructure and application (when needed)
- From Cloud to **Fog/Edge** computing
- Device computation **offloading**

Main issues

http://www.connectedcoast.org

- **Difference between classic IaaS Cloud and our IoT-extended Cloud**
  - IoT devices are resource constrained
  - IoT devices are out of the datacenter
  - IoT devices are at the edge of the Internet and could be behind NATs/firewalls
- IoT devices are unreliable (w.r.t. networking and computation)
- IoT nodes do not have out-of-band/lights-out management systems
- IoT devices do not belong to the same administrative domain (there is a difference between Cloud administrator and resource owner)

Stack4Things: underlying technologies

(Standalone) implementation

- Node.js is a JavaScript runtime built on **V8 JavaScript engine**
- It uses an **event-driven, non-blocking I/O model** that makes it lightweight and efficient
- Node.js' package ecosystem, **npm**, is the largest ecosystem of open source libraries in the world.
- Node.js has become **ubiquitous** in almost every technology niche, and especially so in IoT
  - Node-RED (visual tool for wiring IoT APIs)
  - Cylon.js (framework for cross-platform robotics and physical computing)

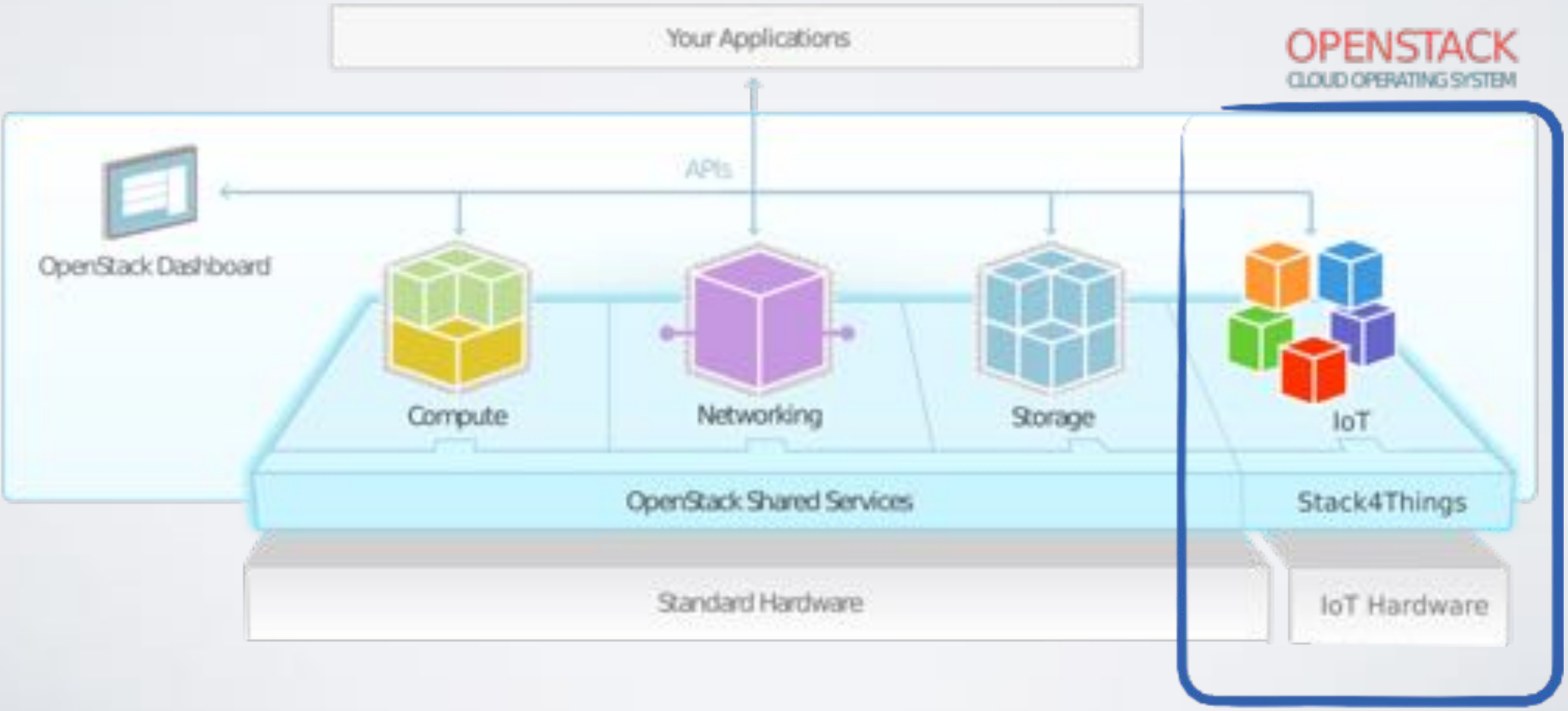
OpenStack-based implementation

- IoT resource management service for OpenStack clouds

- OpenStack (unofficial) project

https://launchpad.net/iotronic

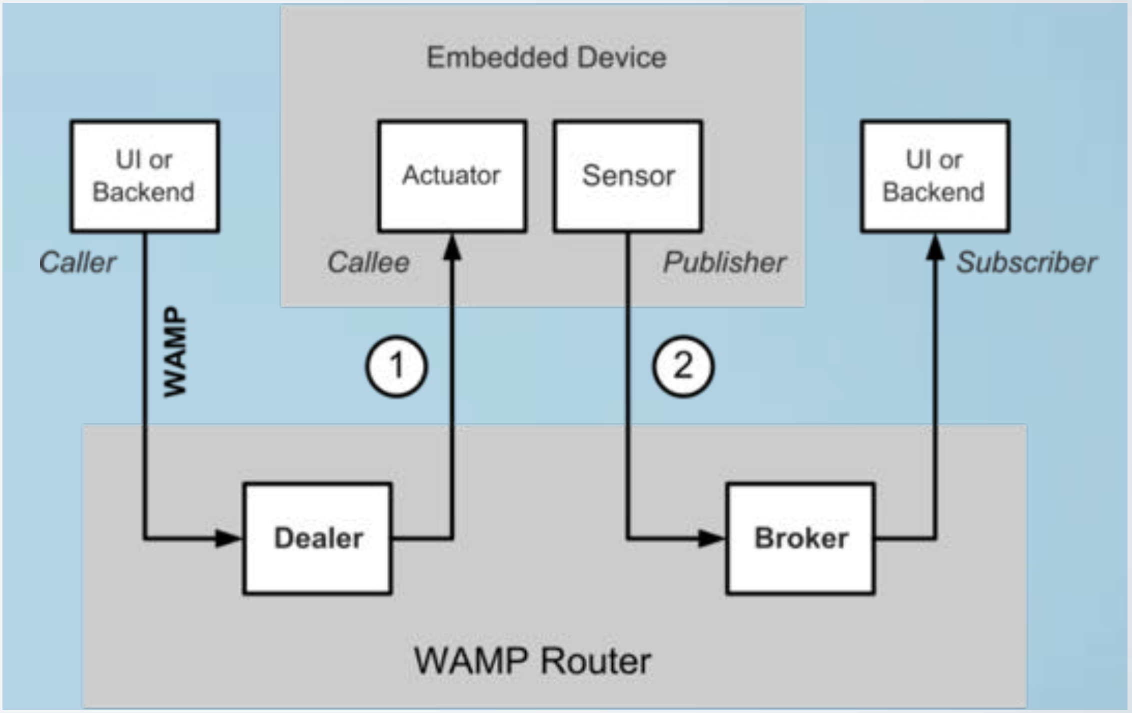
https://git.openstack.org/cgit/openstack/iotronic/



