

Mapping of Service Level Objectives to Performance Queries

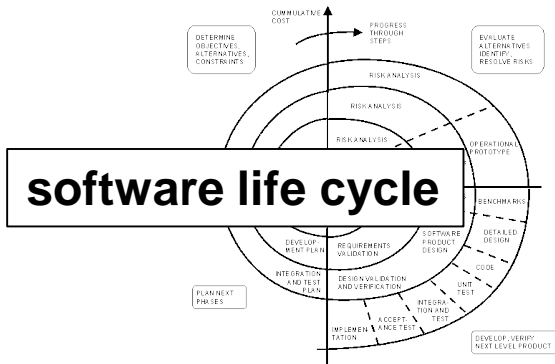
Jürgen Walter^{*}, Dušan Okanovic^{**}, and Samuel Kounev^{*}

* University of Würzburg – Chair of Software-Engineering

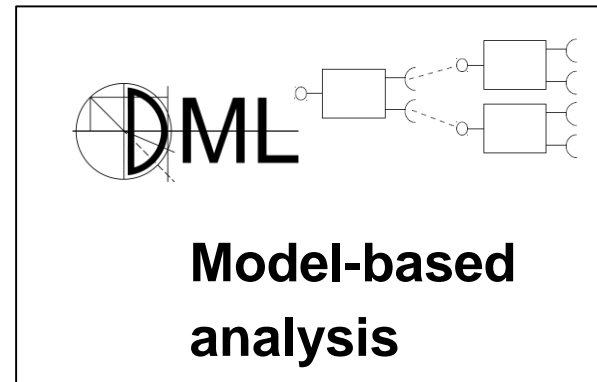
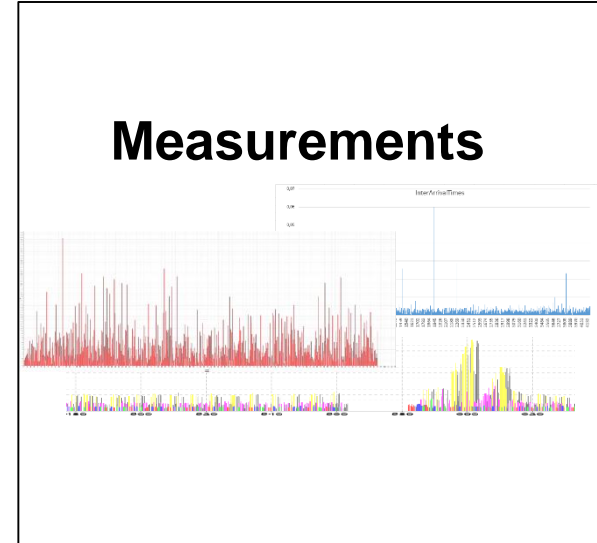
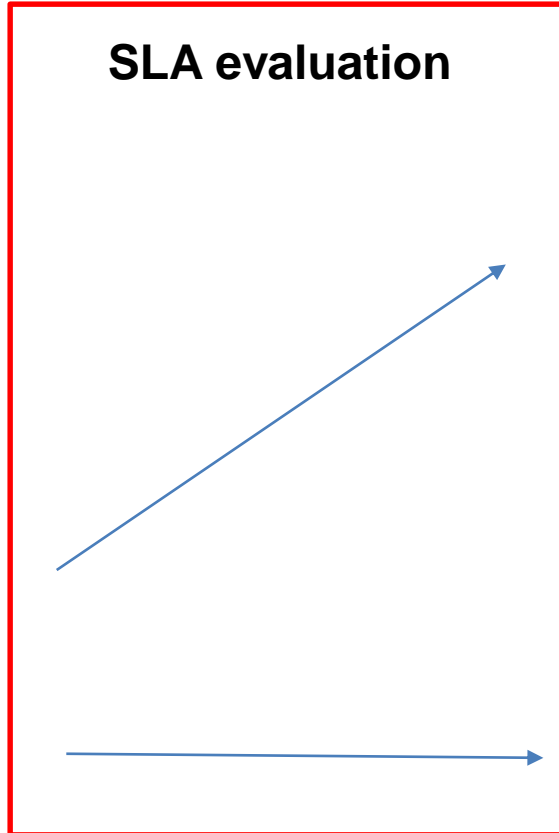
** University of Stuttgart – Institute of Software Technology

