

# Novel Models and Techniques for Monitoring and Analysis of Software-defined Elastic Systems

Daniel Moldovan  
Distributed Systems Group,  
Vienna University of Technology

[d.moldovan@dsg.tuwien.ac.at](mailto:d.moldovan@dsg.tuwien.ac.at)  
<http://www.infosys.tuwien.ac.at/staff/dmoldovan/>

# Agenda

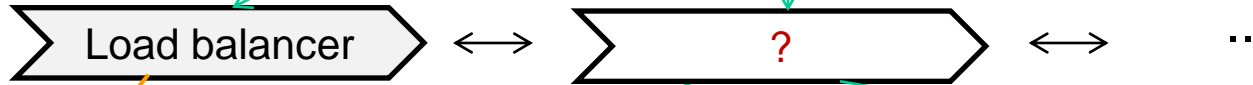
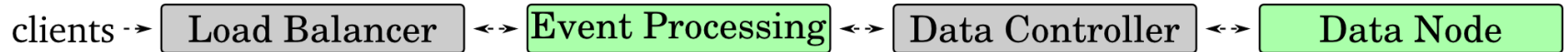
- Analyzing Elasticity of cloud offered services
  - Quantifying elasticity of cloud offered services
- Monitoring elastic cloud services
- Analyzing Elasticity of cloud services
  - Elasticity Space, Boundaries and Pathway
  - Elasticity Dependencies

# Analyzing Elasticity of cloud offered services

## Motivation

Static

Elastic



**Cost:**  
/H & / GB



**Cost:**

- OnDemand
- Spot
- Reserved (1/3 years)

**Cost:**  
/size/month

**Cost:**

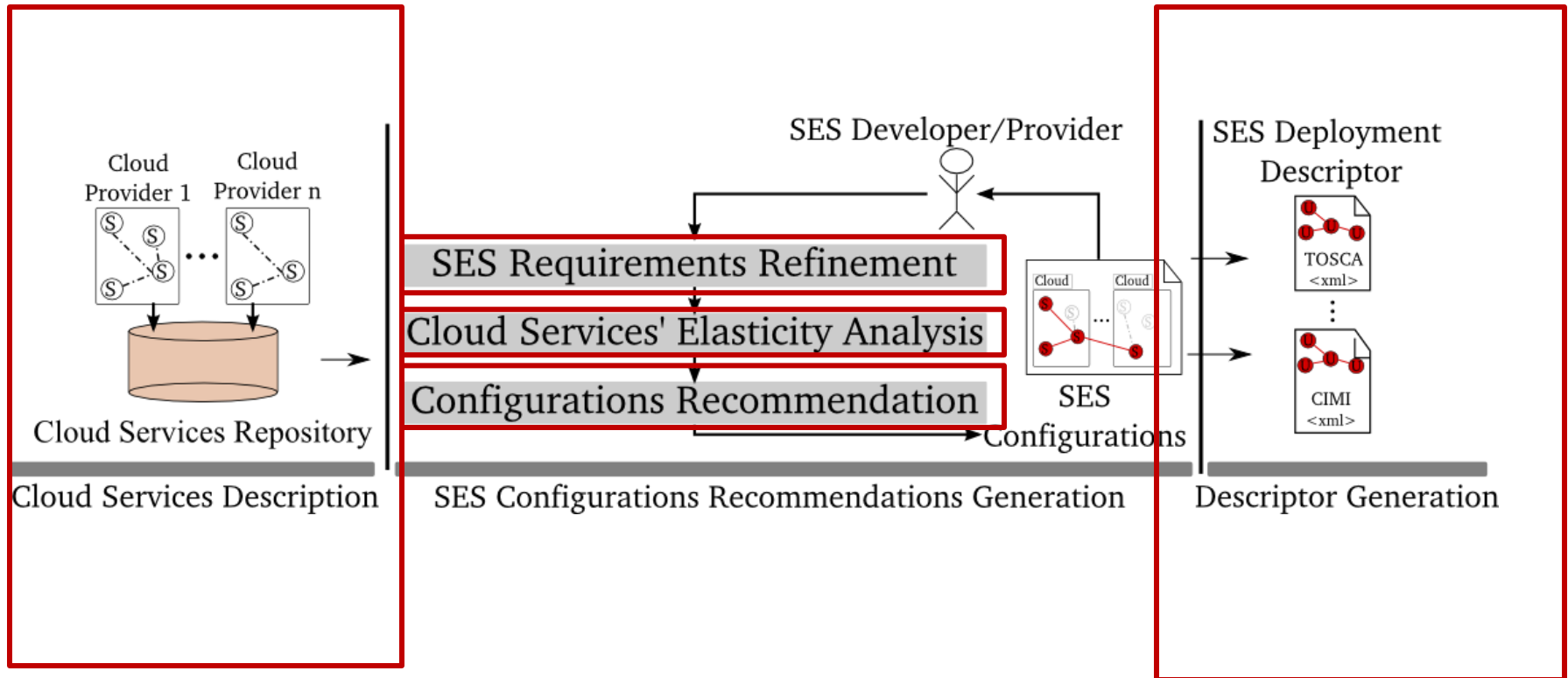
- OnDemand
- Reserved (1/3 years)

**Cost:**  
/size/month



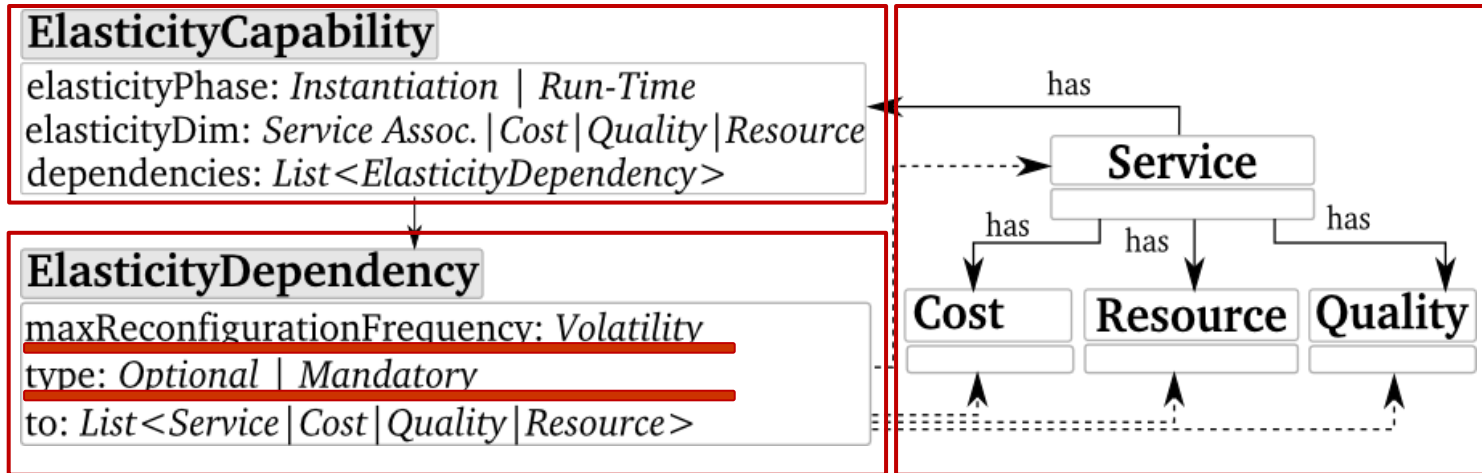
# Analyzing Elasticity of cloud offered services