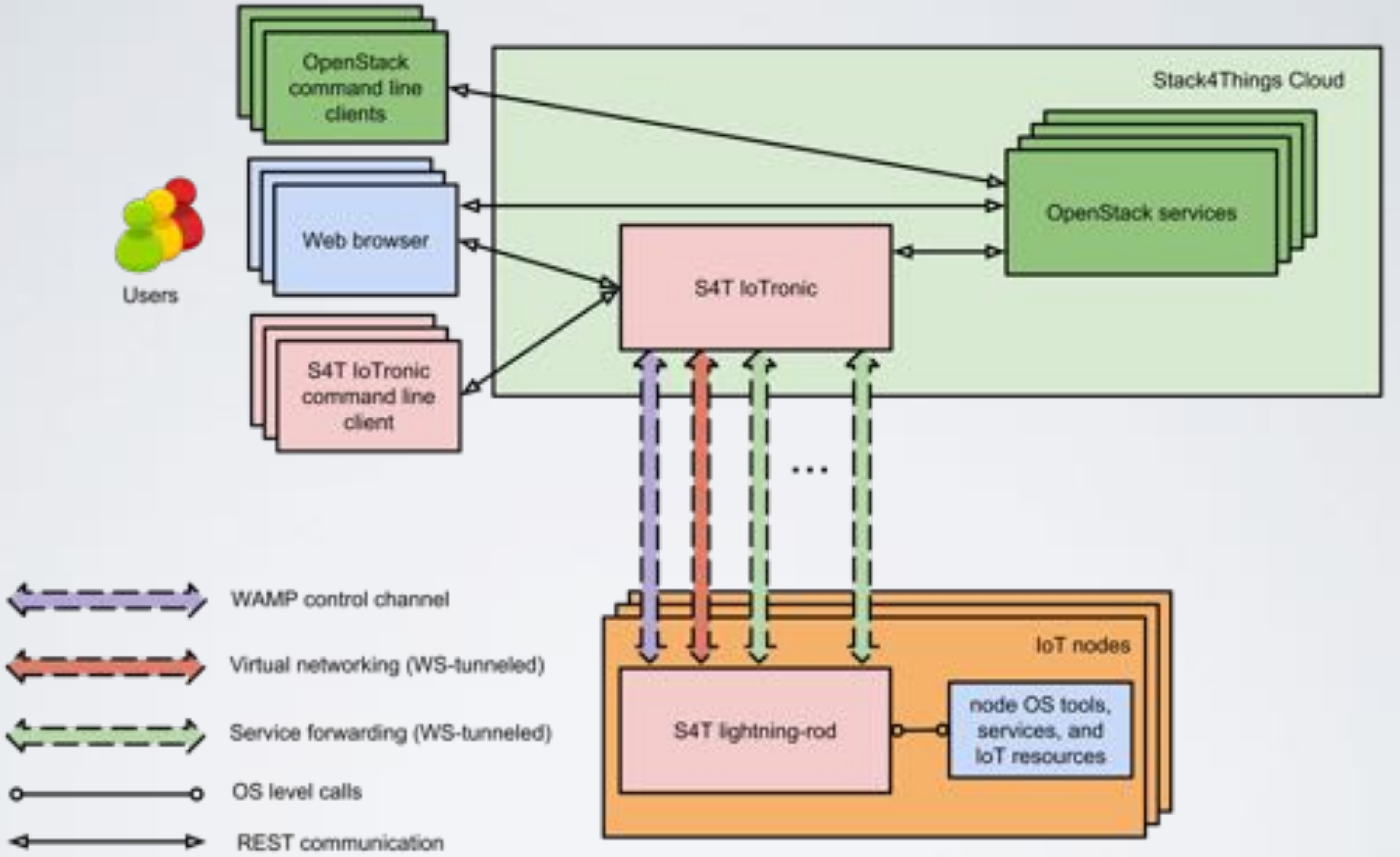
WAMP protocol

- WAMP (http://wamp-proto.org) is an open standard WebSocket subprotocol

- Two application messaging patterns in one unified protocol
  - **Publish & Subscribe**
  - **RPC**



High-level overview

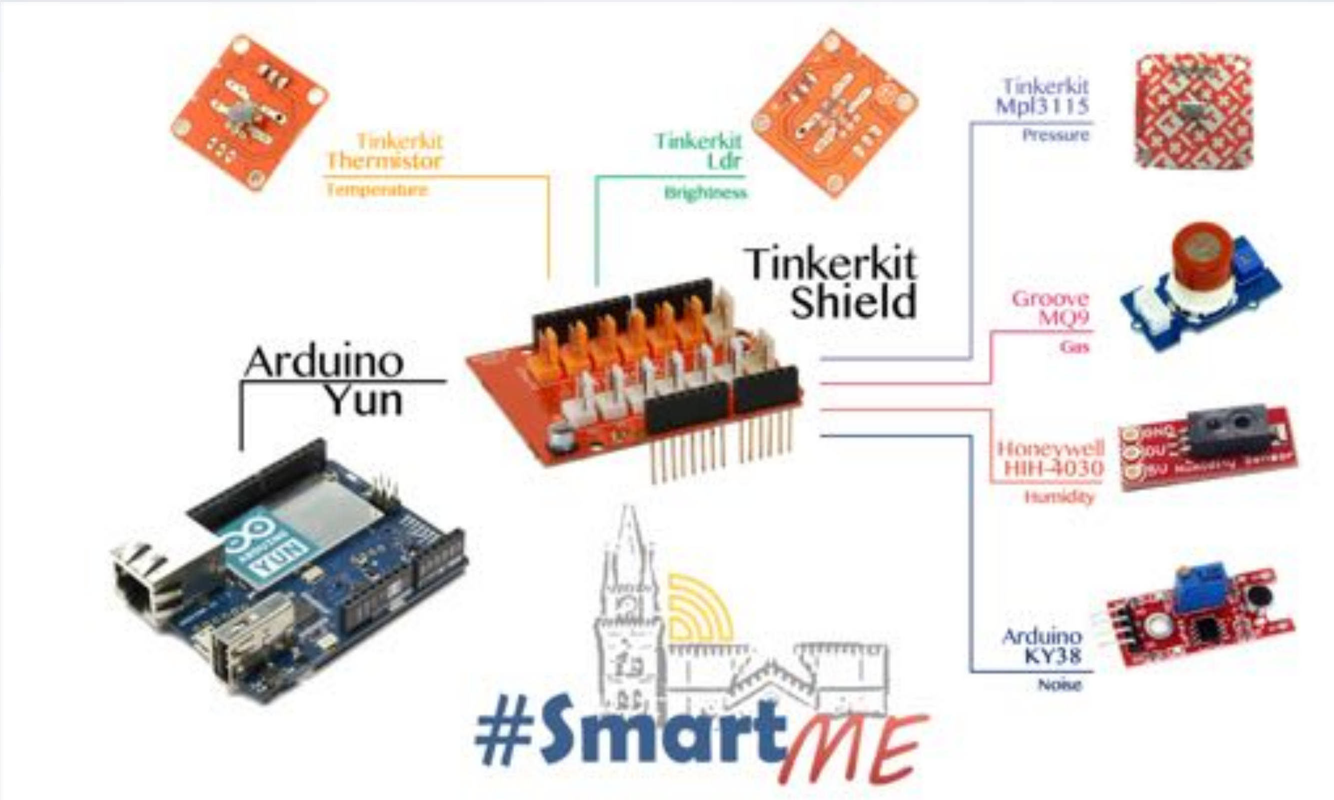


- Use of a software **probe** on the device-side (lightning-rod)
- OpenStack-compliant service (IoTronic)
- Use of WAMP and plain WebSocket control channels
- REST interfaces

The #SmartME project

- **collaboration** of ***MDSLab*** team with **key** actors
  - ***Arduino*** Labs, municipality, university branches
- successful **crowdfunding** initiative
- a **platform** for experimental **testbeds**

