Principles for Engineering Elastic IoT Cloud Systems

Hong-Linh Truong

Joint work with Georgiana Copil, Schahram Dustdar, Duc-Hung Le, Daniel Moldovan, Stefan Nastic

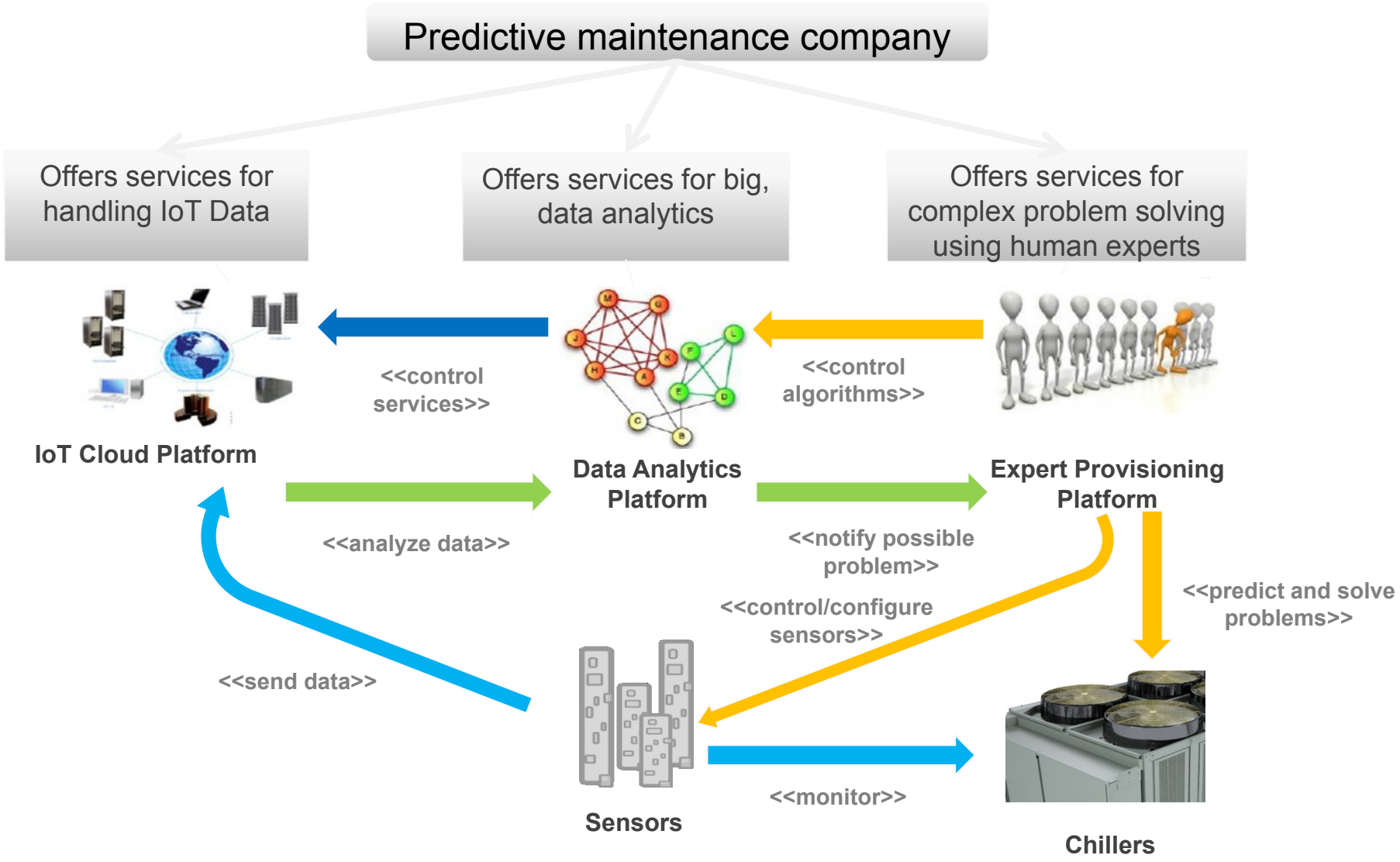
Distributed Systems Group
TU Wien
truong@dsg.tuwien.ac.at
dsg.tuwien.ac.at/research/viecom

Outline

- IoT cloud systems and engineering principles
- Models and techniques
- Tooling
- Demo
- Conclusions and Future Work

Elastic IoT Cloud systems and engineering princinples

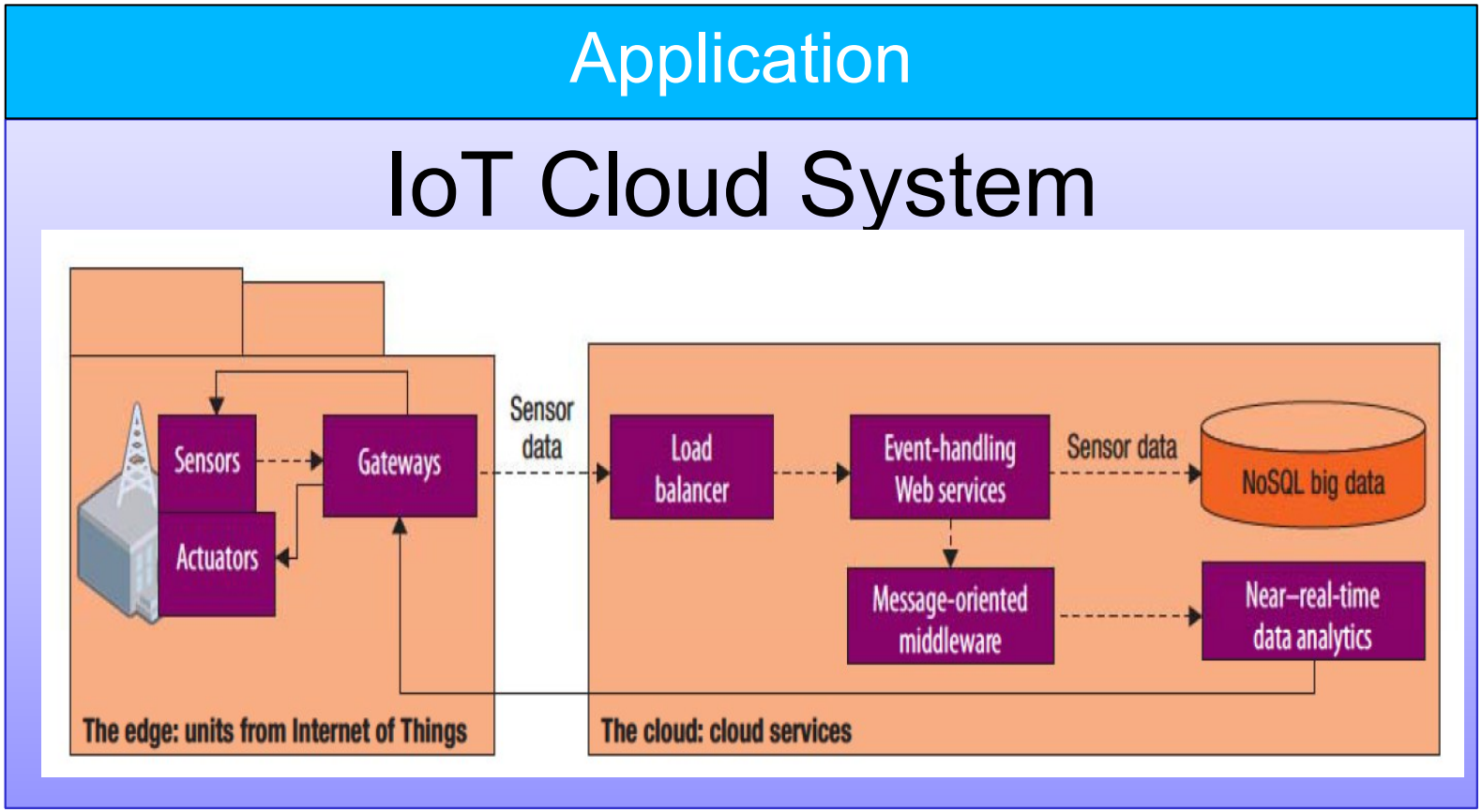
Scenario


Predictive maintenance company
Offers services for handling IoT Data
Offers services for big, data analytics
Offers services for complex problem solving using human experts
<<control services>>
<<control algorithms>>
IoT Cloud Platform
Data Analytics Platform
Expert Provisioning Platform
<<analyze data>>
<<notify possible problem>>
<<predict and solve problems>>
<<control/configure sensors>>
<<send data>>
<<monitor>>
Sensors
Chillers