## Approach



# Analyzing Elasticity of cloud offered services

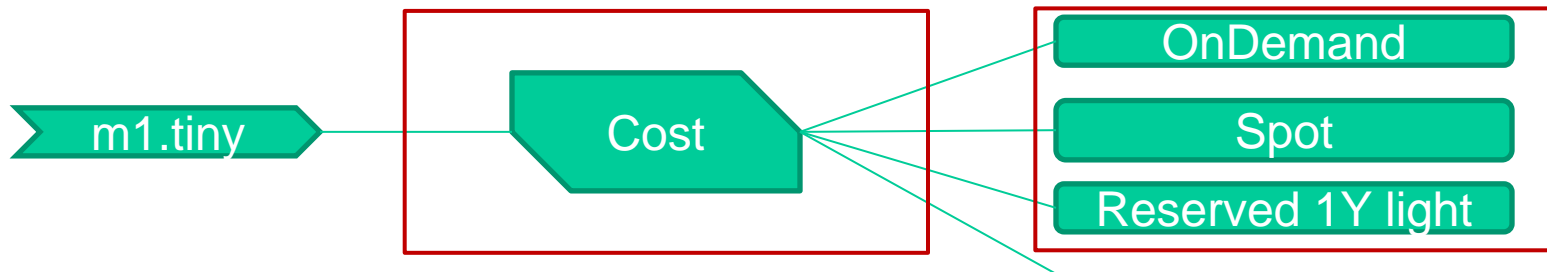
## Modelling Elasticity Capabilities



Service Unit

Elasticity Capability

Elasticity Dependencies



# Analyzing Elasticity of cloud offered services

## Customizable Elasticity Quantification Coefficients

- Elasticity Phase Quantification Coefficients

$$ElPhaseQ(p) = \begin{cases} v_i & \text{if Instantiation Time} \\ v_r & \text{if Run Time} \\ v_{ir} & \text{if Both} \end{cases}$$

### ElasticityCapability

elasticityPhase: *Instantiation* | *Run-Time*  
 elasticityDim: *Service Assoc.* | *Cost* | *Quality* | *Resource*  
 dependencies: *List*<*ElasticityDependency*>

- Elasticity Dependency Type Quantification Coefficients

$$ElDependencyQ(d) = \begin{cases} v_o & \text{if Optional Association} \\ v_m & \text{if Mandatory Association} \end{cases}$$

### ElasticityDependency

maxReconfigurationFrequency: *Volatility*  
 type: *Optional* | *Mandatory*  
 to: *List*<*Service* | *Cost* | *Quality* | *Resource*>

- Elasticity Dependency Volatility Quantification Coefficients

$$VolatilityQ(d)$$

- Elasticity Capability Quantification Function

$$ECQ(c) = ElPhaseQ(c.phase) * \sum_{d \in c.dependencies} VolatilityQ(d) * ElDependencyQ(d)$$

- Elasticity Quantification Function

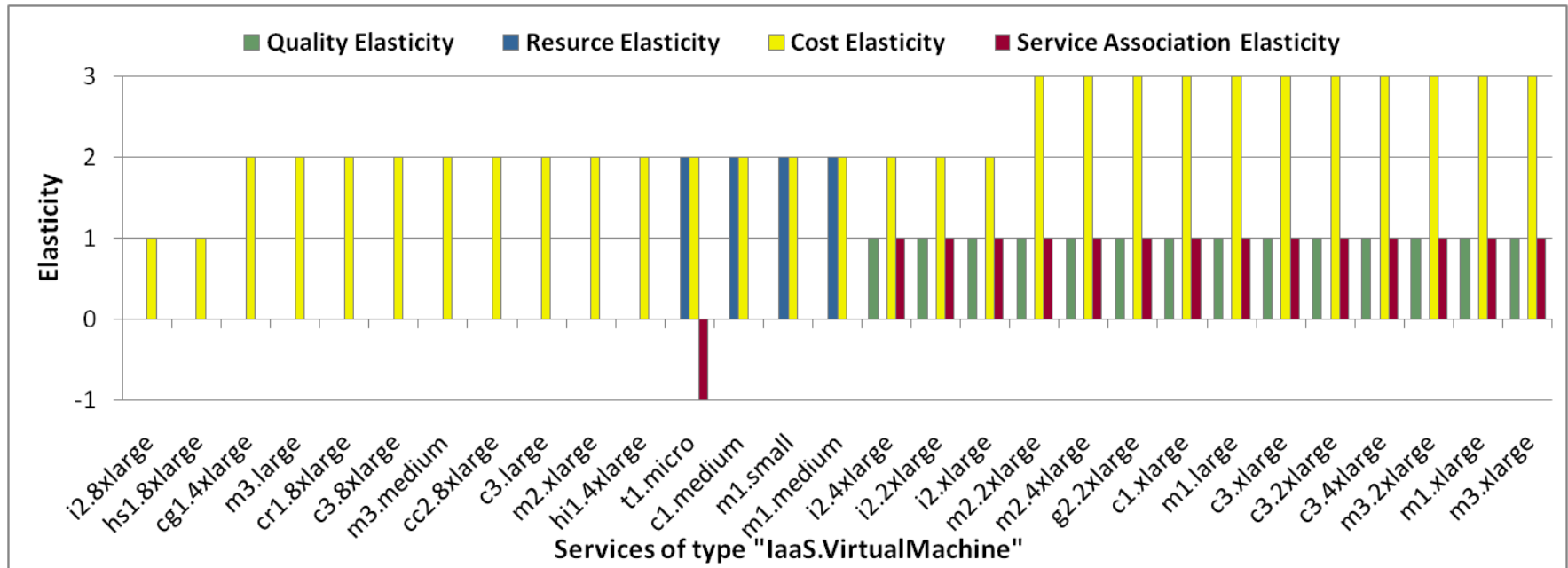
$$EQ(s) = \sum_{d \in cost, quality, resource} W(d) * \sum_{c \in s.capabilities(d)} ECQ(c)$$