Example of #SmartME node



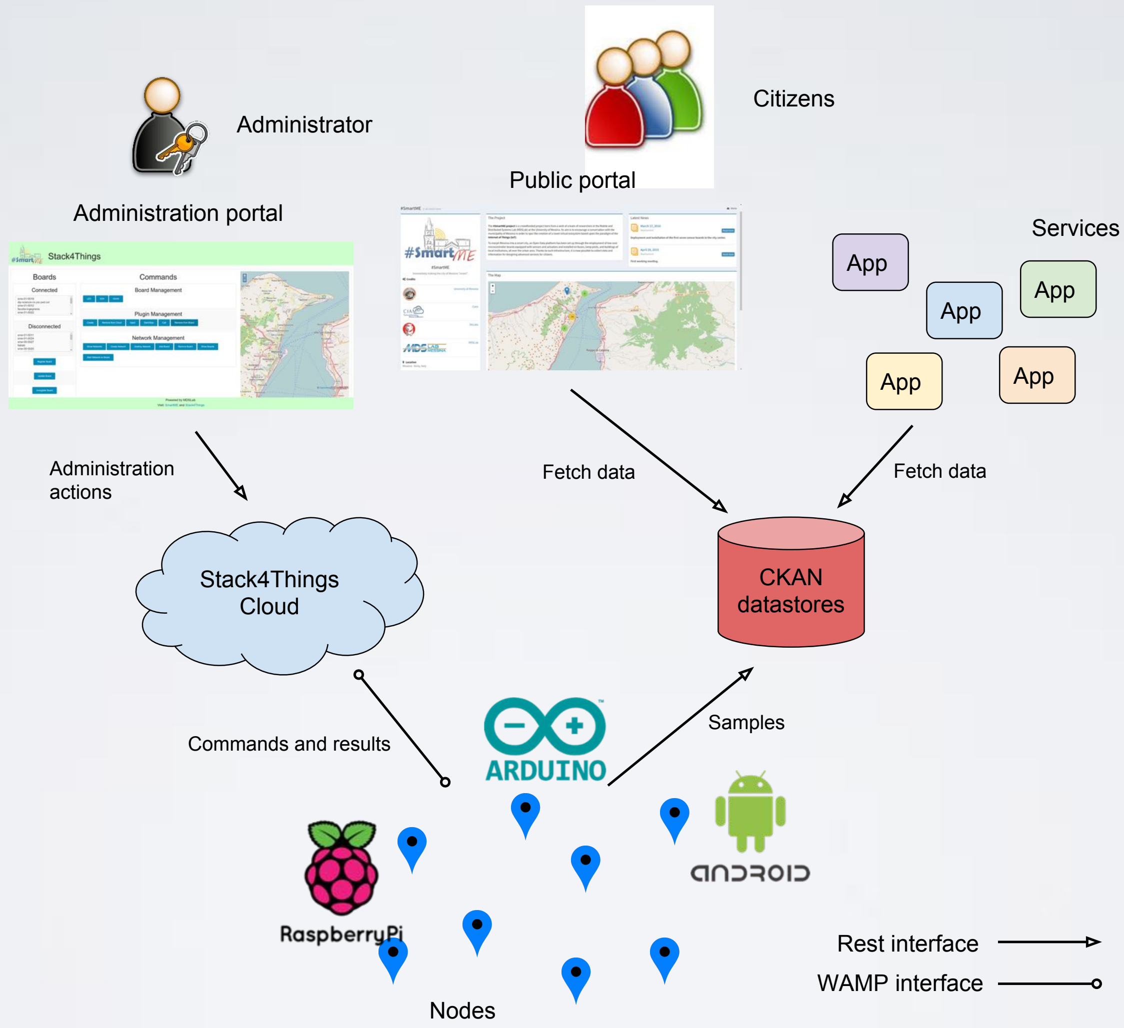
Current implementations

ARDUINO

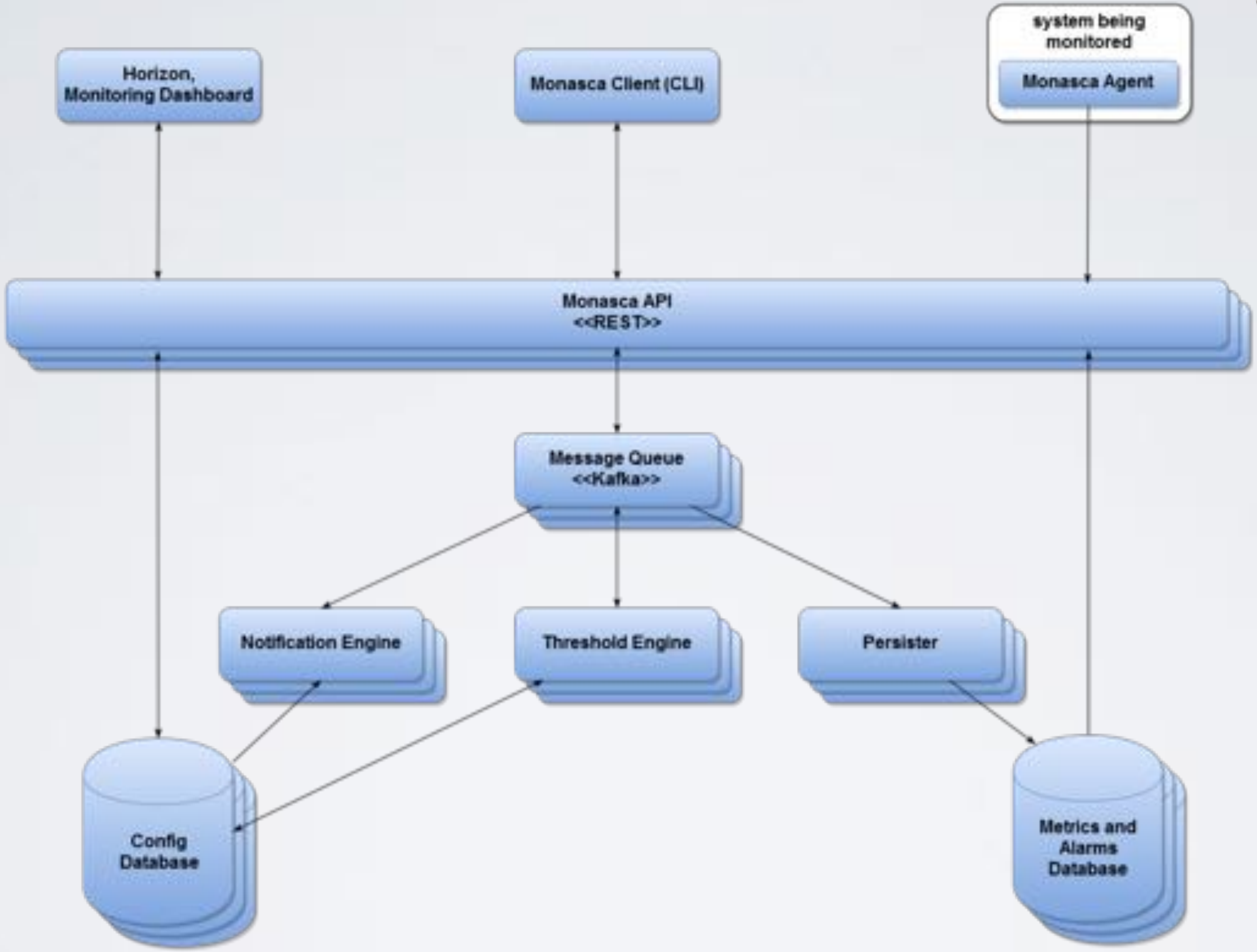
RaspberryPi

ANDROID

Architecture of the #SmartME framework



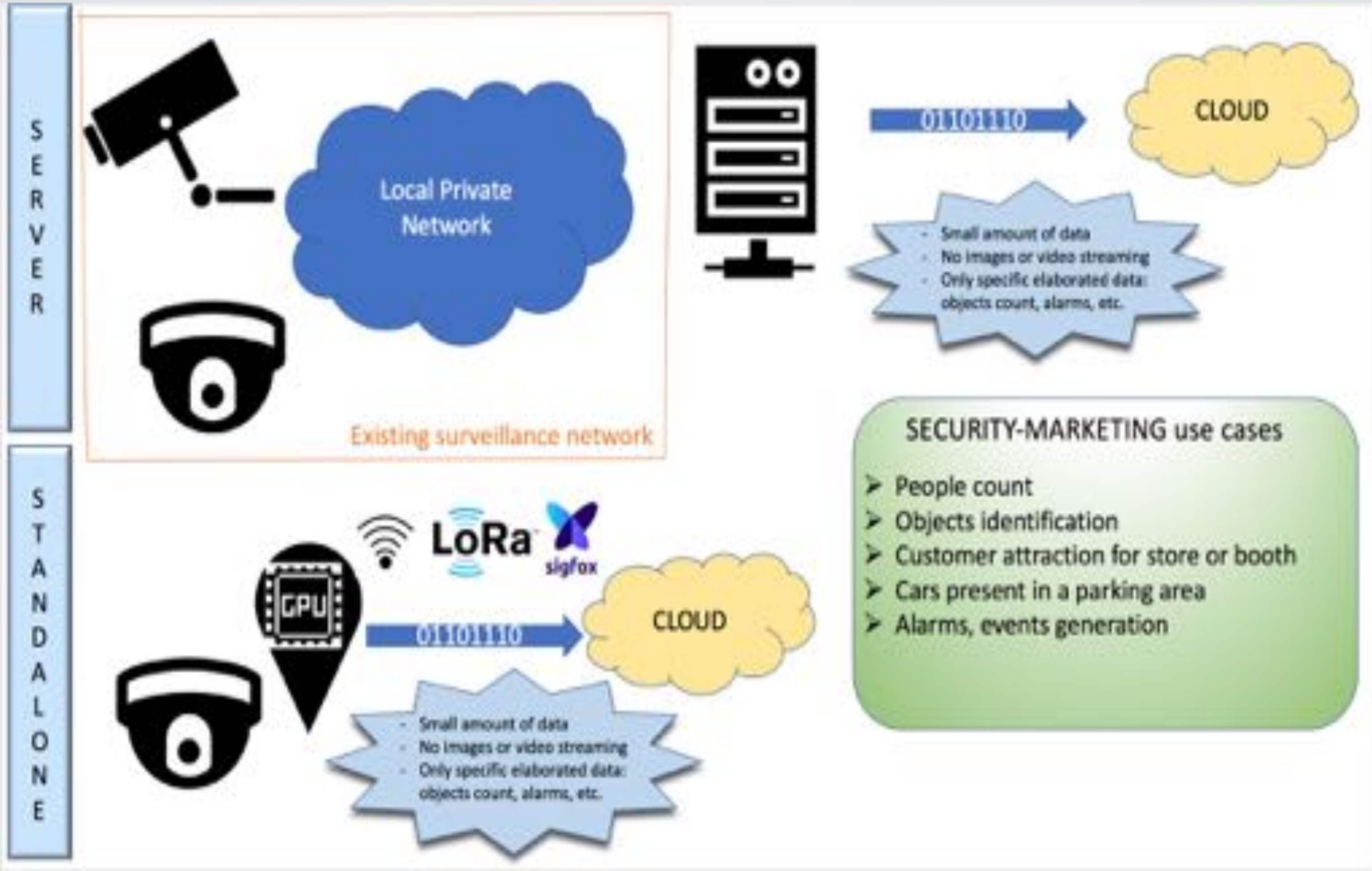
Smart City dataplane: Monasca architecture



