Head in a Cloud: an Approach for Arduino YUN Virtualization

D. Bruneo, S. Distefano, F. Longo, G. Merlino, A. Puliafito, A. Zaia
University of Messina
Kazan Federal University
Arduino
Outline

• Intro
  • device-centric paradigm

• Technology enablers
  • Arduino
  • OpenStack

• Virtualization approach
  • design

• Proof-of-Concept

• Experiment
  • results

• Future work
Cloud and IoT integration: data-oriented

- IoT devices **send** data to the Cloud
- apps built on top of **standard** cloud facilities (VMs, storage, network)
- apps make use of stored (**non-real time**) IoT data
- indirect, IoT **device-initiated** only, retrieval of actuation commands
Cloud and IoT integration: application-specific

- apps use **ad-hoc mechanisms** to interact with IoT devices
- **no explicit interactions** between Cloud components and IoT infrastructure
- **static** infrastructure deployment
Cloud and IoT integration: full thing “cloudification”

- 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**
Device-centric paradigm: I/Ocloud

Approaches (to I/O extensions for the Cloud):

- Bare-metal (VMs with I/O)
- Virtualization (VNs)
OpenStack extensions for IoT: Stack4Things
Stack4Things architecture
PoC experiment: scenario

Virtual YUN can migrate from edge to cloud and vice versa: application logic does not need to change

Arduino YUN

Virtual Arduino YUN (LXD container)

REST server

virtual MCUIO sysfs

temperature sensor

Internet

Client

Firewall need to be manually set up for user to be able to connect to the REST server

Firewall is automatically setup by S4T on the Cloud side

Virtual Arduino YUN (LXD container)

REST server

virtual MCUIO sysfs

Cloud Compute node
Experimental results

![Graph showing experimental results comparing Virtual YUN on the Cloud and Virtual YUN on physical YUN.

- **Y-axis**: Average response time (msec)
- **X-axis**: Computational load (# of FFT performed)

- **Red line**: Virtual YUN on the Cloud
- **Green line**: Virtual YUN on physical YUN

The graph illustrates the comparison of response times for different computational loads between the two configurations.
Future work

• manage pin naming/mapping, access control and concurrency
• forward hardware interrupts (poll-like POSIX semantics)
• migrate virtual IoT devices across ISAs
Q&A

Thanks!

gmerlino@unime.it