Elasticity analytics – observations

- Elasticity of IoT elements
  - Activate/change sensor deployment/configurations for required data; changing communication protocols; deploying new sensors
- Elasticity of cloud platform services
  - Deploy/reconfigure cloud services handling changing data
- Elasticity of data analytics
  - Switch and combine different types of data analytics processes and engines due to the severity of problems and quality of results
- Elasticity of teams of human experts
  - Forming and changing different configurations of teams during specific problems and problem severity

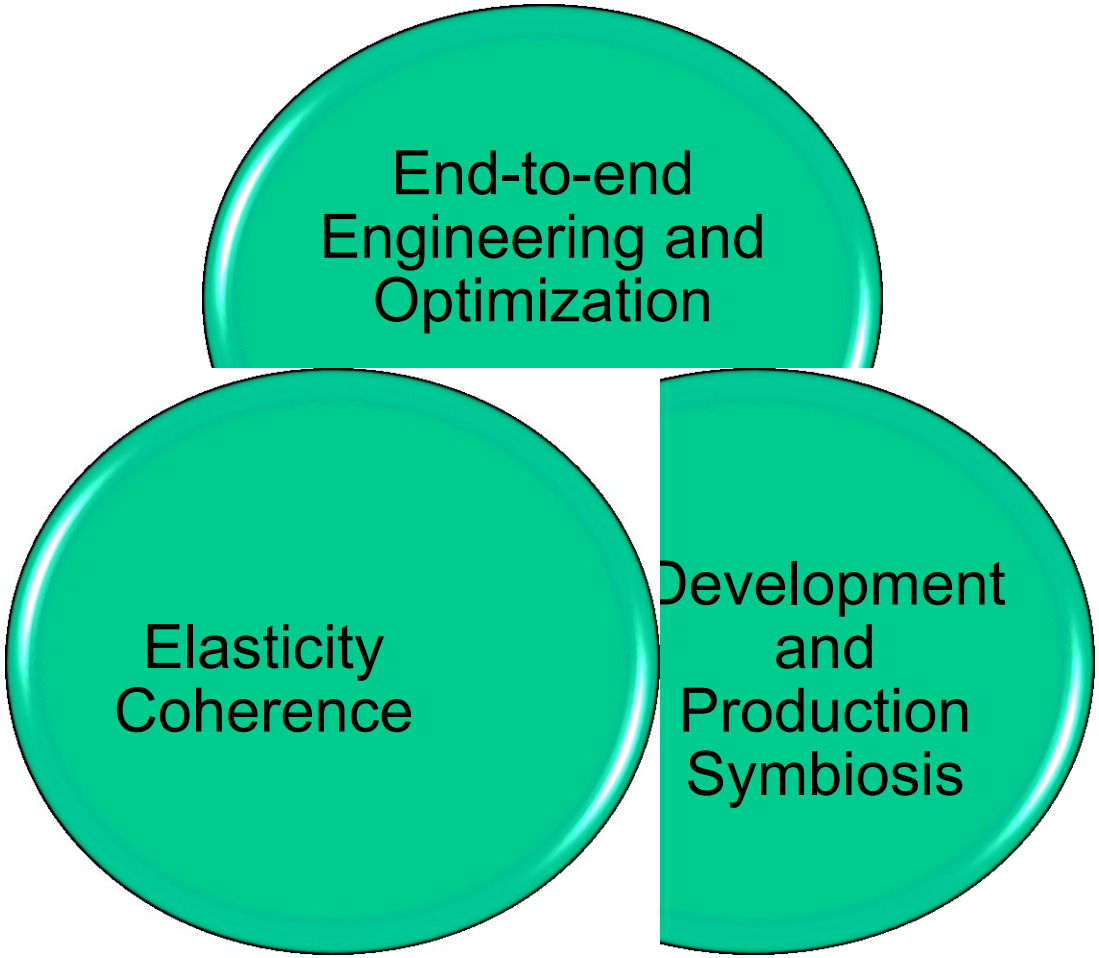
Our view on IoT Cloud Systems



- IoT cloud systems: IoT elements and cloud services
- A coherent view atop IoT elements and cloud services!

Hong Linh Truong, Schahram Dustdar: Programming Elasticity in the Cloud. IEEE Computer 48(3): 87-90 (2015)

Engineering perspectives



Hong Linh Truong, Schahram Dustdar: Principles for Engineering IoT Cloud Systems. IEEE Cloud Computing 2(2): 68-76 (2015)

Principles (1-2)

1. Enable virtualization and composition of IoT components as unit

   Selection, composition, pay-per-use

2. Enable emulated/simulated IoT parts working with production cloud services

   Symbiotic development and operation

Principles (3-5)

3. Enable dynamic provisioning of IoT and cloud service units through uniform marketplaces and repositories for multiple stakeholders

4. Provide multi-level software stack deployment and configuration

5. Provide software-defined elasticity and governance primitive functions for all IoT units and cloud service units

Principles (6-7)

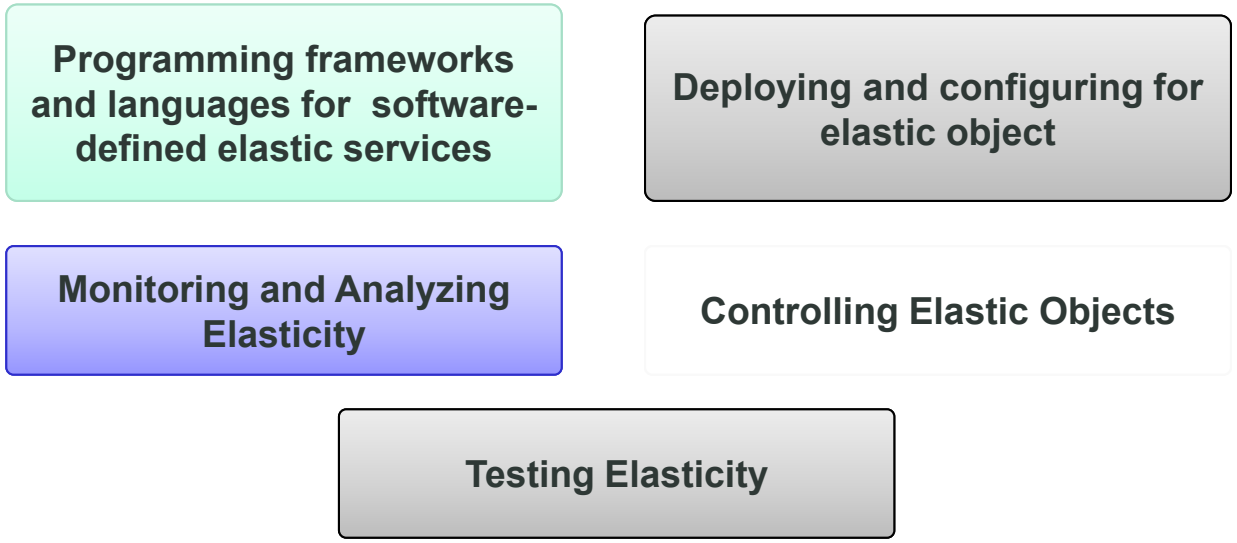
6. Provide monitoring and analysis for an end-to-end view on elasticity and dependability properties