Some more details on the dataplane

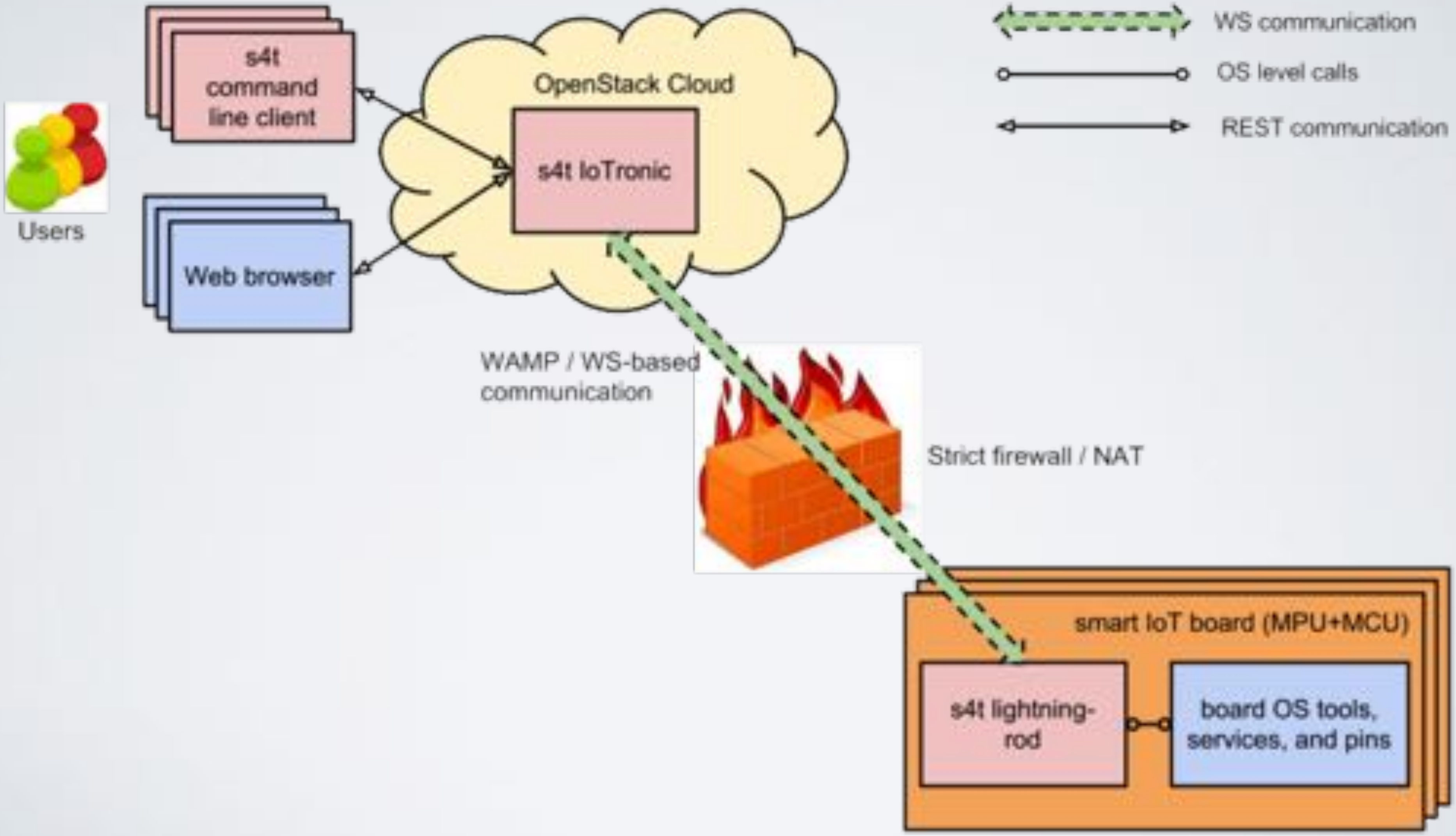
- Please refer to our previous presentation (video included) at the OpenStack Summit 2017 in Boston:
  - https://www.openstack.org/summit/boston-2017/summit-schedule/events/17951/a-monitoring-case-study-for-monasca-smart-city-infrastructure