# Analyzing Elasticity of cloud offered services

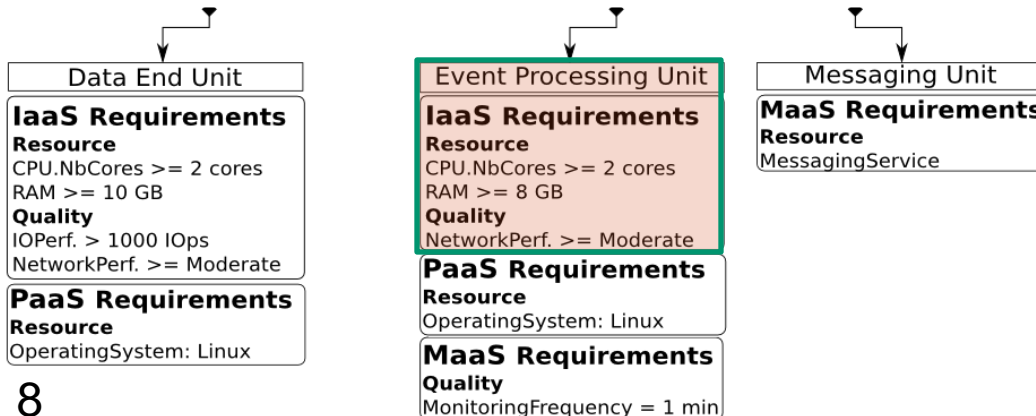
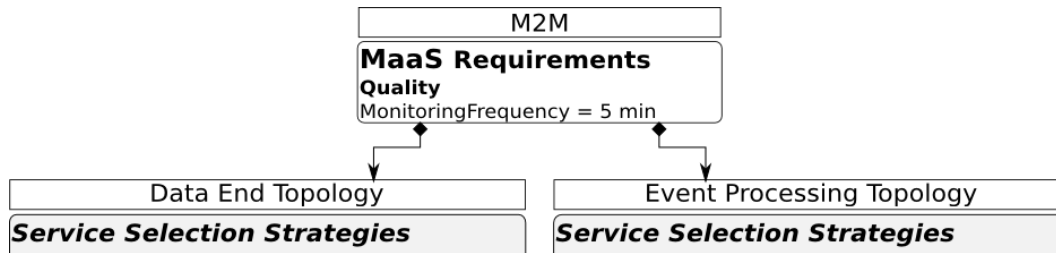
## Example: Elasticity of Amazon EC2 IaaS Services

- Elasticity Phase Quantification Coefficient
  - $$ElPhaseQ(p) = \begin{cases} 1 & \text{if Instantiation Time} \\ 2 & \text{if Run Time} \\ 3 & \text{if Both} \end{cases}$$
- Elasticity Dependency Type Quantification Coefficient
  - $$ElDependencyQ(d) = \begin{cases} 1 & \text{if Optional Association} \\ -1 & \text{if Mandatory Association} \end{cases}$$
- Elasticity Dependency Volatility Quantification Coefficient



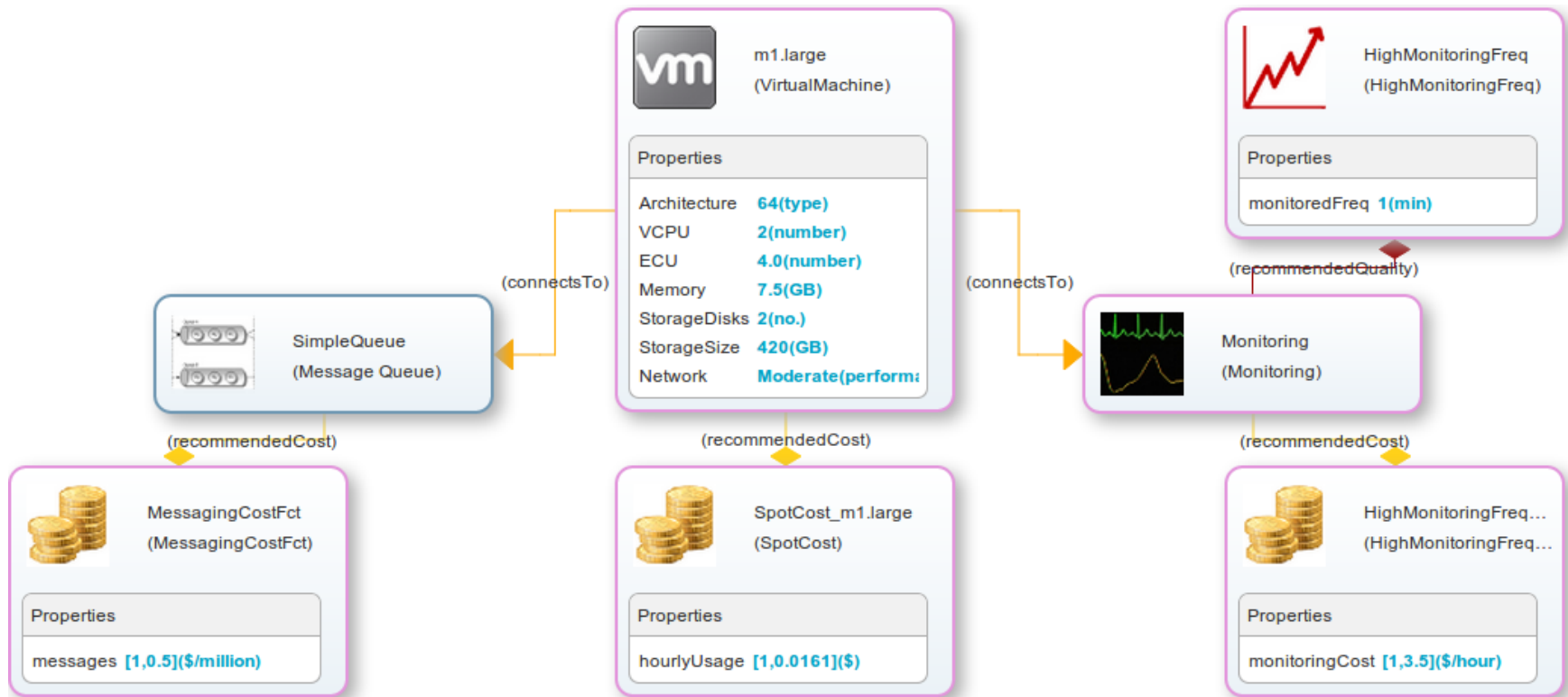
# Analyzing Elasticity of cloud offered services