7. Coordinate elasticity to enable a coherent elastic execution through the whole IoT cloud systems

Models & Techniques

Programming Elasticity in IoT Cloud Systems

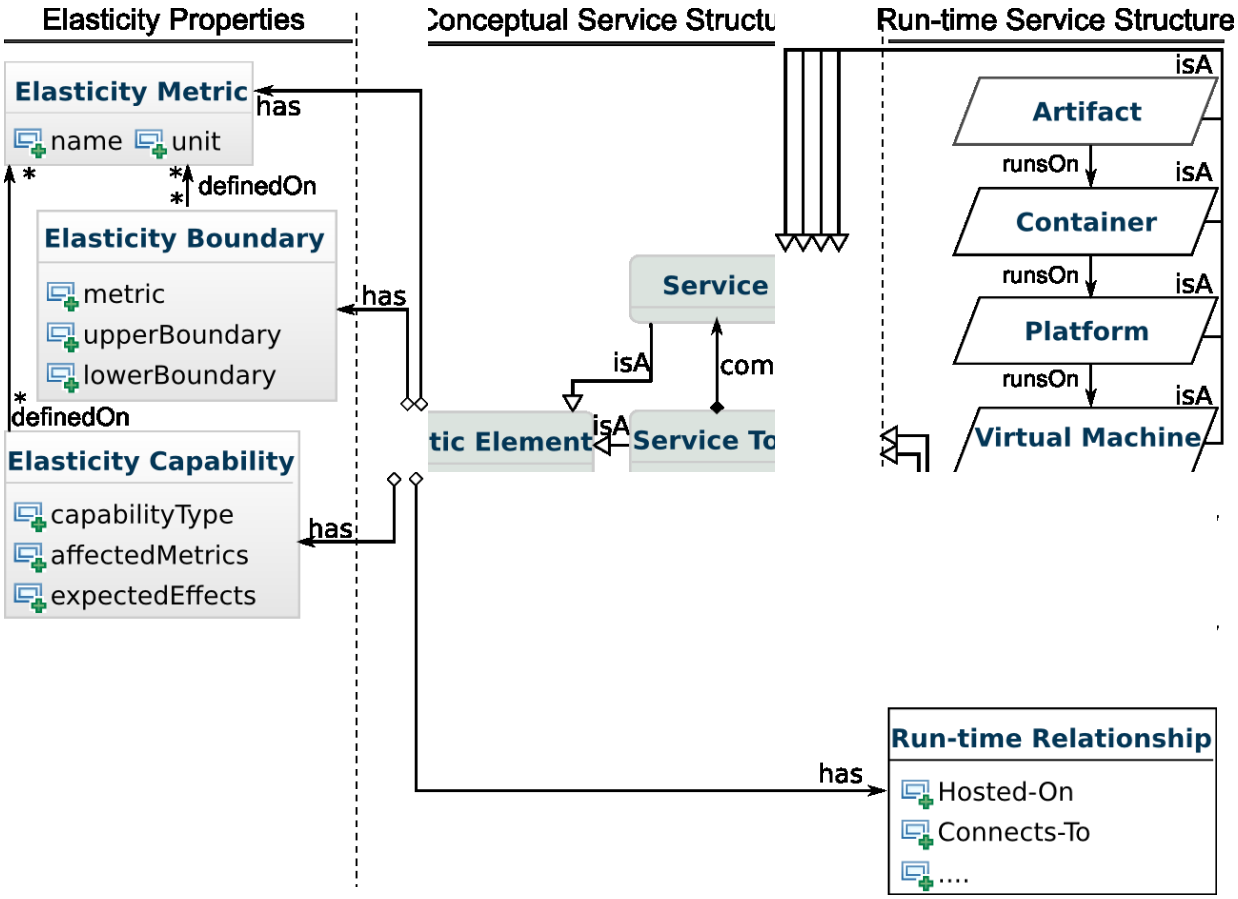
- Conceptualizing elastic objects for IoT elements and cloud services
  - Programming „the world of elastic objects“
- Developing elastic cloud software



Hong Linh Truong, Schahram Dustdar: Programming Elasticity in the Cloud. IEEE Computer 48(3): 87-90 (2015)

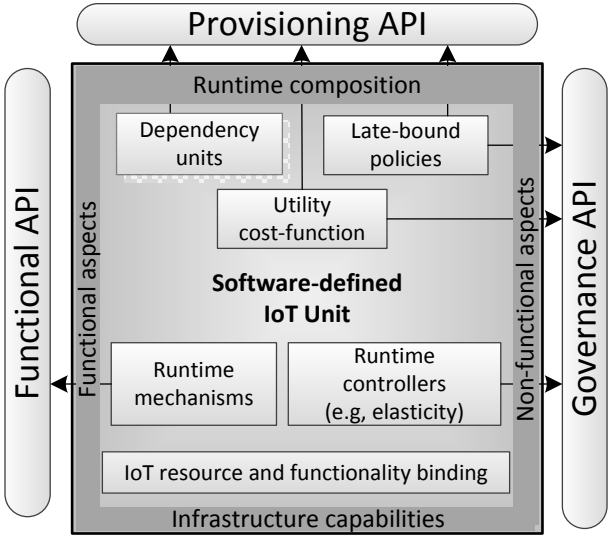
Software-defined Elastic Service

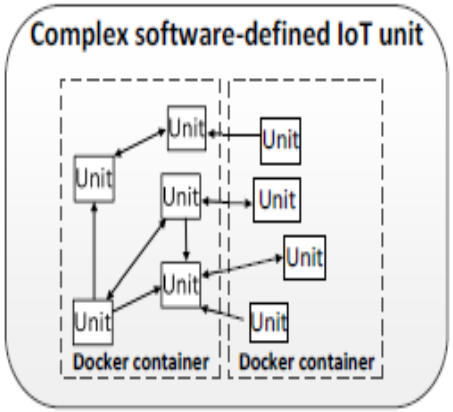
How to represent IoT elements and cloud services under the same view?



Software-Defined IoT Units

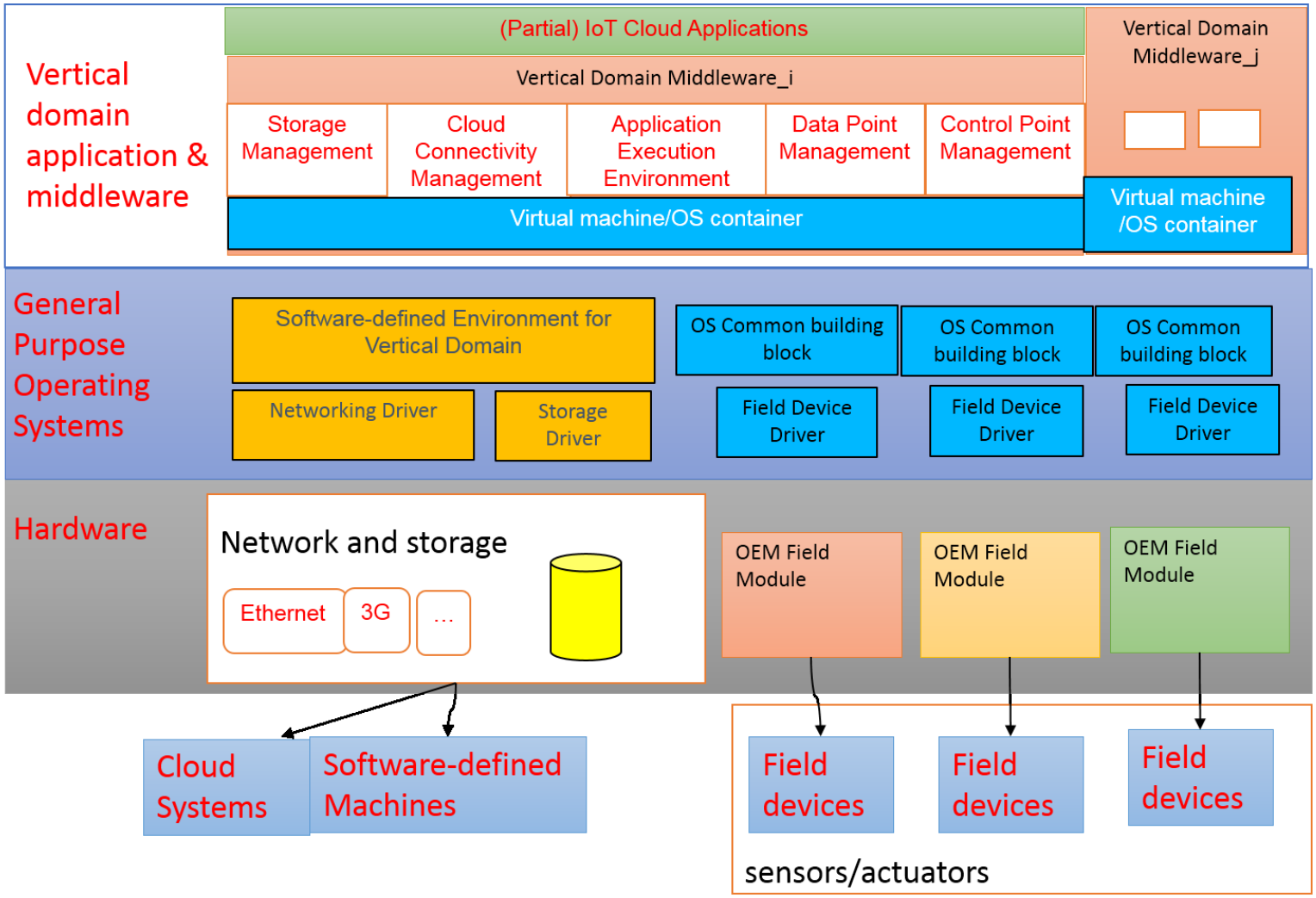
- Virtualizing IoTs resources under “service units” with software-defined API for accessing, configuring and controlling units
- Composing and creating gateways and virtual topologies (of multiple gateways)
- Provisioning (atomic and composite) units dynamically and on-demand in cloud and edge computing environments