#SmartMECam: surveillance



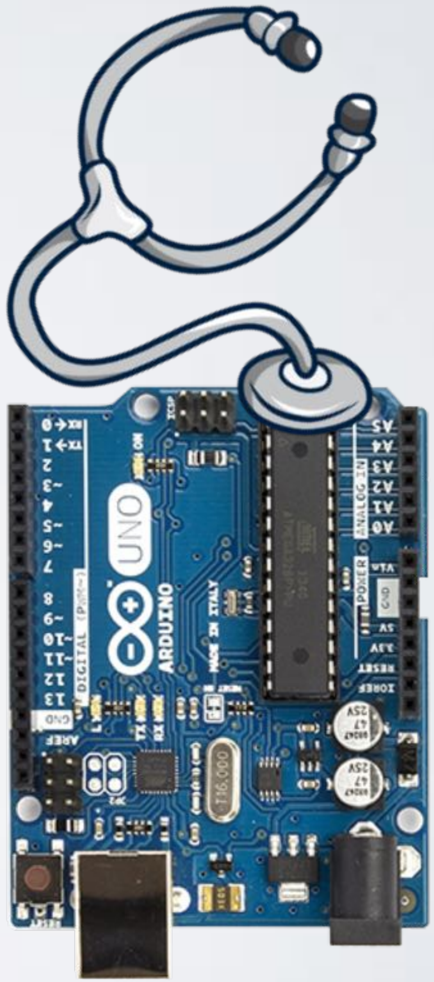
Demo time

Tipical deployment scenario



Device-side enablement: requirements

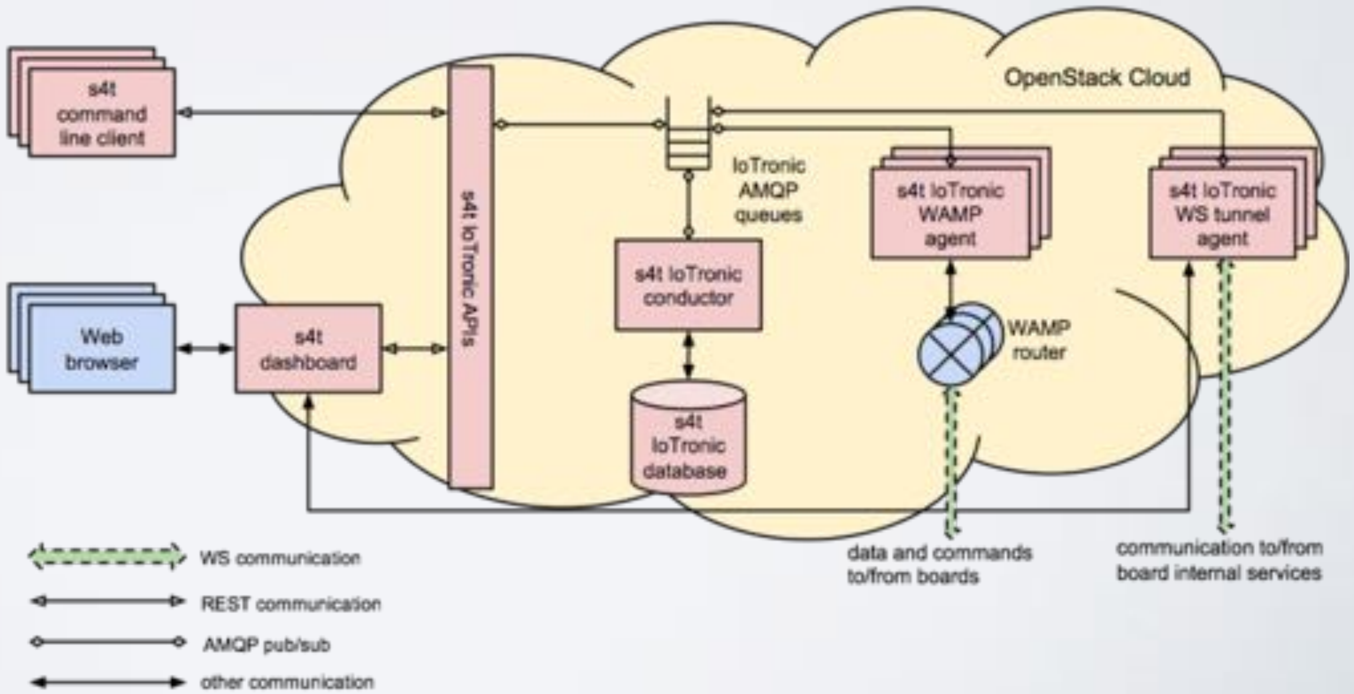
- How to deal with the lack of out-of-band/lights-out management hardware?
  - The *lightning-rod* process acts as the **software counterpart** of such management systems
  - Always running (watchdog-like behavior)
  - Always **remotely accessible** at the lowest level (e.g., console-like)
  - **Sandboxed** where possible



Cloud side

- Infrastructure management and interaction services exposed as **RESTful APIs**

- The **Horizon dashboard** as control surface for any kind of resource, including IoT-borne ones
- Deep integration with OpenStack (OS) frameworks and services, i.e., Cloud-side functionalities:
  - modeled after OS interactions
  - implemented as OS services
  - leveraging OS facilities