Service Selection Strategies	Recommended Services	Quality Elasticity			Cost Elasticity		
		Avg.	Min.	Max.	Avg.	Min.	Max.
Max Requirements	23	0.6	0	1	2.39	1.0004	3.0004





# Analyzing Elasticity of cloud offered services



# Agenda

- Analyzing Elasticity of cloud offered services
  - Quantifying elasticity of cloud offered services
- **Monitoring elastic cloud services**
- Analyzing Elasticity of cloud services
  - Elasticity Space, Boundaries and Pathway
  - Elasticity Dependencies

# Monitoring elastic cloud services

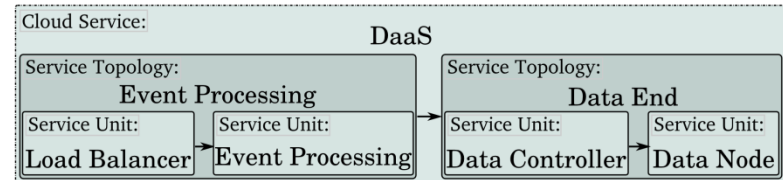
## Service Level Monitoring

- Response time
- Number of clients
- Other specific metrics

## System Level Monitoring

- Ganglia, Nagios, etc.
- CPU usage
- Memory usage
- Network transfer

Design-time View



### Controlling the service's elasticity

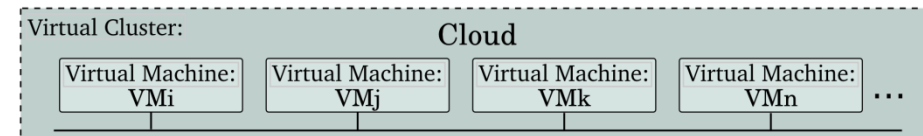
User-Defined Requirements violation:

- Cost per client too high

Reasons:

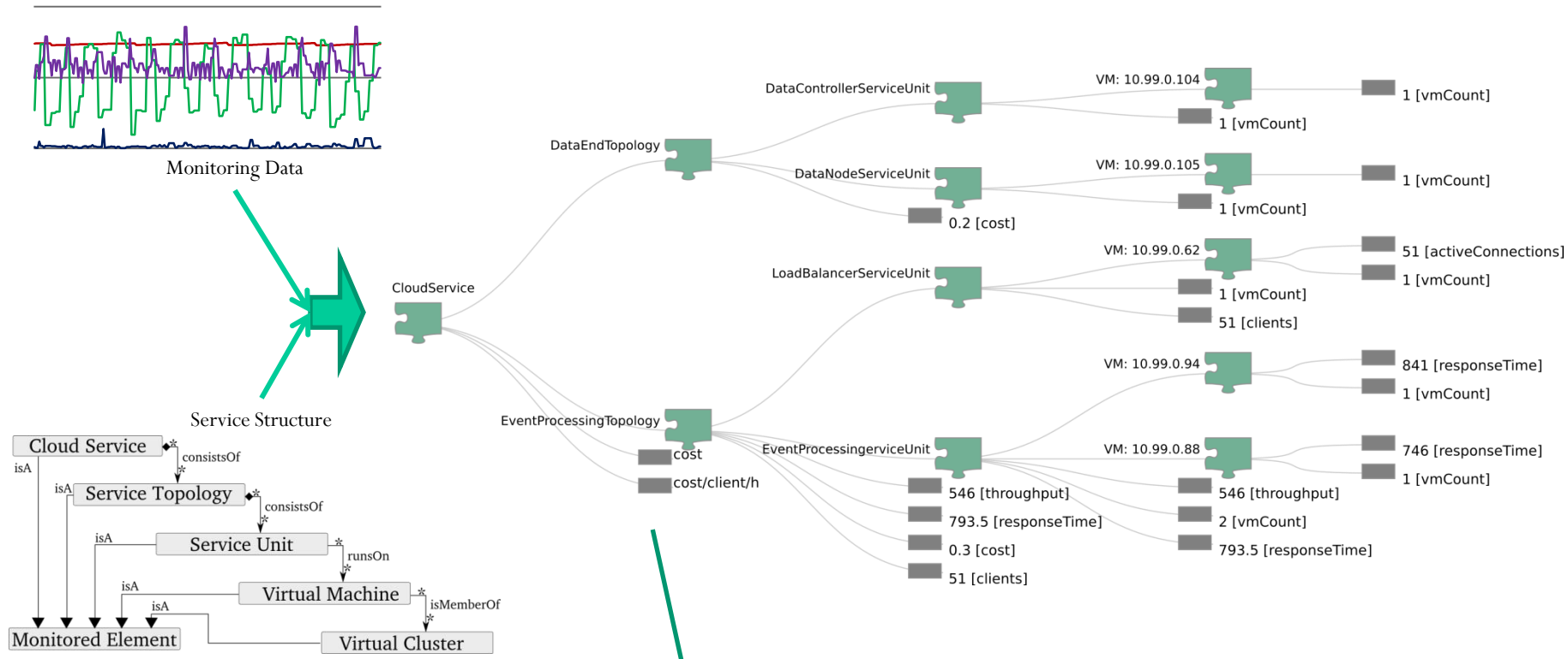
- Too much logging? Monitoring chatter?
- Too expensive VMs? Which one can be downsized?
- Not enough clients? Why?