Stefan Nastic, Sanjin Sehic, Le-Duc Hung, Hong-Linh Truong, and Schahram Dustdar (2014). Provisioning Software-defined IoT Cloud Systems. The 2nd International Conference on Future Internet of Things and Cloud (FiCloud-2014), August27-29, 2014, Barcelona, Spain.

Software-defined machines (SDMs) for IoT

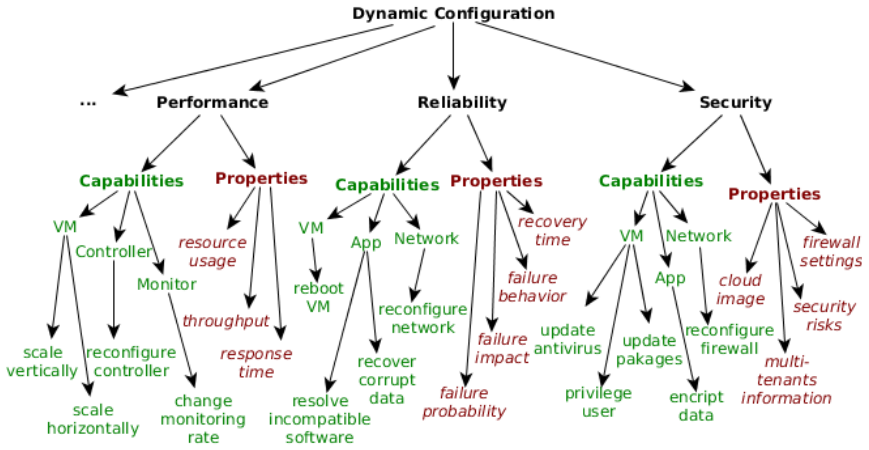


Hong Linh Truong, Schahram Dustdar: Principles for Engineering IoT Cloud Systems. IEEE Cloud Computing 2(2): 68-76 (2015)

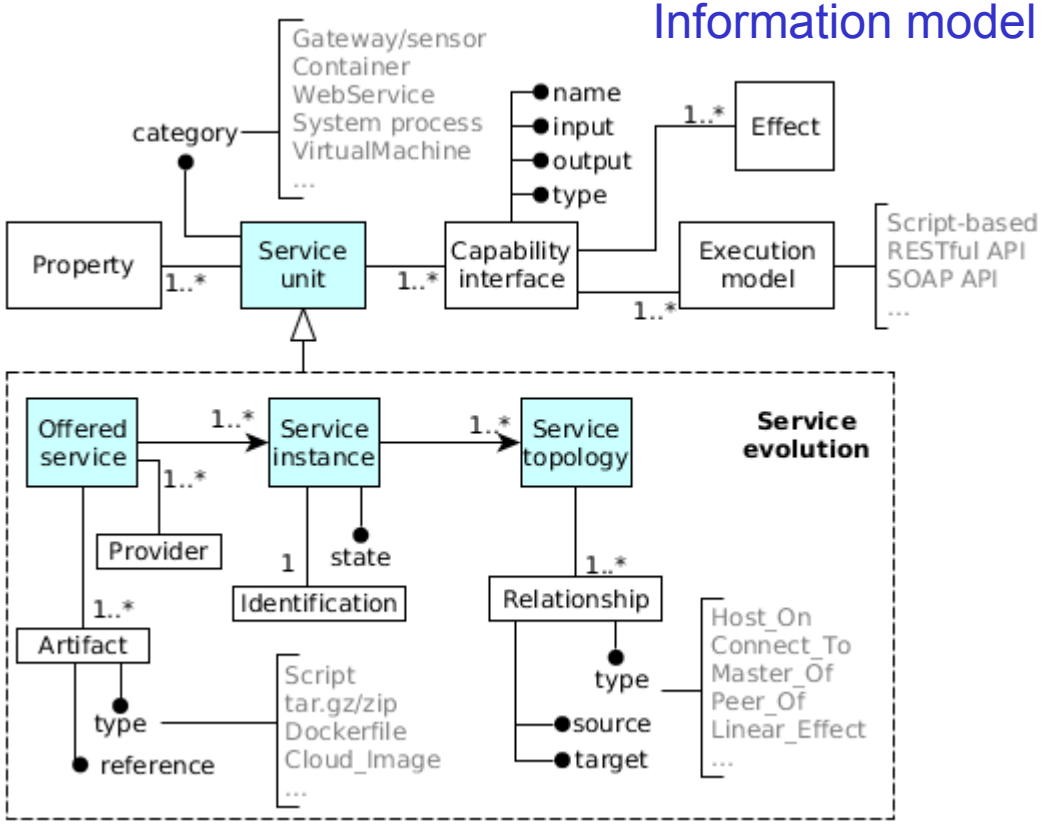
Information for elastic configuration

We must be able to capture different types of information

Types of information



Information model

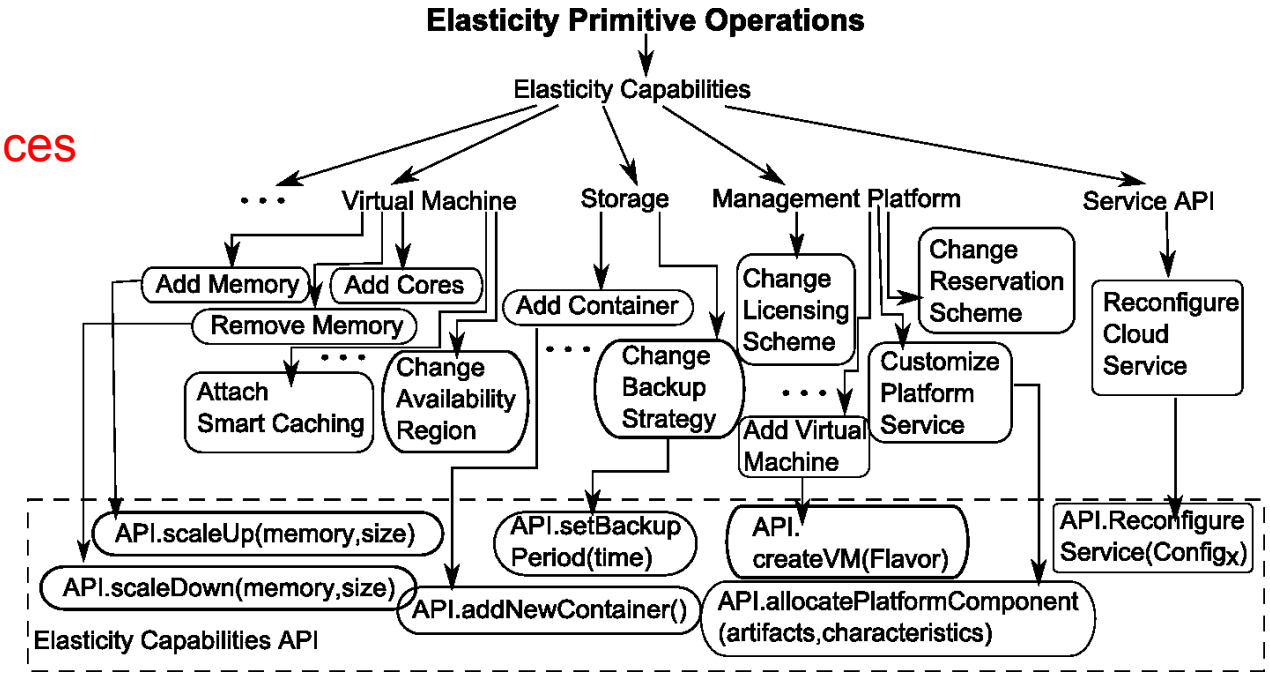


Duc-Hung Le, Hong-Linh Truong and Schahram Dustdar, Managing Information for Dynamic Configuration of Elastic IoT Cloud Systems, June 2015. On submission