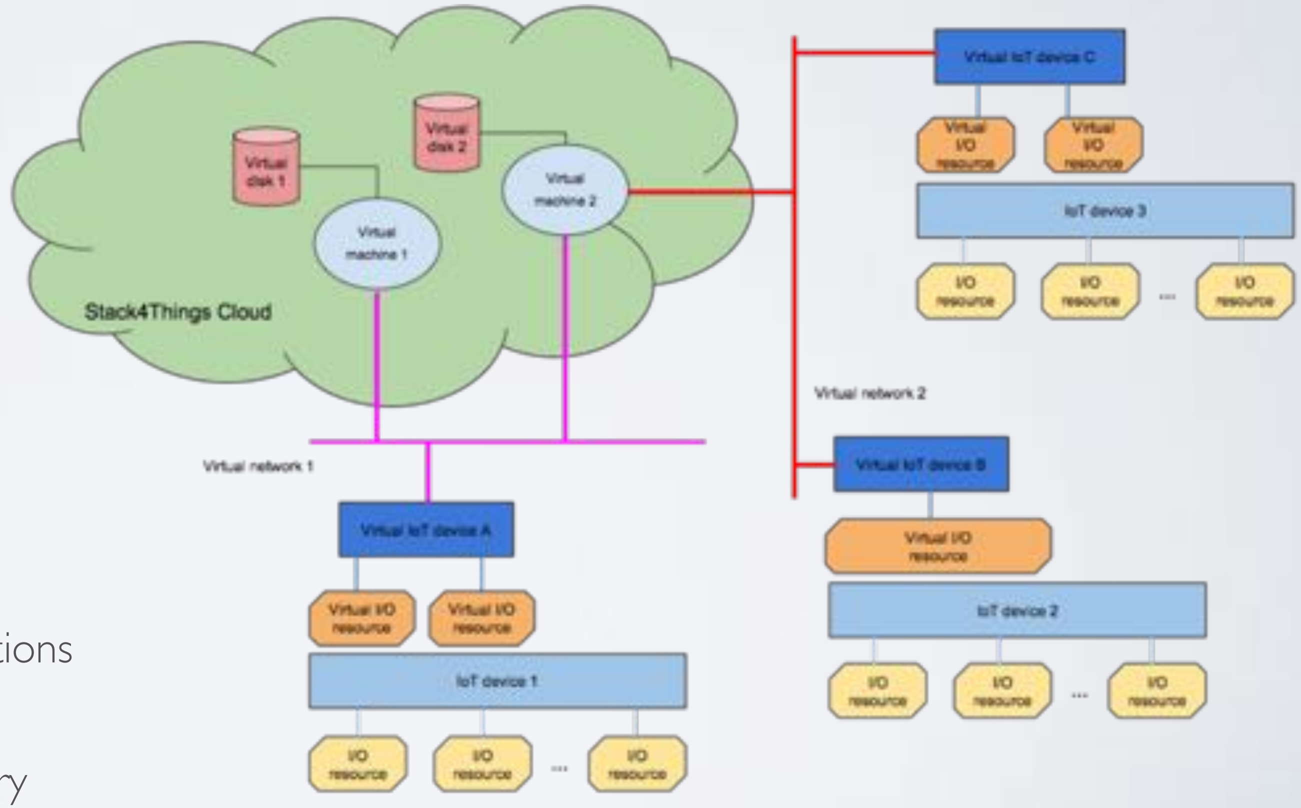
Networking

- OpenStack **Neutron** is exploited to create virtual networks

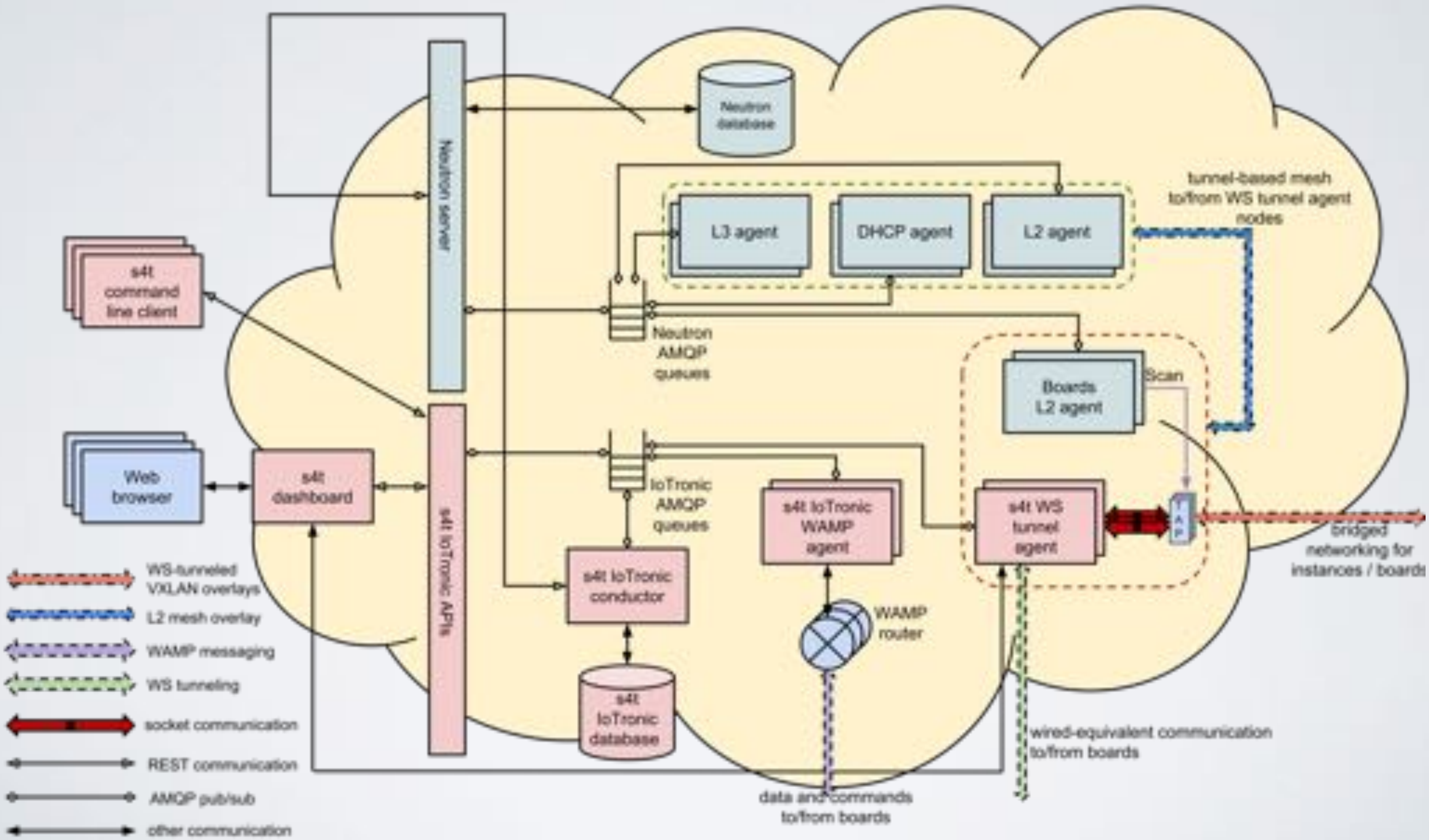
- On the device side, WebSocket and Linux low-level tools are used to create **virtual interfaces**

- Virtualization both at the **network and datalink layers** enables flexible overlay networking topologies
  - infrastructure-agnostic applications
  - also extending the scope of applicability of service discovery protocols (e.g., AllJoyn)

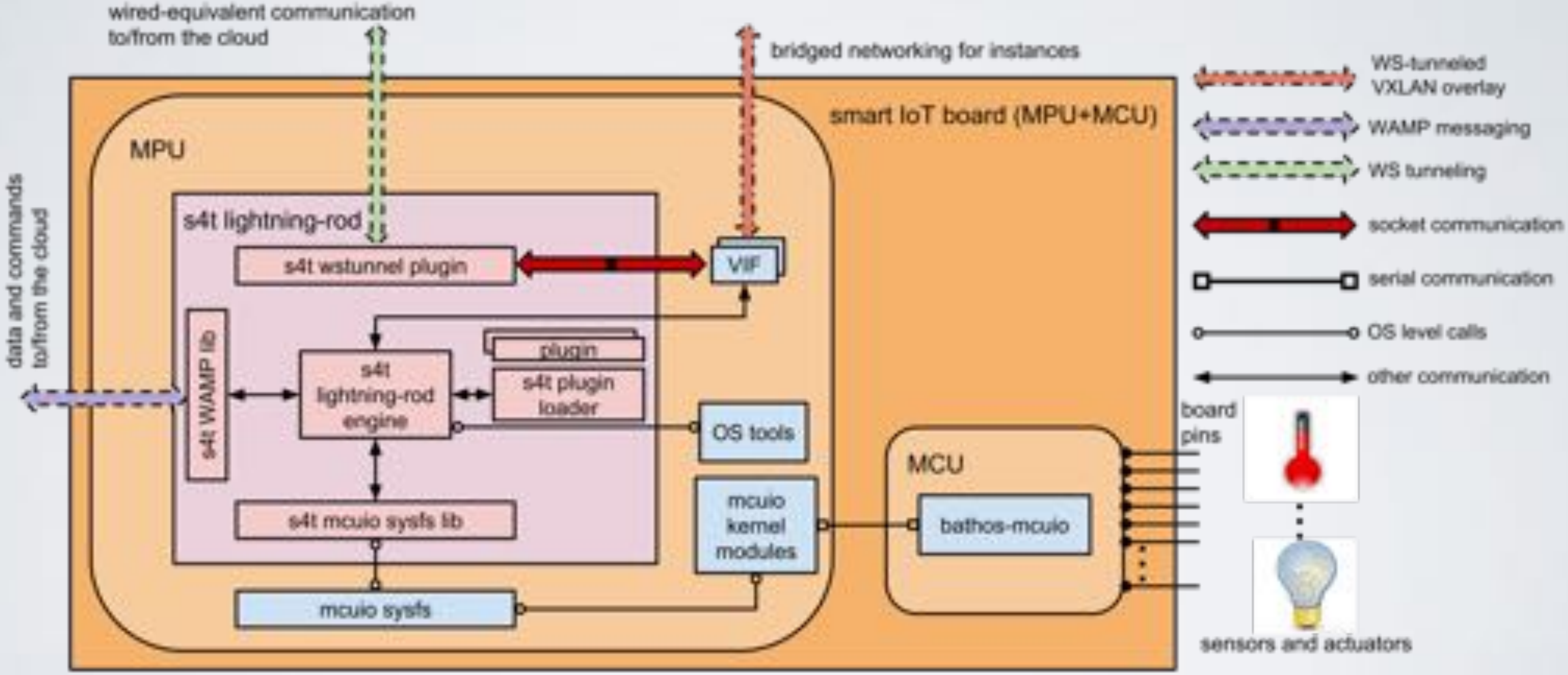
Virtual networks may span both (datacenter-hosted) VMs and virtual IoT devices