Virtual Infrastructure View



# Monitoring elastic cloud services

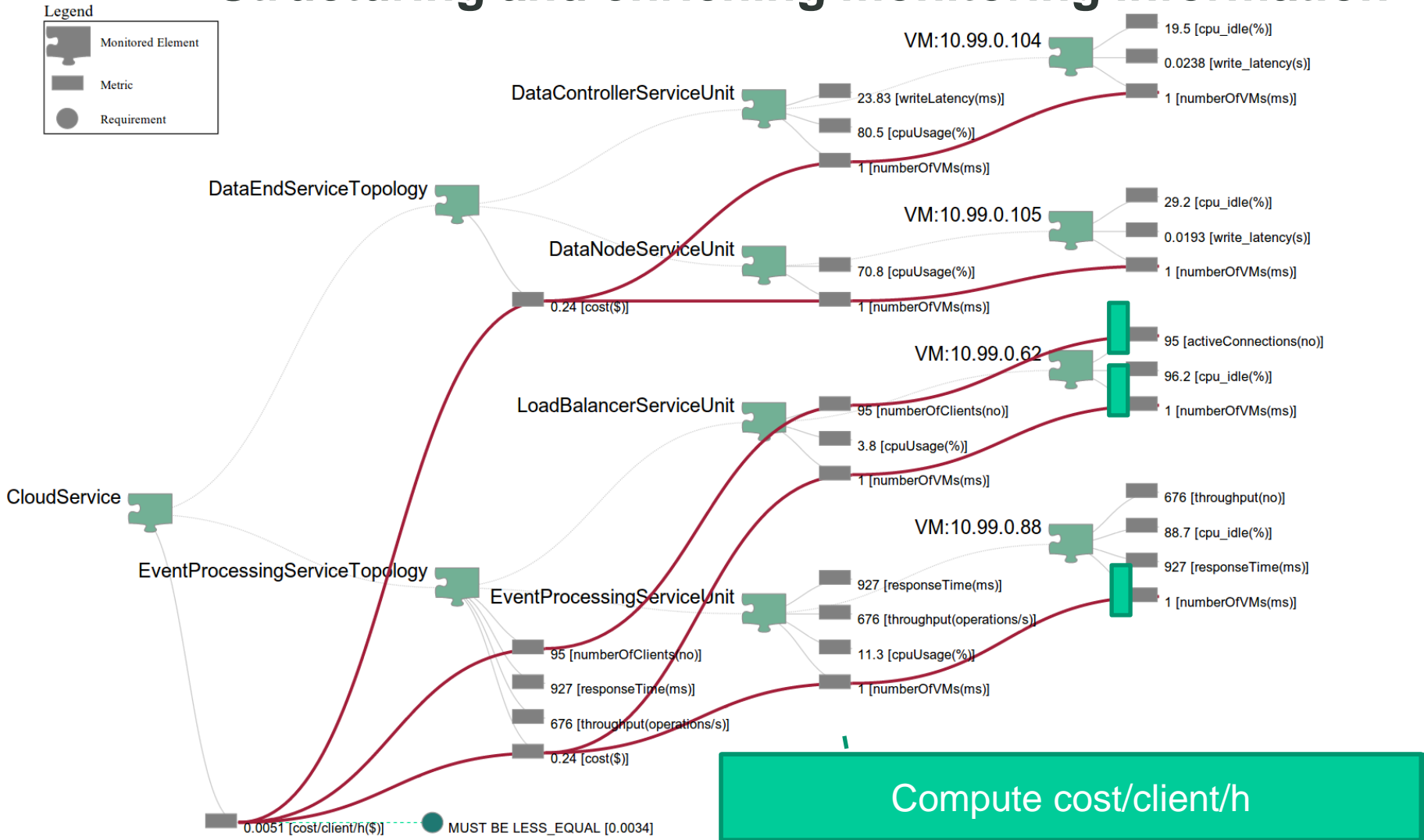
## Structuring monitoring information



Impose service structure over collected monitoring data

# Monitoring elastic cloud services

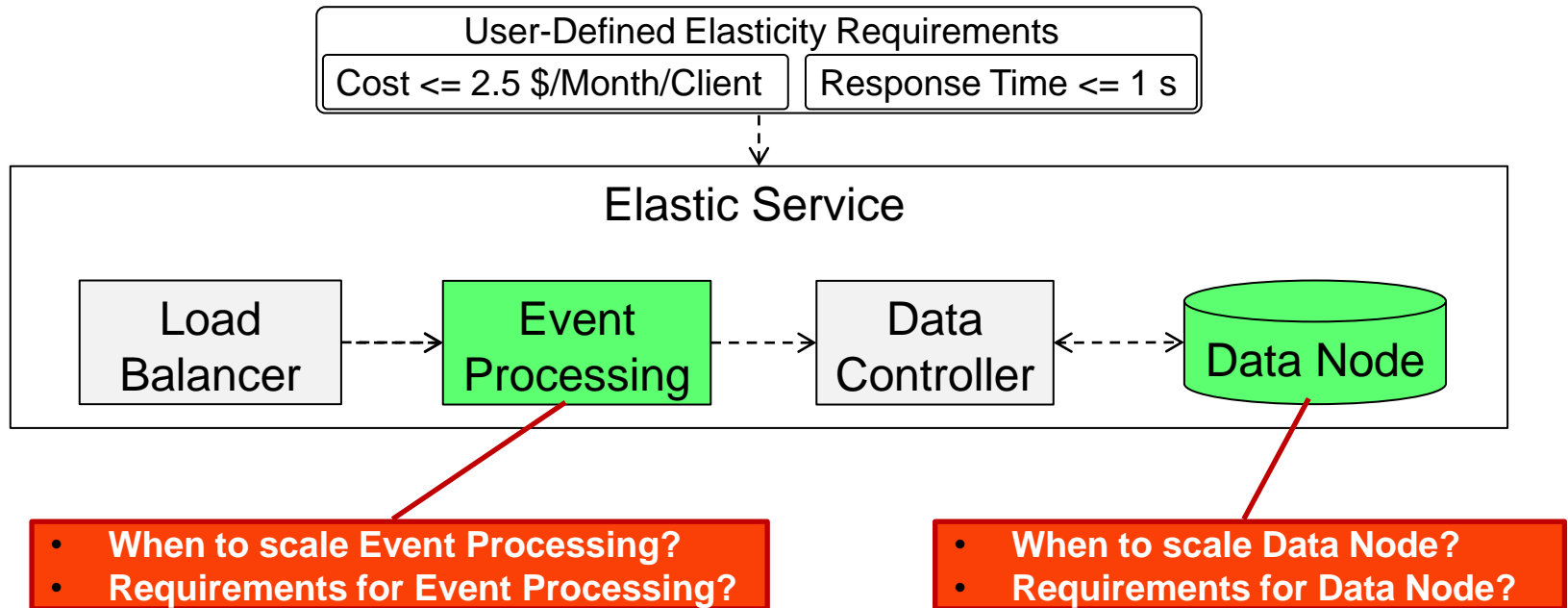
## Structuring and enriching monitoring information