Elasticity primitive operations

Primitive operations: actions can be performed on elastic objects to change their elasticity states

For the cloud services

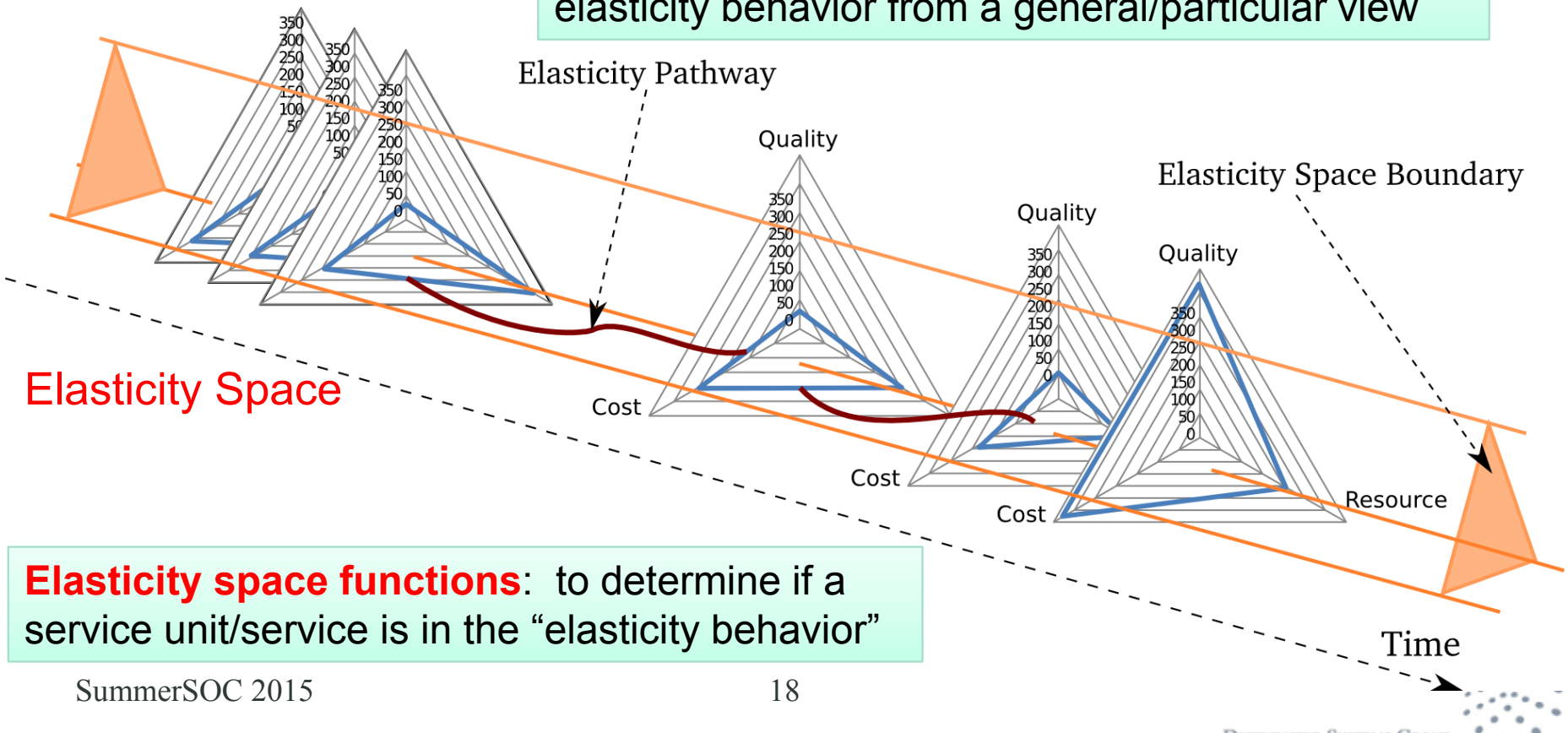


For IoT elements Change communication protocols; change sensor frequency; activating/deactivating sensors, gateways configuration, etc.

Elasticity Model for Cloud Services

Moldovan D., G. Copil,Truong H.-L., Dustdar S. (2013). **MELA: Monitoring and Analyzing Elasticity of Cloud Service. CloudCom 2013**

Elasticity Pathway functions: to characterize the elasticity behavior from a general/particular view

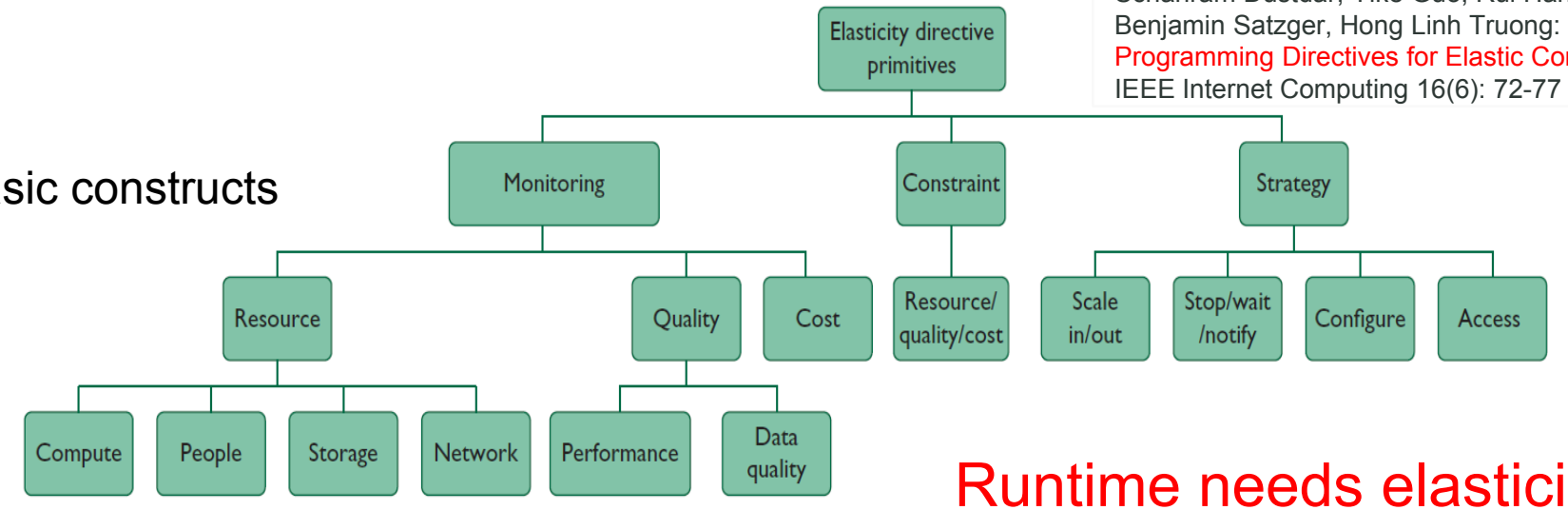


Elasticity space functions: to determine if a service unit/service is in the "elasticity behavior"

Specifying and controling elasticity

Schahram Dustdar, Yike Guo, Rui Han, Benjamin Satzger, Hong Linh Truong: Programming Directives for Elastic Computing. IEEE Internet Computing 16(6): 72-77 (2012)

Basic constructs



SYBL (Simple Yet Beautiful Language) for specifying elasticity requirements

SYBL-supported requirement levels

- Cloud Service Level
- Service Topology Level
- Service Unit Level
- Relationship Level
- Programming/Code Level

Runtime needs elasticity primitive opertations!

