Provisioning Software-defined IoT Cloud Systems

Stefan Nastic

Distributed Systems Group, Vienna University of Technology

http://infosys.tuwien.ac.at/research/viecom/SDIoTCloud

Advanced School on Service Oriented Computing

July, 2014, Hersonissos, Crete, Greece

Introduction

Data Center: Processing, Storage, Networking

IoT devices: Gateways, Sensors, Actuators

Introduction

Data Center: Computing, Storage, Networking

IoT devices: Gateways, Sensors, Actuators

- Our aim is to:
  - Abstract IoT Cloud resources in a unified manner.
  - Enable development and operations processes for IoT Cloud applications.

In this talk we focus on provisioning

Why IoT Cloud?

- Fleet-vehicles management
  - World-wide deployment
  - Different types of vehicles (on-board gateways)
  - Different environments
  - Various stakeholders

- Variety of required services
  - vehicle maintenance (fault history,battery health)
  - vehicle tracking (driving history, geo-fencing)
  - vehicle info (charging status, odometer)

Why IoT Cloud?

- Fleet-vehicles management
  - World-wide deployment
  - Different types of vehicles (on-board gateways)
  - Different environments
  - Various stakeholders
- Variety of required services
  - vehicle maintenance (fault history,battery health)
  - vehicle tracking (driving history, geo-fencing)
  - vehicle info (charging status, odometer)
- How to connect all the vehicles?
- How to provide a global view on the fleet?
- Where to store the data?
- How to process the data/alarms in a timely manner?

Core principles

- *From physically isolated, rigid IoT infrastructure to virtualized, elastic IoT Cloud, by utilizing software-defined principles*

Core principles

- *From physically isolated, rigid IoT infrastructure to virtualized, elastic IoT Cloud, by utilizing software-defined principles*

- *From task-specific solutions to fully-fledged IoT Cloud ecosystem and design/management processes, based on DevOps approach*

Traditional Cloud vs. Software-defined IoT Cloud

- IoT Cloud has virtualization at its foundation, but
  - Unlike cloud computing IoT Cloud combines datacenter resources with IoT resources (sensory data, edge devices and actuators).
  - Unlike traditional cloud IoT Cloud has geographically distributed infrastructure and it is much larger in scale.

- IoT Cloud enables better resource utilization, but
  - New models are required.
  - Also some traditional models can be applied.

IoT Cloud provisioning challenges

- **Managing configurations** IoT Cloud systems:
  - Geo-distributed systems (on-site presence?)
  - Very large number of devices
  - Heterogeneity of devices and platforms

IoT Cloud provisioning challenges

- **Managing configurations** IoT Cloud systems:
  - Geo-distributed systems (on-site presence?)
  - Very large number of devices
  - Heterogeneity of devices and platforms

- Expressing **complex relationships** and delivering high-level functionality:
  - Diversity of stakeholders and requirements
  - Plethora of available functions (e.g., communication protocols)

Main aspects

- **Software-defined IoT Units** – conceptual model to abstract IoT Cloud resources and capabilities
  - Main building blocks of Software-defined IoT Cloud Systems