April 22, 2017

WOSP-C 2017 L'Aquila, Italy



Performance Engineer



Technology-agnostic

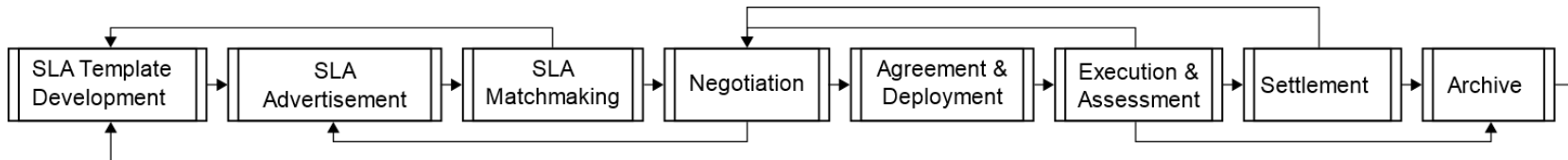


Technology-specific

- As a user, I want to evaluate my SLAs with different SPE methods in different stages of the software life cycle.
- As a user, I want the approach to require low effort.
- As a user, I want to use SLAs for different use cases.



Performance Engineer



SLA life cycle and process management

- Kritikos et al. (2013)
- Comuzzi et al. (2009)