Current SYBL implementation

in Java using Java annotations

```
@SYBLAnnotation(monitoring=„“,constraints=„“,strategies=„
    “)
```

in XML

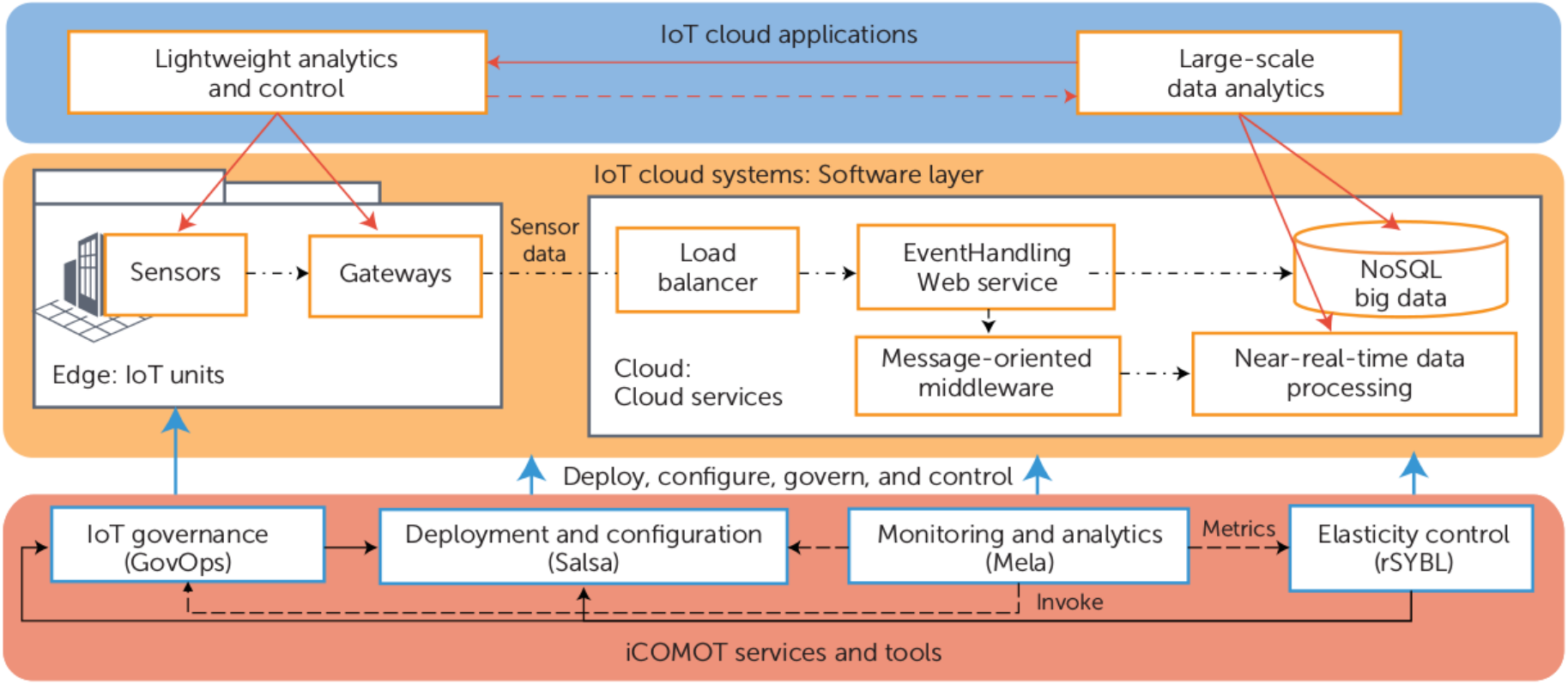
```
<ProgrammingDirective><Constraints><Constraint
    name=c1>...</Constraint></Constraints>...</Programm
    ingDirective>
```

as TOSCA Policies

```
<tosca:ServiceTemplate name="PilotCloudService">
    <tosca:Policy name="St1"
    policyType="SYBLStrategy">    St1:STRATEGY
    minimize(Cost) WHEN high(overallQuality)
    </tosca:Policy>...
```