- Analyzing Elasticity of cloud offered services
  - Quantifying elasticity of cloud offered services
- Monitoring elastic cloud services
- **Analyzing Elasticity of cloud services**
  - Elasticity Space, Boundaries and Pathway
  - Elasticity Dependencies

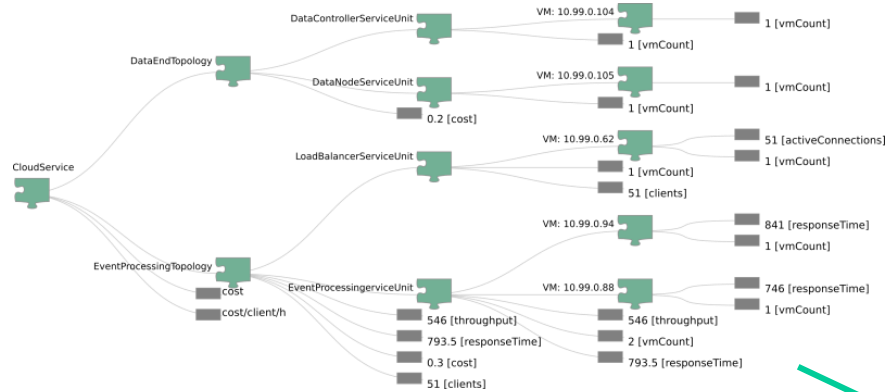
# Analyzing elasticity of cloud services

## Motivation



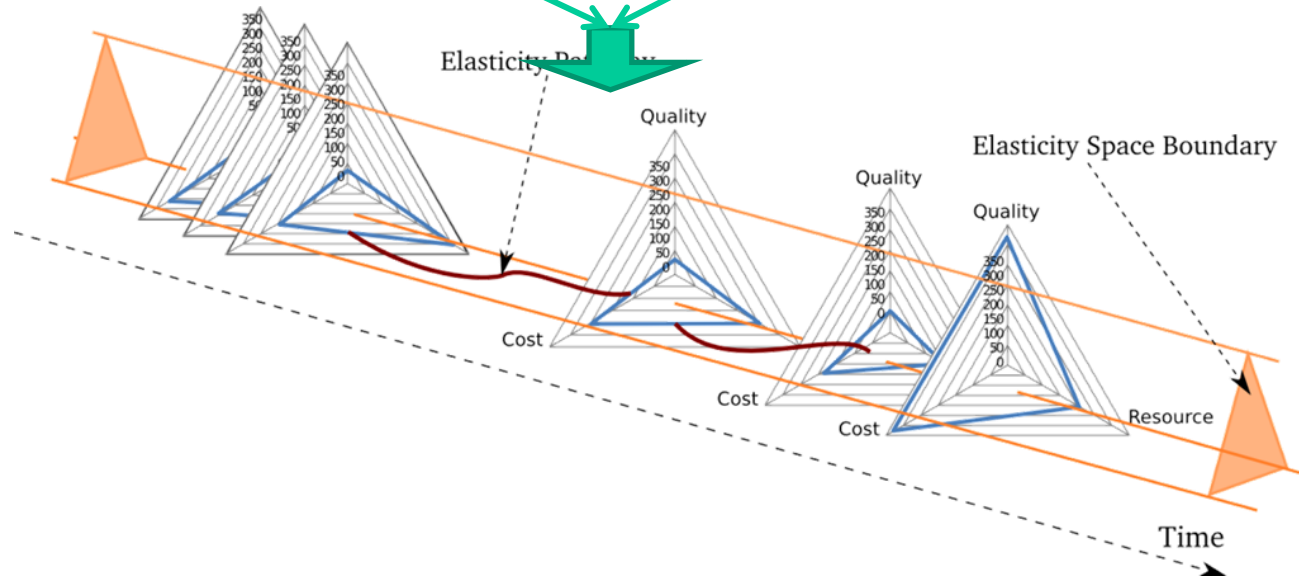
# Analyzing elasticity of cloud services