No solution how to integrate measurement and model-based analysis

Languages

- WSLA, SLAng, GXLA, SLA*

XML based

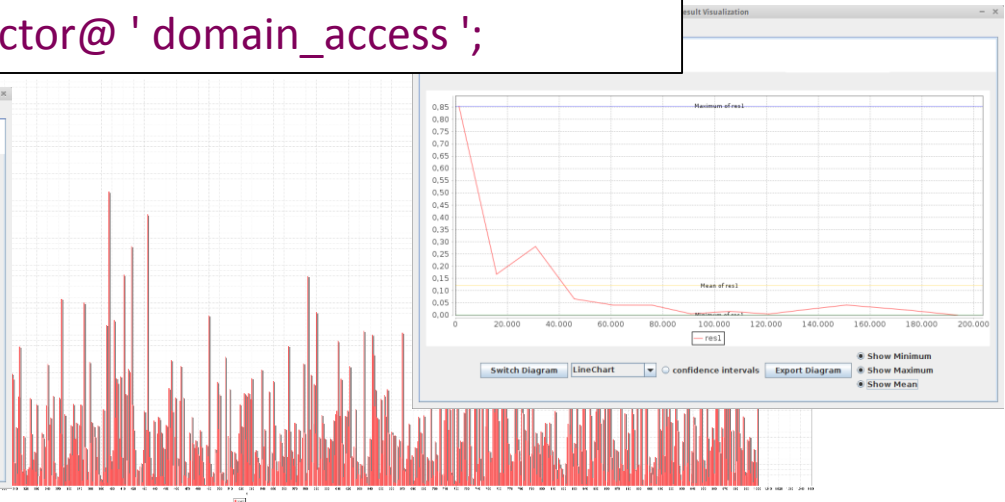
SLA assessment

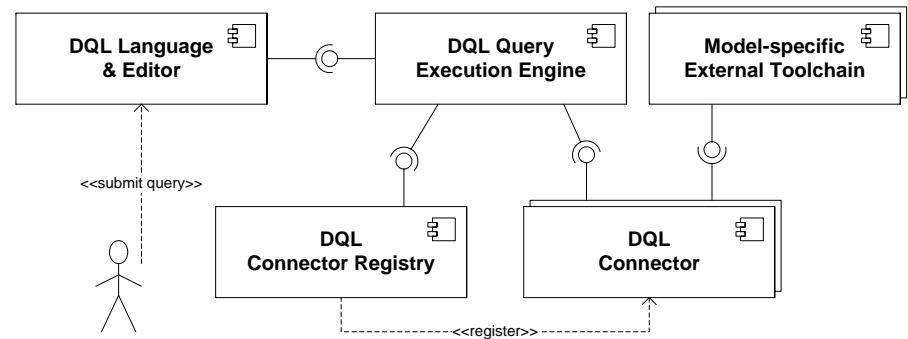
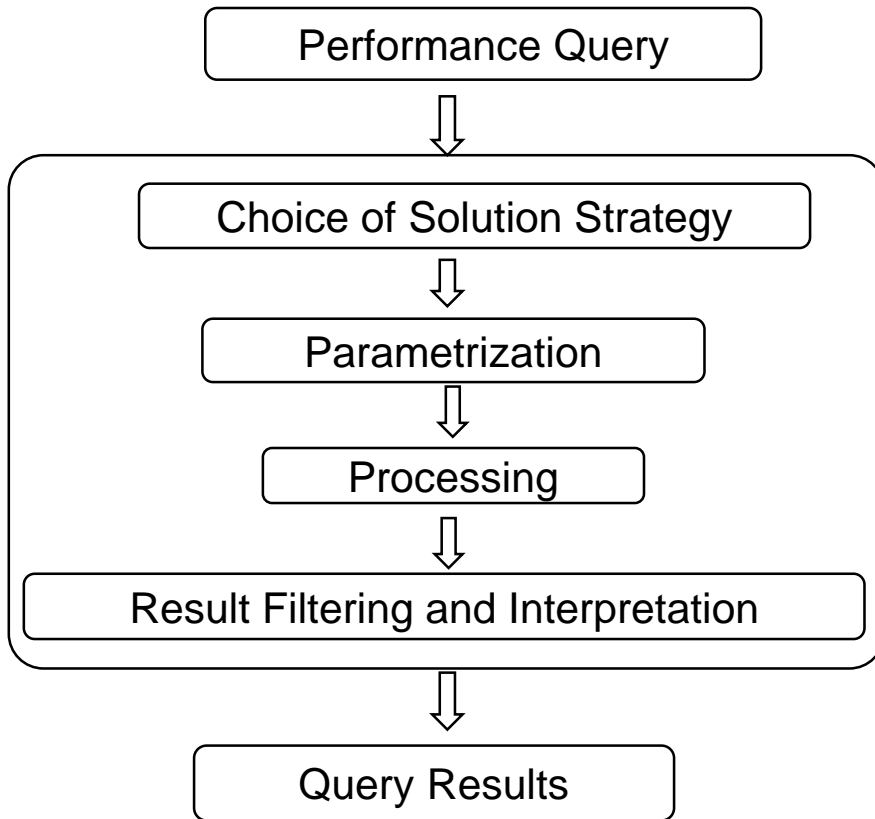
- Measurement-based
- Model-based Klatt et. al. (2011)

No unified approach

- Performance query language
- Measurement- and model-based analysis

```
SELECT service1 .responseTime  
FOR SERVICE " service1Identifier " AS service1  
CONSTRAINED AS fast  
USING connector@ ' domain_access ';
```