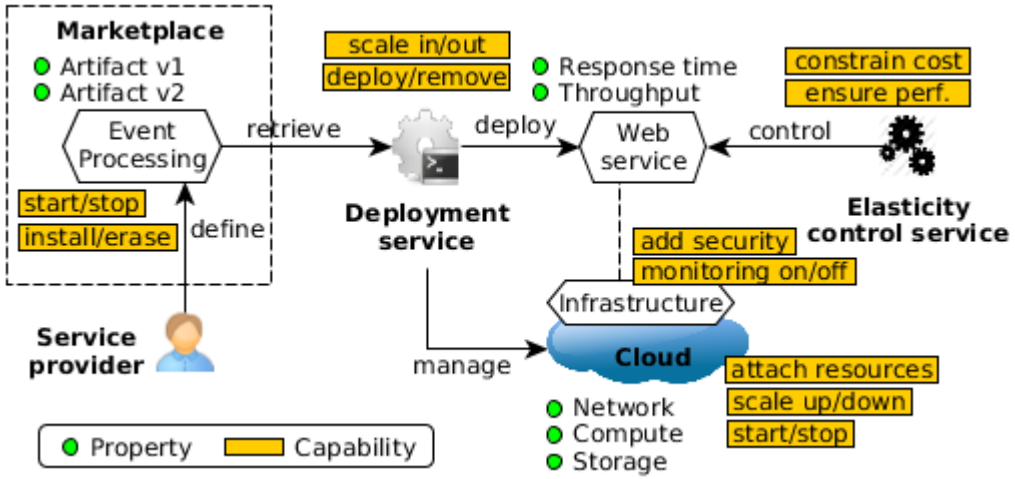
TOOLS

Monitoring, Controlling and Testing IoT Cloud Systems

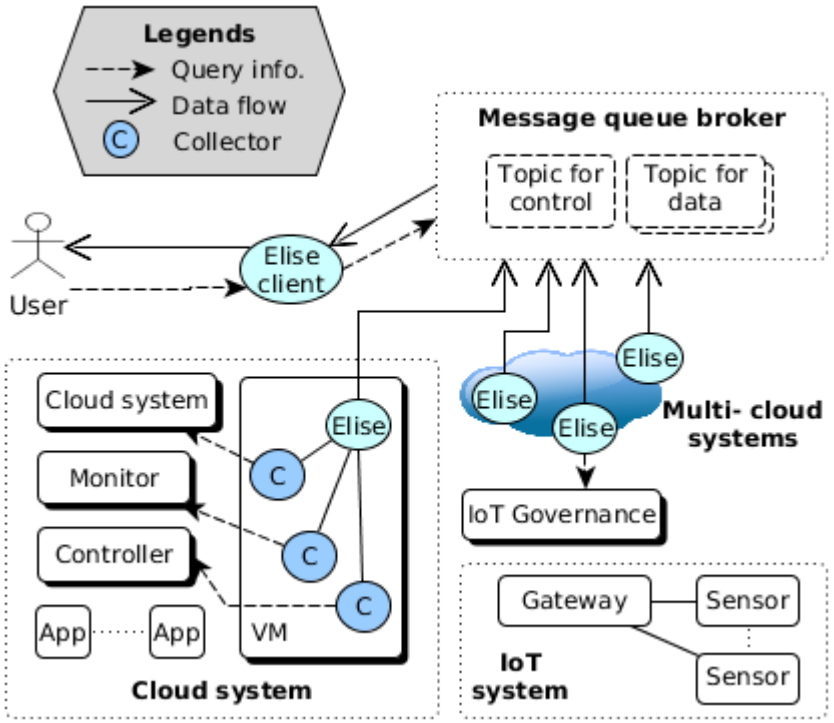


Check: http://tuwiendsg.github.io/iCOMOT/demo.html

Elasticity Information as a Service



Scalable and extensible runtime system



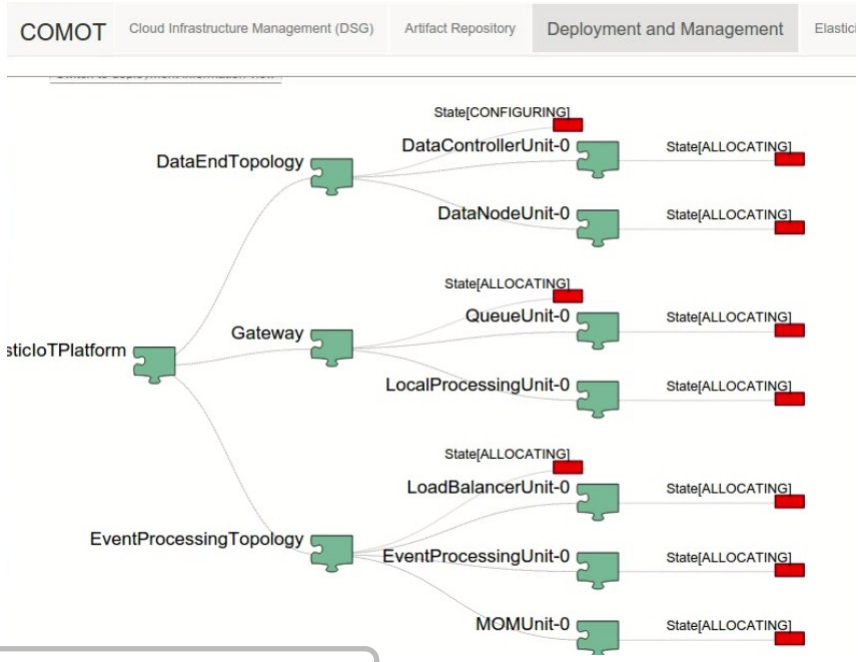
Collecting configuration information from different phases

https://github.com/tuwiendsg/ELISE

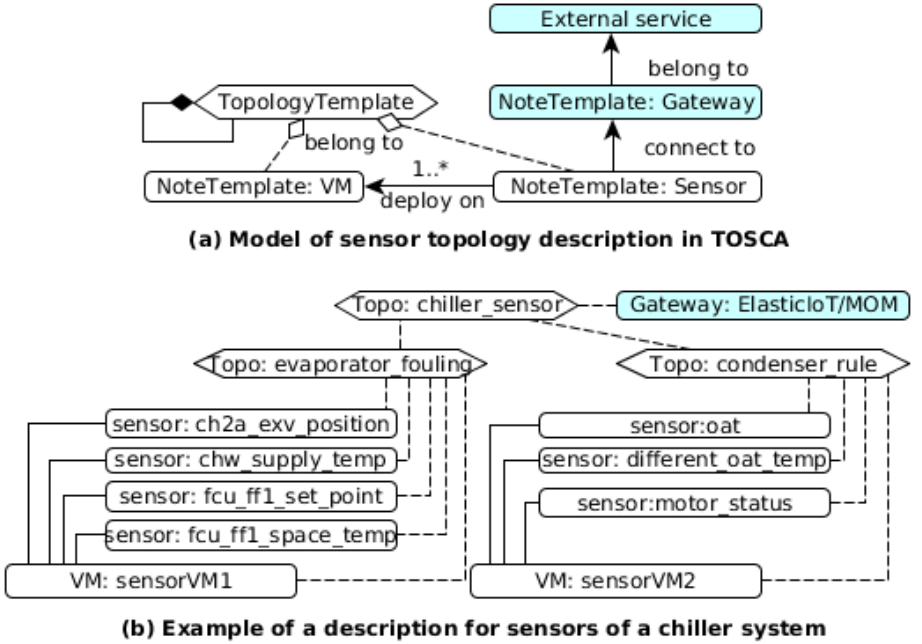
Duc-Hung Le, Hong-Linh Truong and Schahram Dustdar, Managing Information for Dynamic Configuration of Elastic IoT Cloud Systems, June 2015. On submission

SALSA- Multi-cloud, multi-stack, complex topologies configuration

- Well-defined APIs for manipulating and provisioning objects
- Support different types of objects, e.g., VMs, OS containers, services, service containers, IoT sensors, and gateways



Data center services



Sensors

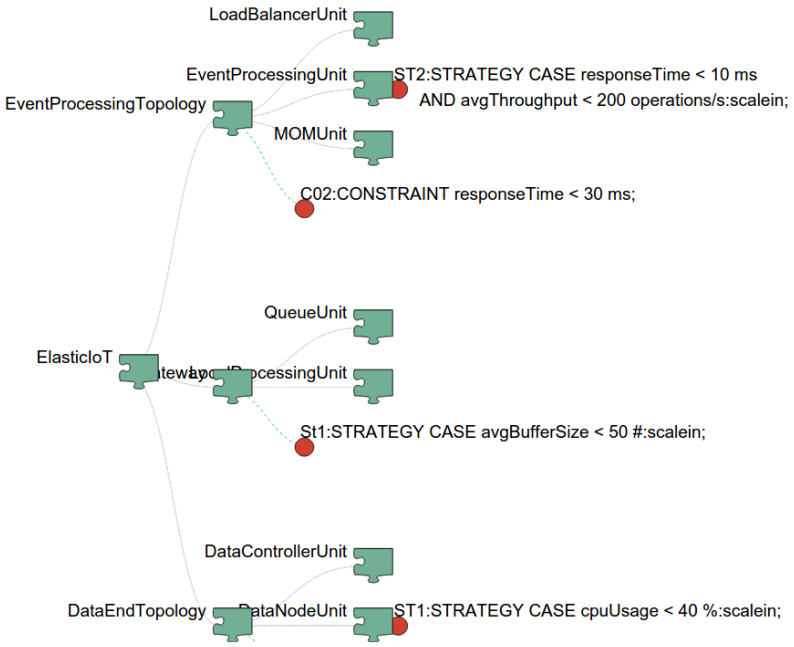
https://github.com/tuwiendsg/SALSA

High level elasticity control

#SYBL.CloudServiceLevel
Cons1: CONSTRAINT responseTime < 5 ms
Cons2: CONSTRAINT responseTime < 10 ms
WHEN nbOfUsers > 10000
Str1: STRATEGY CASE fulfilled(Cons1) OR fulfilled(Cons2): minimize(cost)

#SYBL.ServiceUnitLevel
Str2: STRATEGY CASE ioCost < 3 Euro : maximize(dataFreshness)