## Elasticity Space, Boundaries and Pathway



User-Defined Elasticity Requirements

Cost  $\leq$  2.5 \$/Month/Client      Response Time  $\leq$  1 s



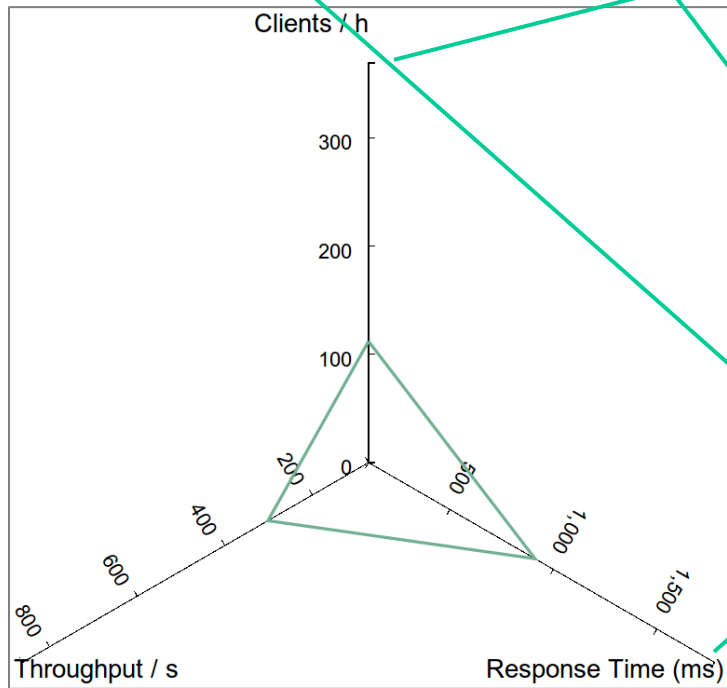


# Analyzing elasticity of cloud services

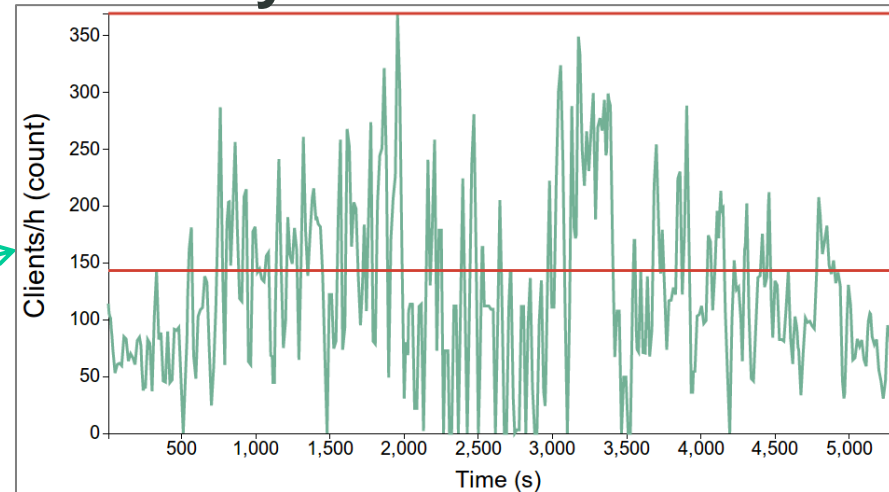
## Elasticity Space and Boundary

User-Defined Elasticity Requirements

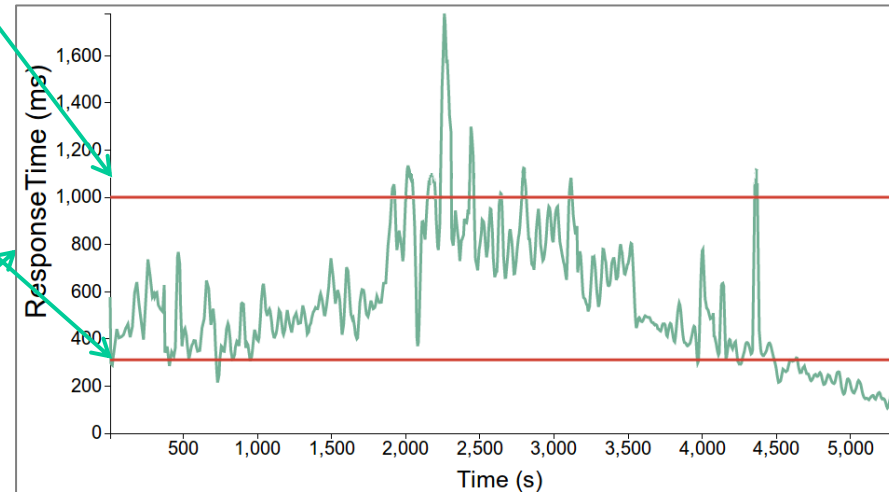
Cost  $\leq 2.5$  \$/Month/Client      Response Time  $\leq 1$  s



Elasticity Space Snapshot



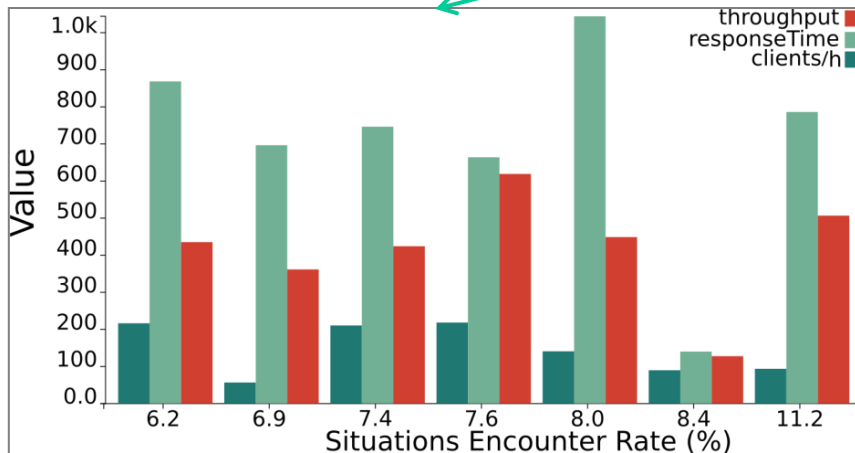
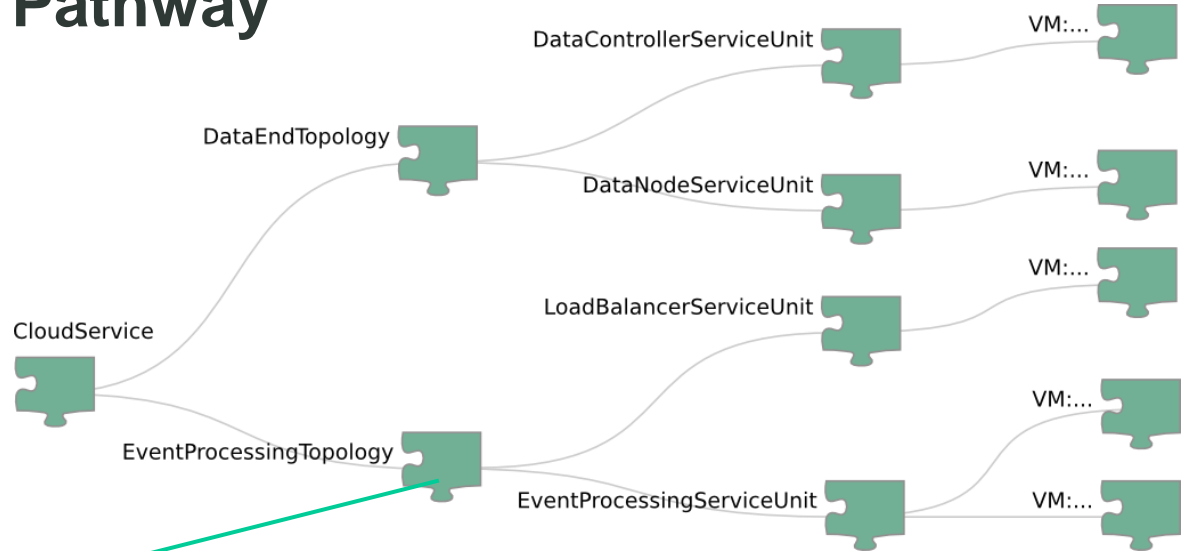
Elasticity Space "Clients/h" Dimension