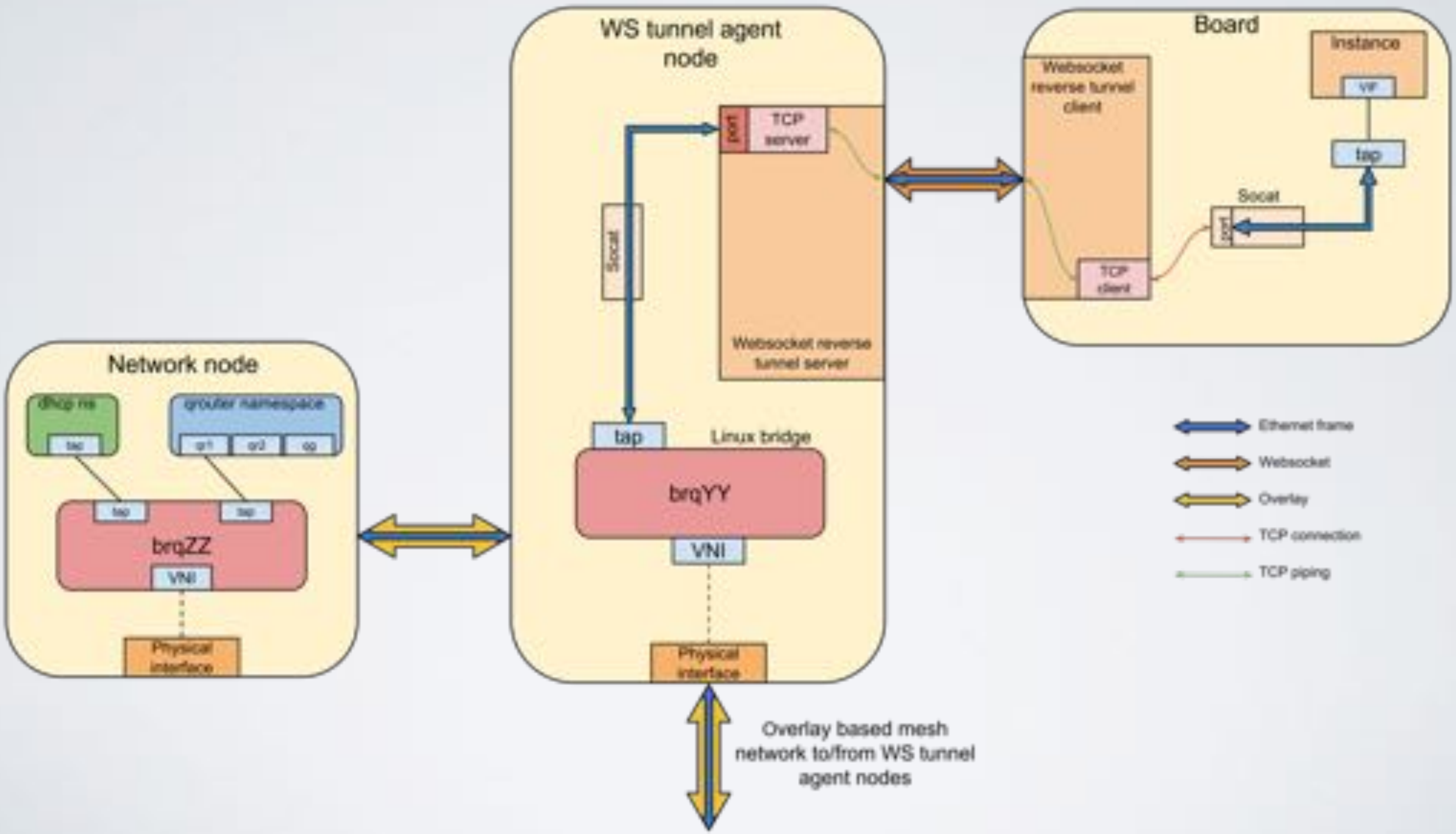
Neutron integration: Cloud-side architecture



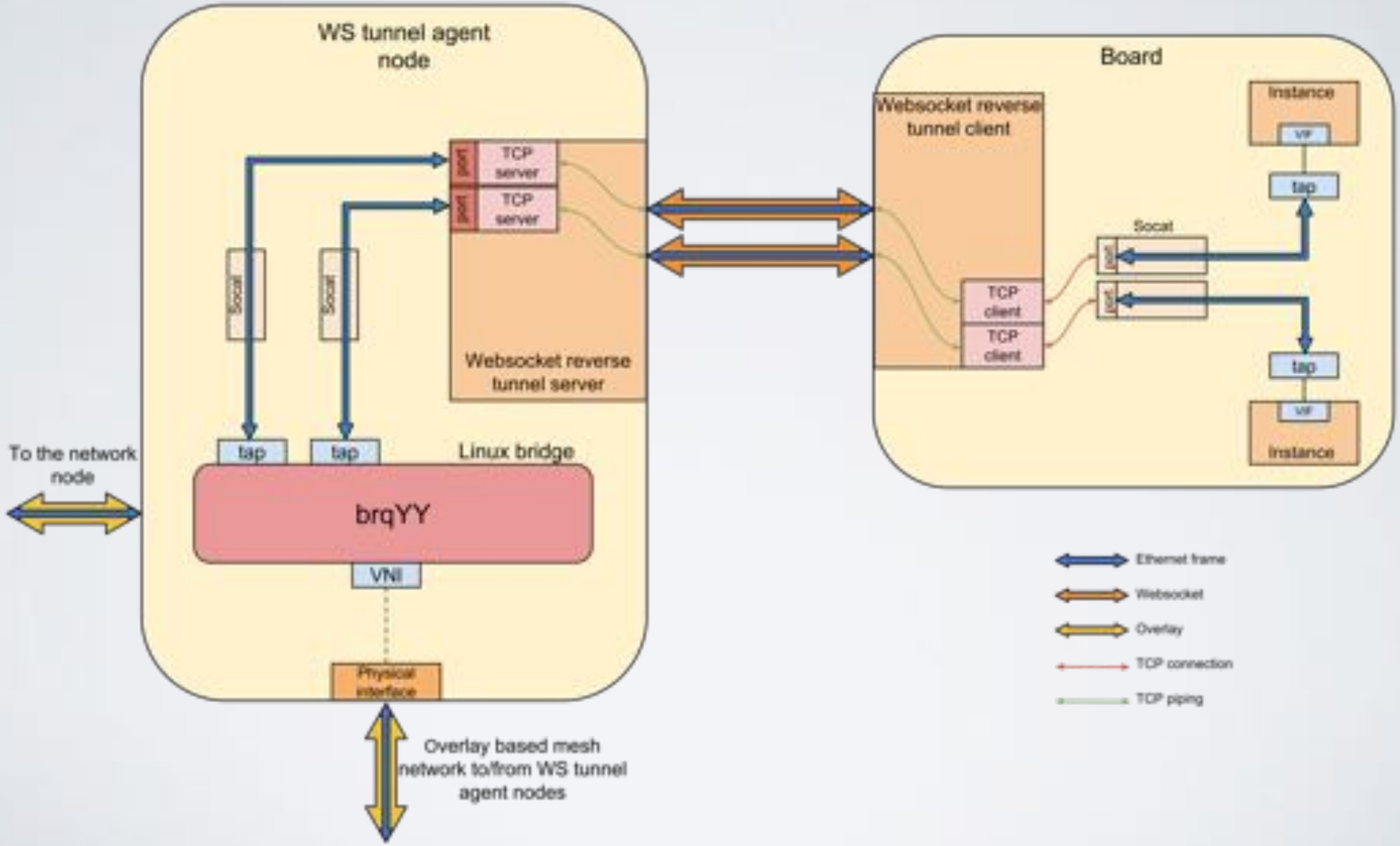
Neutron integration: node-side architecture