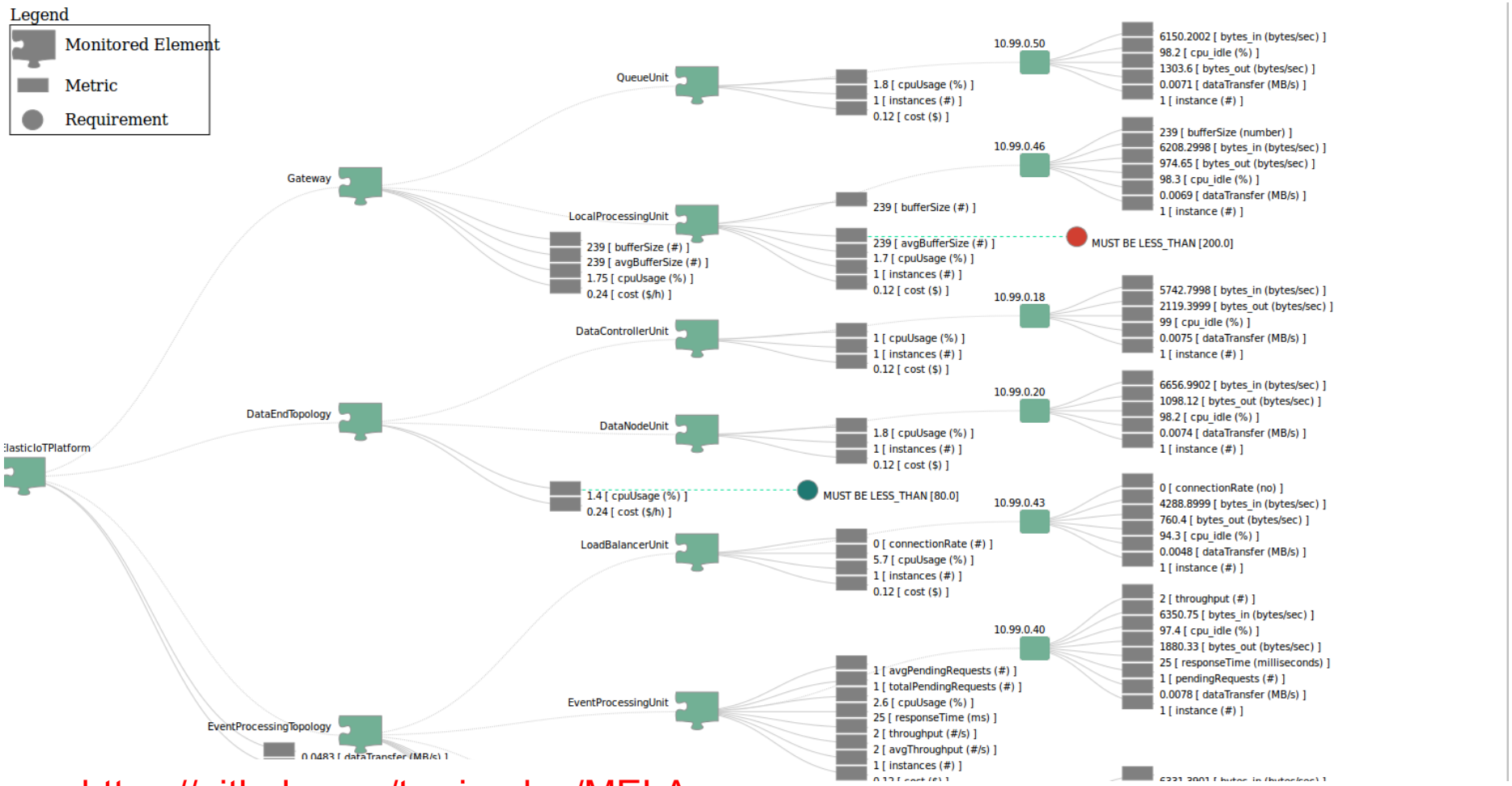
#SYBL.CodeRegionLevel
Cons4: CONSTRAINT dataAccuracy>90% AND cost<4 Euro



https://github.com/tuwiendsg/rSYBL

Georgiana Copil, Daniel Moldovan, Hong-Linh Truong, Schahram Dustdar, **"SYBL: an Extensible Language for Controlling Elasticity in Cloud Applications"**, 13th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), May 14-16, 2013, Delft, Netherlands

Elasticity space and pathway analytics

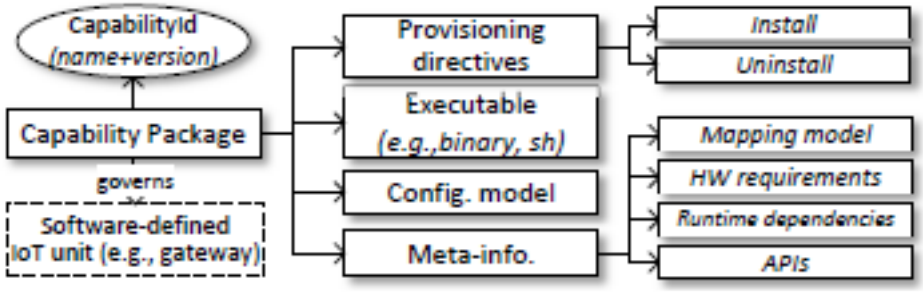


https://github.com/tuwiendsg/MELA

Daniel Moldovan, Georgiana Copil, Hong-Linh Truong, Schahram Dustdar, "MELA: Elasticity Analytics for Cloud Services", International Journal of Big Data Intelligence, 2015, Vol. 2, No. 1

rtGovOps – Governance capabilities

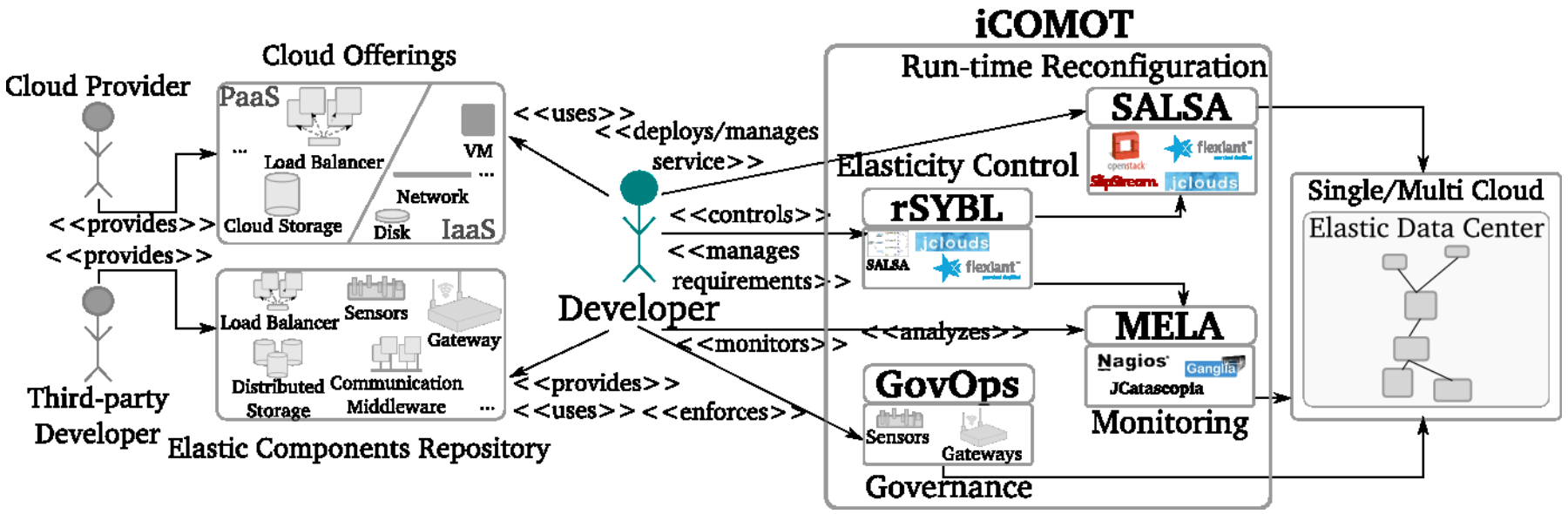
- Governance capabilities:
  - Any function that „manipulates“ an IoT cloud resource
  - Building blocks of operational governance (GovOps) processes
  - Executed „inside“ software-defined machines (SDMs)
- Governance processes/strategies
  - Functional configuration
  - Performance
  - Uncertainty study
  - Risk study



https://github.com/tuwiendsg/GovOps/

Stefan Nastic, Michael Vögler, Christian Inzinger, Hong-Linh Truong, Schahram Dustdar, "rtGovOps: A Runtime Framework for Governance in Large-scale Software-defined IoT Cloud Systems", The 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, 2015

iCOMOT -- Toolsets and actions for IoT Cloud Systems



http://tuwiendsg.github.io/iCOMOT/

Hong-Linh Truong, Georgiana Copil, Schahram Dustdar, Duc-Hung Le, Daniel Moldovan, Stefan Nastic, "iCOMOT – a Toolset for Managing IoT Cloud Systems", 16th IEEE International Conference on Mobile Data Management, 15-18 June, 2015, Pittsburg, USA. (Demo)

http://tuwiendsg.github.io/iCOMOT/

DEMO

Conclusions and Outlook

- **Engineering IoT cloud systems**
  - Deal with complex IoT elements and cloud services
  - Coordinating elasticity across IoT platforms and cloud platforms is needed
  - Engineering an end-to-end elasticity for IoT cloud systems needs a complex set of tools
- **Ongoing work**
  - Coordinated elasticity control for people and data elasticity in IoT cloud systems (ICSOC submissions)
  - Using iCOMOT to support testing, privacy/risk and uncertainty studies for IoT cloud systems
  - Data elasticity management in IoT cloud systems

Thanks for your attention!

Questions?

Hong-Linh Truong

Distributed Systems Group
TU Wien

dsg.tuwien.ac.at/research/viecom