Elasticity Space "Response Time" Dimension

# Analyzing elasticity of cloud services

## Elasticity Pathway



Event Processing service unit Elasticity Pathway

- Analyzing Elasticity of cloud offered services
  - Quantifying elasticity of cloud offered services
- Monitoring elastic cloud services
- **Analyzing Elasticity of cloud services**
  - Elasticity Space, Boundaries and Pathway
  - Elasticity Dependencies

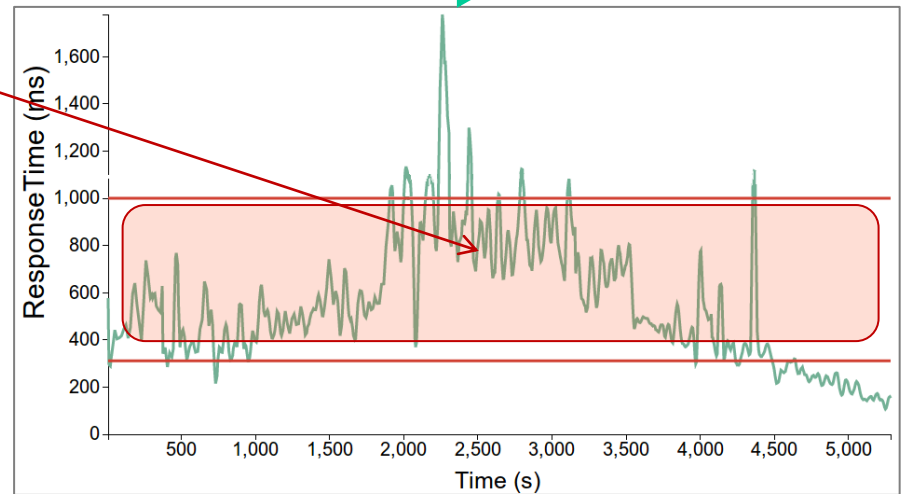
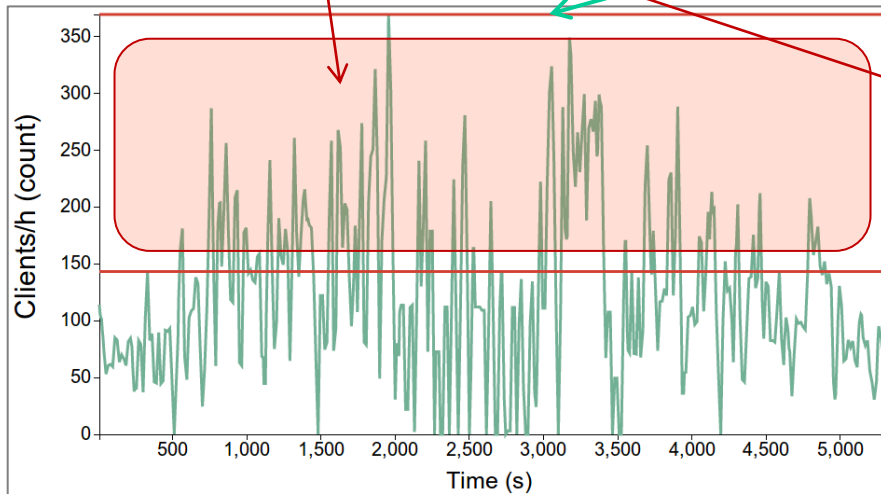
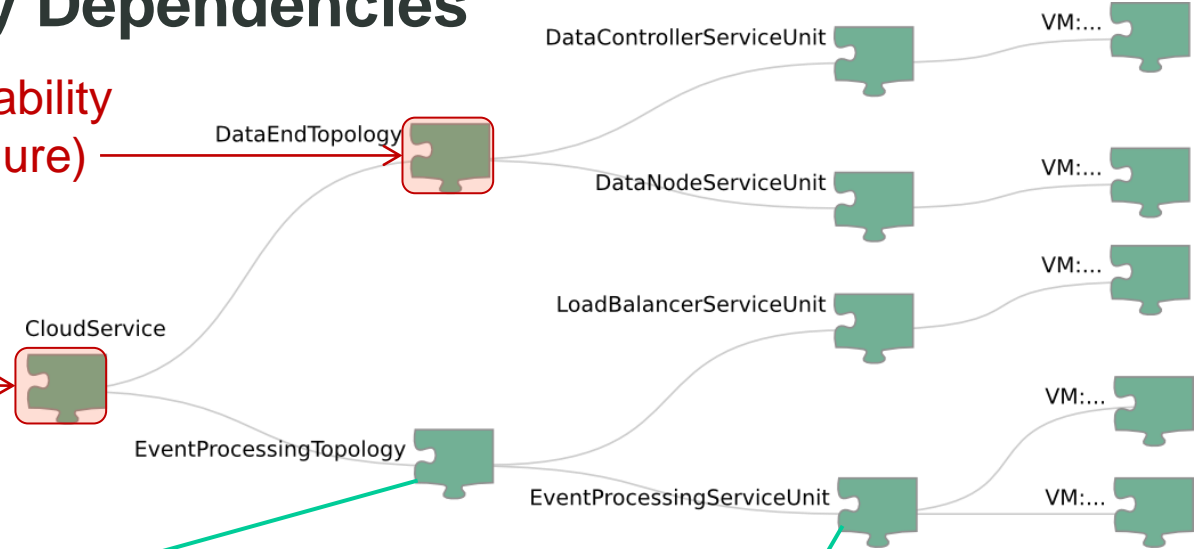
# Analyzing elasticity of cloud services

## Elasticity Dependencies

Enforce Elasticity Capability  
(scale in/out, reconfigure)

Change in Load

Effect ?



# Conclusions

- Concepts
  - Elasticity Space, Boundary, Pathway
  - Elasticity Dependencies
  - Elasticity Capabilities
- Mechanisms
  - Quantifying elasticity of cloud offered services
  - Constructing cross-layer monitoring snapshots
  - Analyzing elasticity
- Framework/Tools
  - QUELLE: a Framework for Accelerating the Development of Elastic Systems (<http://tuwiendsg.github.io/QUELLE>)
  - MELA: Monitoring and analyzing elasticity of cloud services (<http://tuwiendsg.github.io/MELA/>)

Distributed Systems Group (<http://dsg.tuwien.ac.at/>)

Vienna University of Technology (<http://www.tuwien.ac.at/>)

Work partially supported by the **European Commission** in terms of the **CELAR FP7** project (<http://www.celarcloud.eu/>)