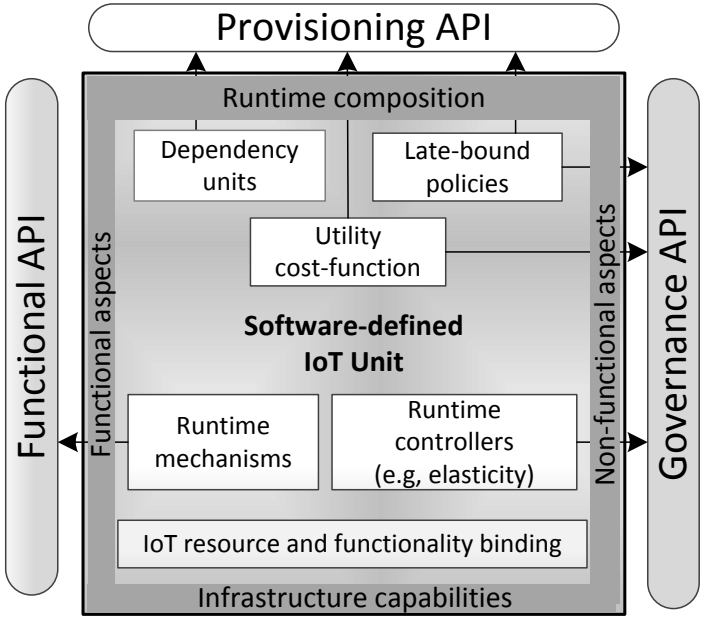
Main aspects

- **Software-defined IoT Units** – conceptual model to abstract IoT Cloud resources and capabilities
  - Main building blocks of Software-defined IoT Cloud Systems

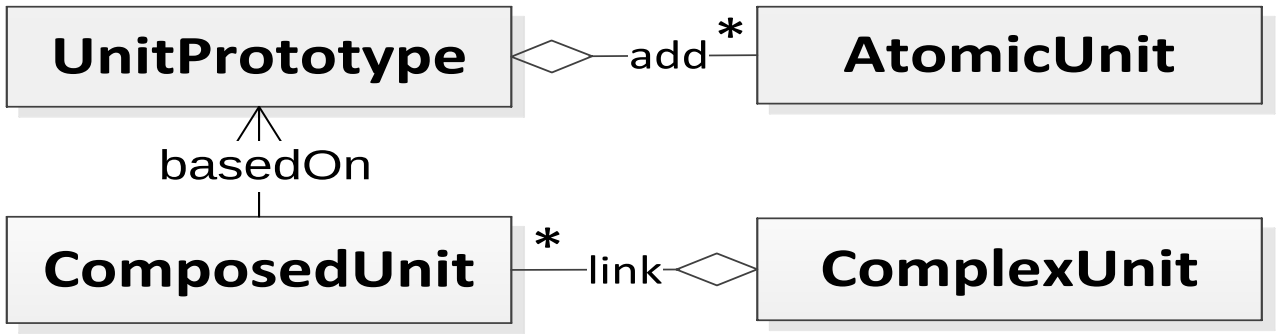
- Main mechanisms to support provisioning of Software-defined IoT Cloud systems:
  - **Managed configuration models**
  - **Automated composition** of IoT Units

Software-defined IoT units

- Provide software-defined API which unifies the view on IoT Cloud resources and capabilities.
- Support fine-grained internal configurations.
- Have utility cost-function that enables pricing the IoT resources as utilities.
- Can be composed at higher levels, in order to provide complex functionality.

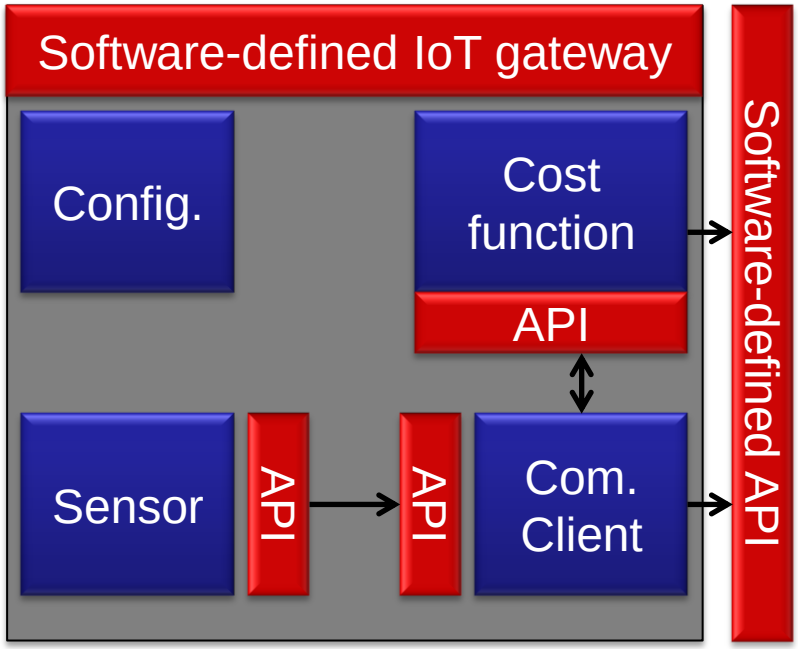


Unit types



- AtomicUnits represent finest-grained IoT Cloud resources and capabilities (e.g., a communication protocol).
- Lower-level units can be interconnected into more complex (higher-level) units.
- UnitPrototypes can be based on VMs, Linux containers, IoC container, OSGi, etc.