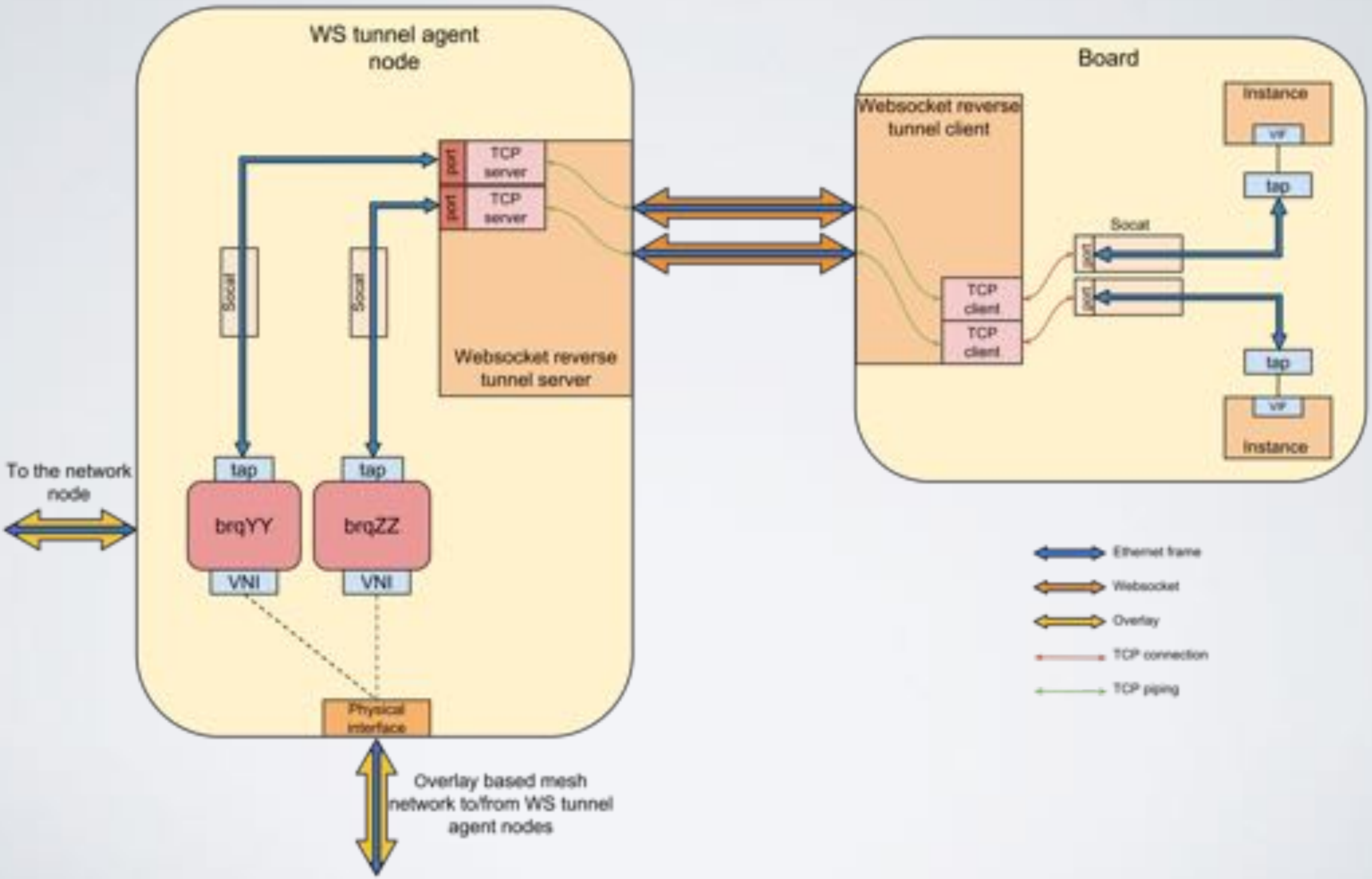
Topology and datapath



Same-board instances, same overlay



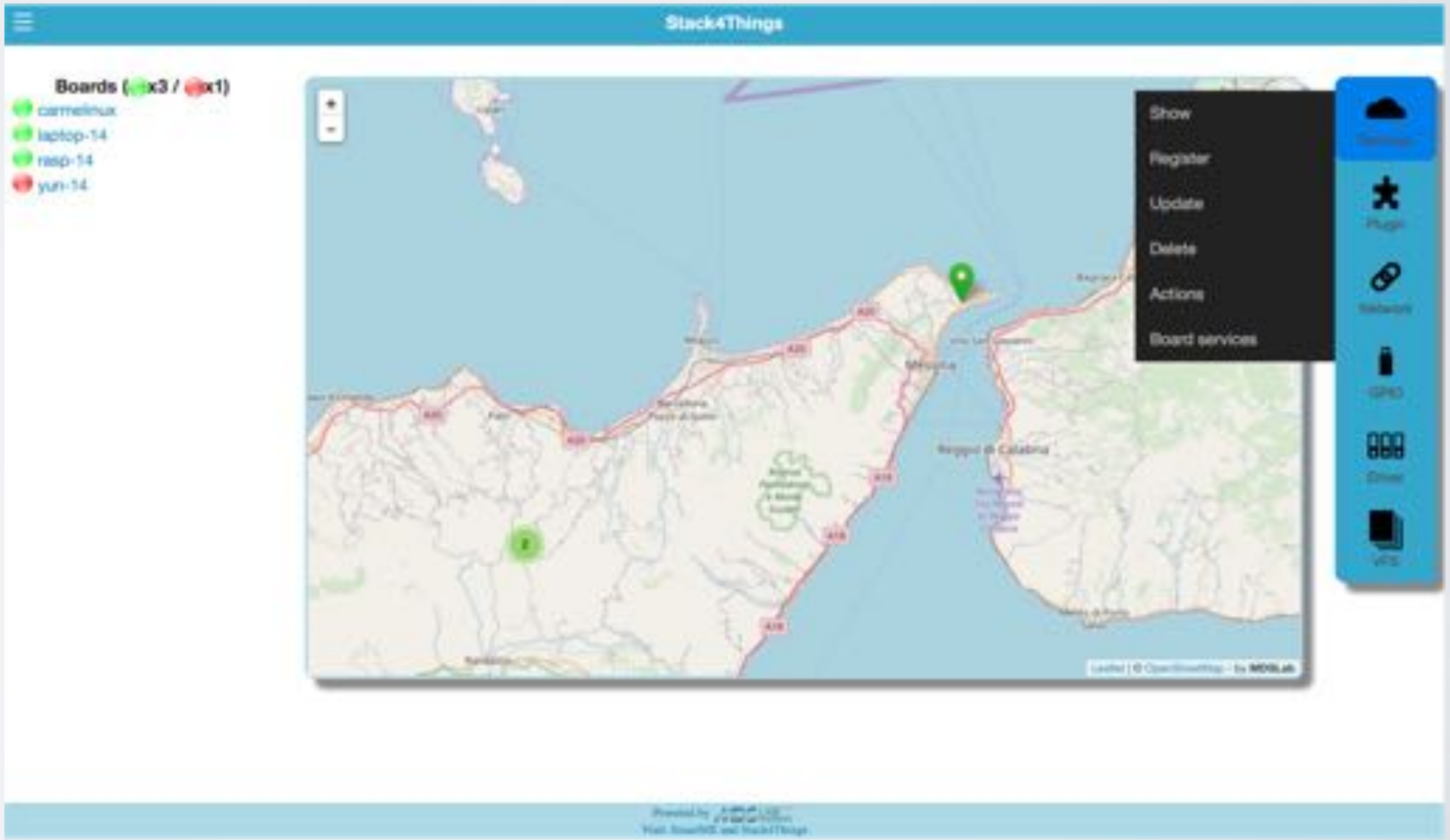
Same-board instances, separate overlays



PaaS

- **Plugin-based development**:
  - plugins as minimal, high-level, self-contained code units (design-time advantage)
  - plugins as isolated (sandboxed) processes (run-time advantage)

- Enabling mechanisms:
  - plugin wrappers for runtime execution
  - plugin **registration** on the Cloud
  - plugin **injection** (deployment)



Demo time (again)

Application domains

- **(interacting) Cyber-Physical Systems:**
  - smart Home/smart Building
  - smart City
  - smart Industry
  - smart Mobility
- **Smart communities:**
  - makers
  - meteorologists
  - safety and law enforcement officers
  - (the list grows day by day)
- **Fleet management:**
  - software upgrades for fleets of products
  - security and surveillance
  - vehicles position tracking
  - (the list grows day by day)

Future work

- **Spin some built-in functions off IoTronic (and into other, more apt subsystems):**
  - breaking it down to its core
  - specializing it for IoT-unique features (not applicable or uninteresting to other communities in the infrastructure/platform space)
- **Further integration/interaction with OpenStack and its communities:**
  - Placement (traits specified according to available physical resources, see vGPU)
  - Mogan (with IoTronic as a bare metal provisioning driver for IoT resources)
  - Keystone (authentication, authorisation, delegation mechanisms dealing with the IoT different administrative domains and ownership models)

Thanks for your interest!

Giovanni Merlino gmerlino@unime.it
giovanni@smartme.io

Project(s) / Company links:
http://stack4things.unime.it
http://smartme.unime.it
http://smartme.io

A shout to our devs / collaborators,
for making the S4T demo possible:

Carmelo Romeo
Fabio Verboso
Nicola Peditto
Zakaria Benomar