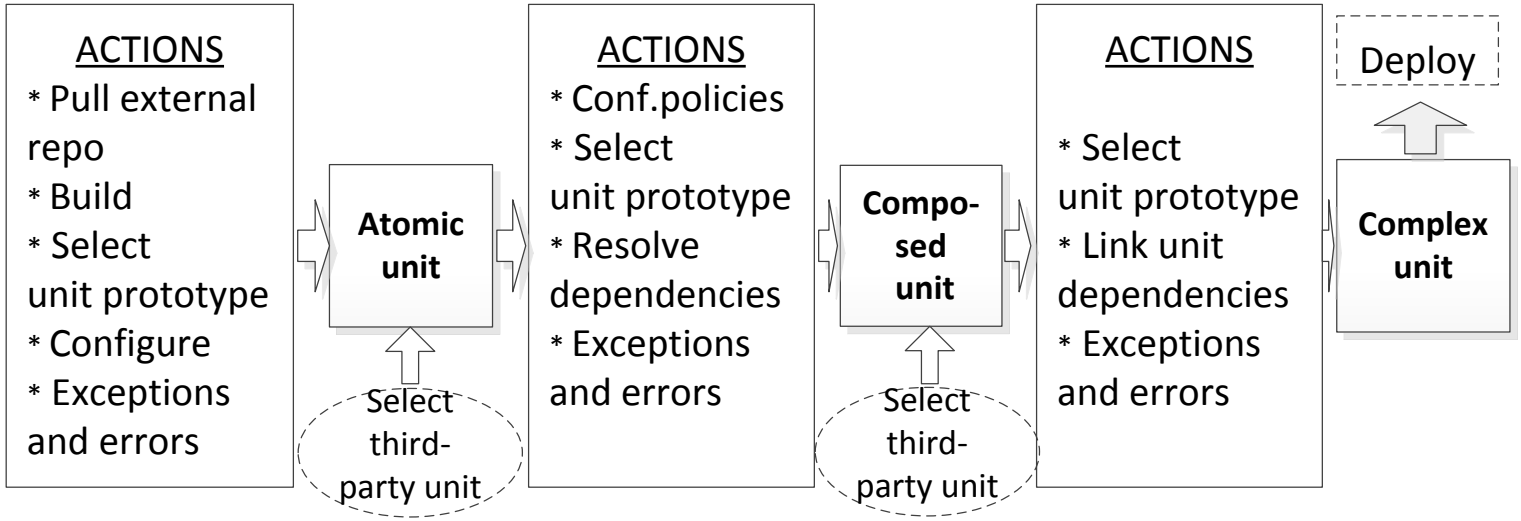
Example of software-defined IoT gateway



Provisioning software-defined IoT Cloud systems

- **Managed configuration models:**
  - We treat functional units and configuration policies in the same manner
  - They are managed centrally and propagated to the edge of the infrastructure automatically.
  - They support late runtime binding
  - Idempotence of configurations
- **Automated composition** and deployment of IoT units:
  - Provisioning single gateways
  - Provisioning complex elasticity-aware, reliable topologies of multiple gateways

Provisioning software-defined IoT Cloud systems



Docker

VM

BOSH

Puppet

Chef

Cloud Formation

OS Heat

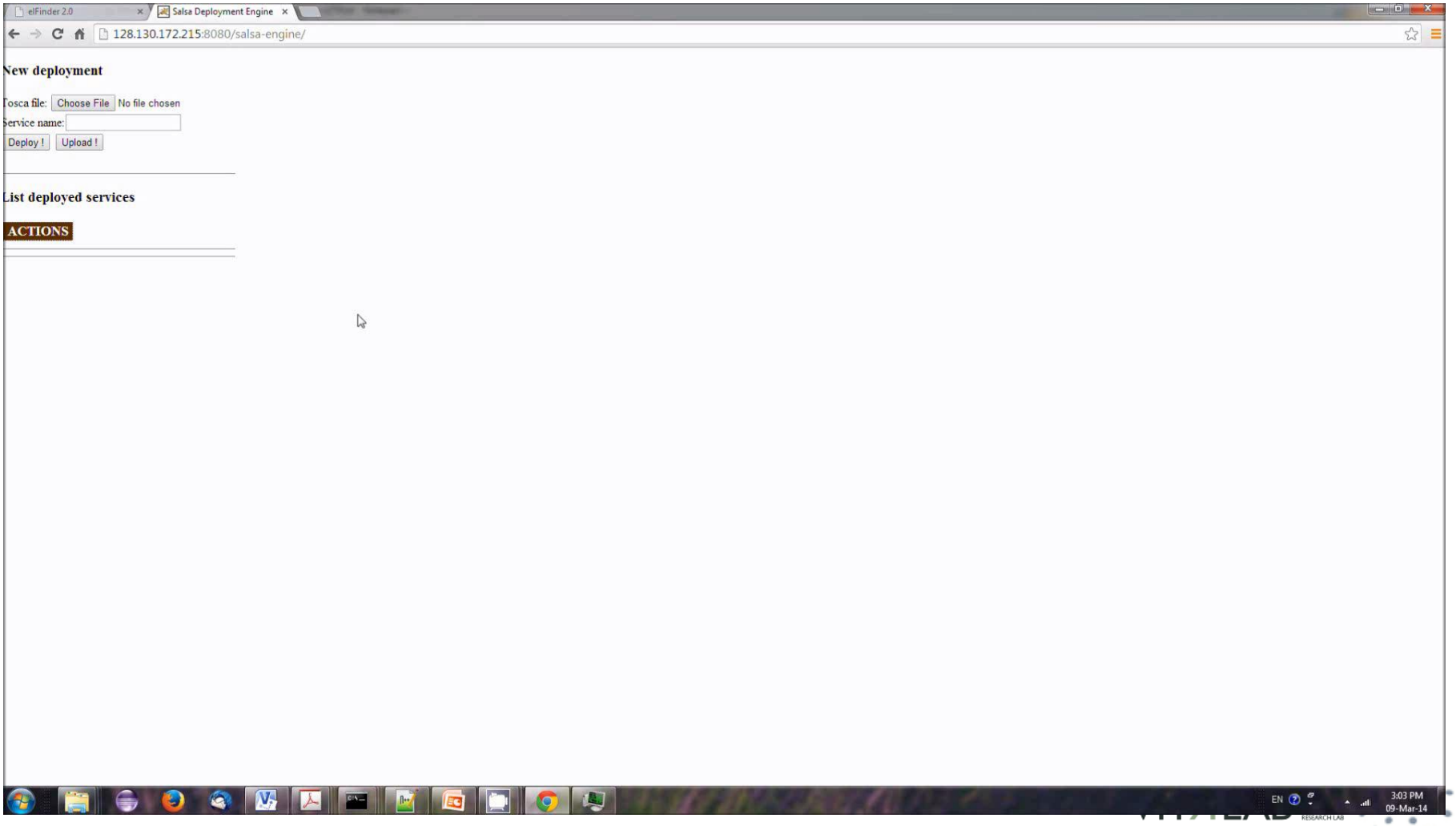
TOSCA

DEMO:

Provisioning a vehicle tracking application with SD IoT Units

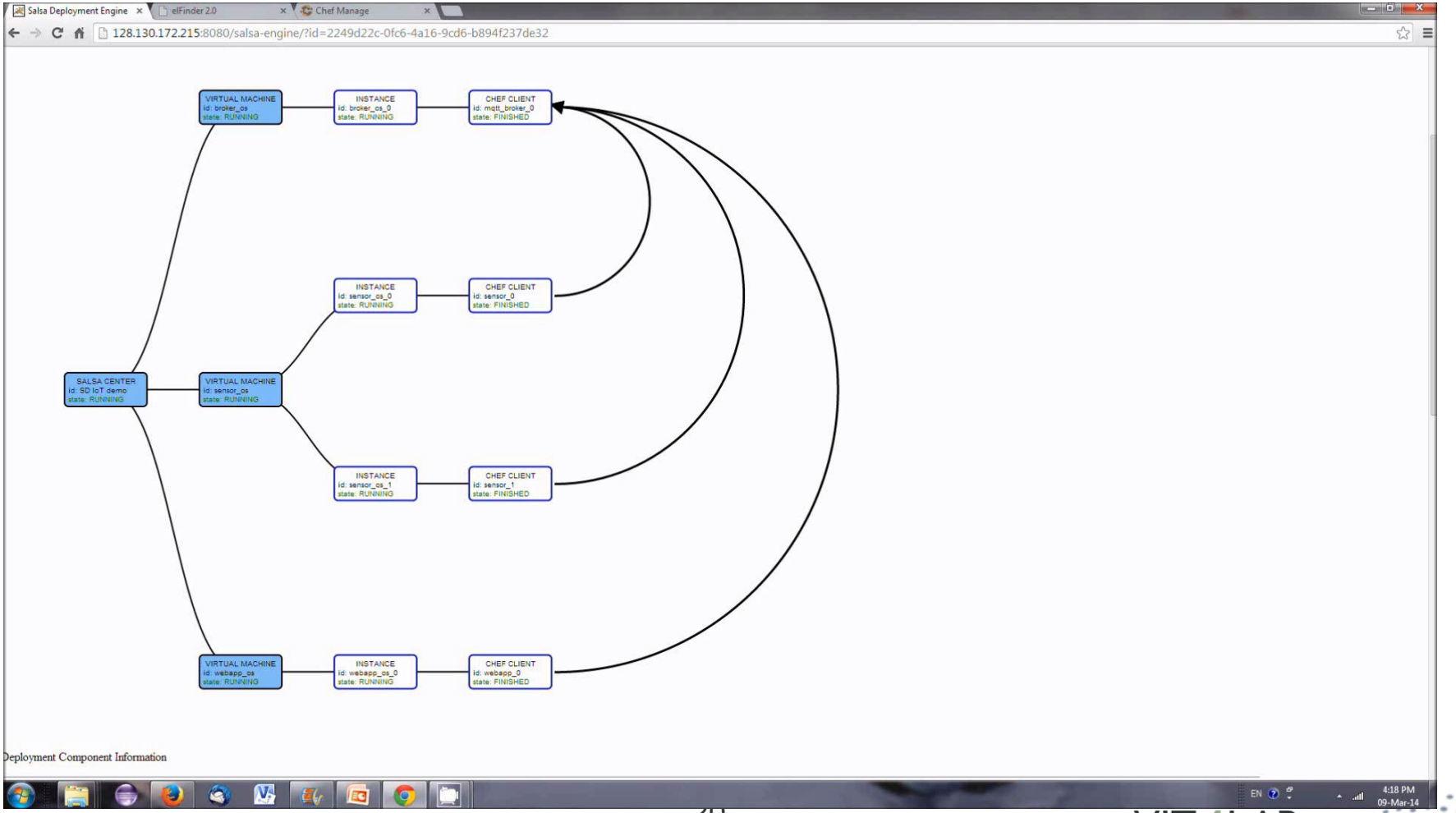
Demo – part 1

Provisioning and deploying virtual topologies atop OpenStack



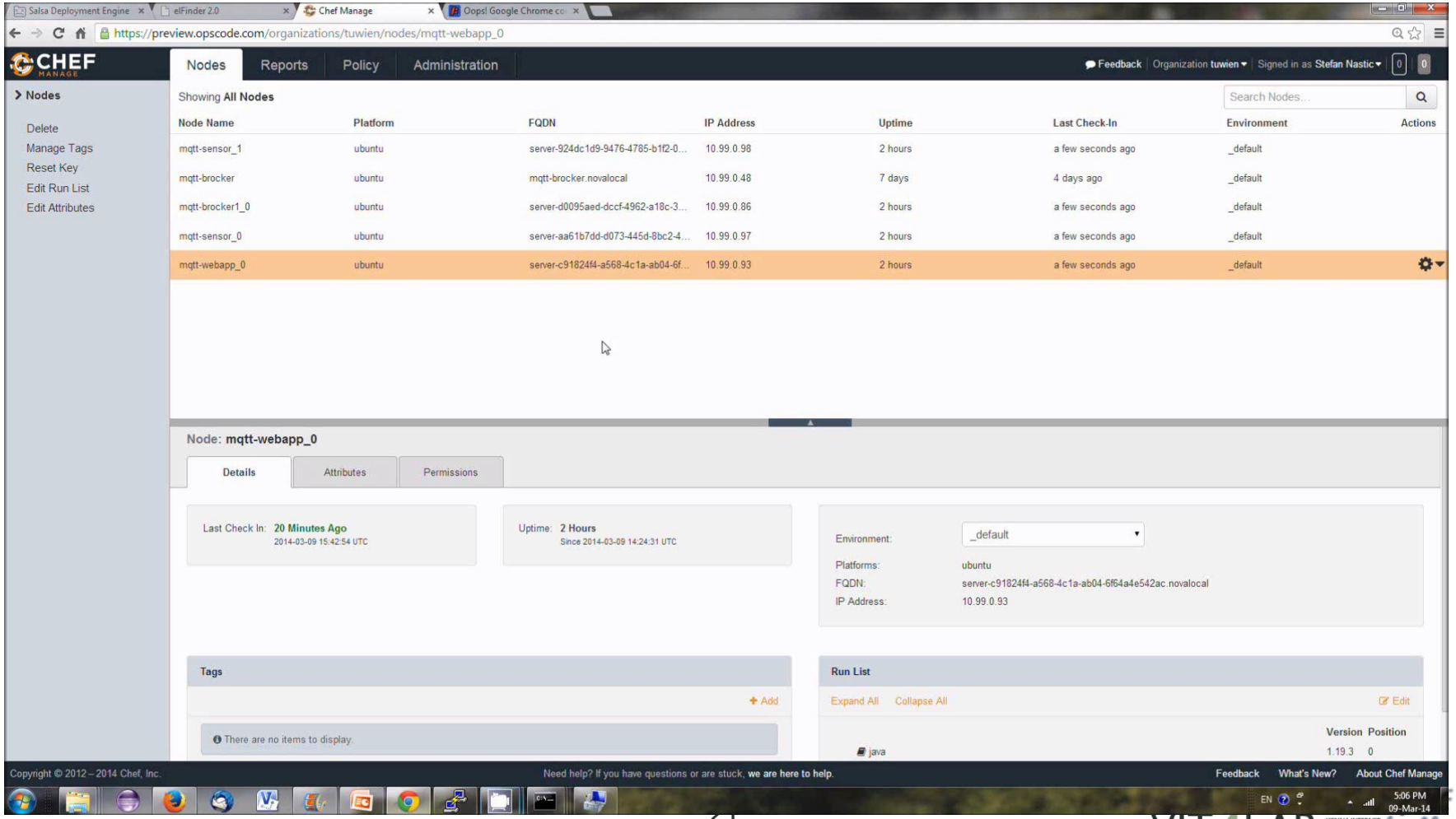
Demo – part 2

- Automated provisioning and configuring individual units



Demo – part 3

- Vehicle tracking application



Coclusion

- Abstracting IoT Cloud resources in a unified manner.
- Fine-grained delivery and consumption of IoT Cloud resources and capabilities (e.g., to support cross-domain applications).
- Simplified provisioning and deploying complex software-defined IoT systems.
- Flexible customization of IoT units, to exactly meet required functional capabilities.
- Automation and managed configuration models.

Thank you for your attention!

Distributed Systems Group
Institute of Information Systems
Vienna University of Technology
Austria

http://dsg.tuwien.ac.at/