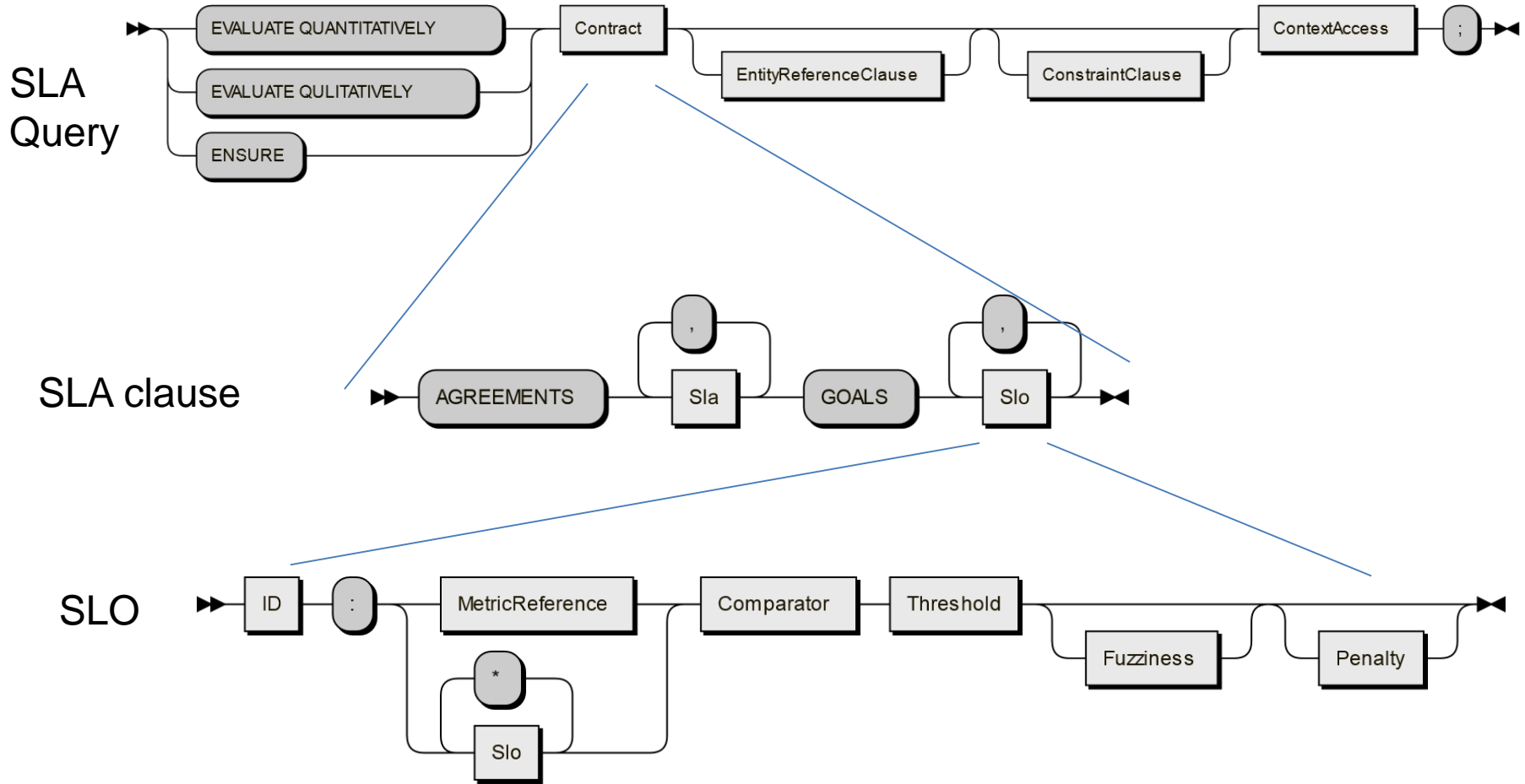




- Evaluate SLAs by mapping SLOs to the existing performance metrics query interface
- Integrate SLA evaluation into the existing DQL language and architecture
- Reuse connectors to external tool chains (measurement, simulation or analytical solvers) to evaluate SLAs

- DQL language extension to formulate SLAs
- DQL framework extension to process SLAs
- Mapping of SLOs to performance metric queries

SLA Language Extension



EVALUATE QUANTITATIVELY
AGREEMENTS

sla1 CONTAINS (slo1 , slo2)

GOALS

slo1 : service1 . availability > 90%

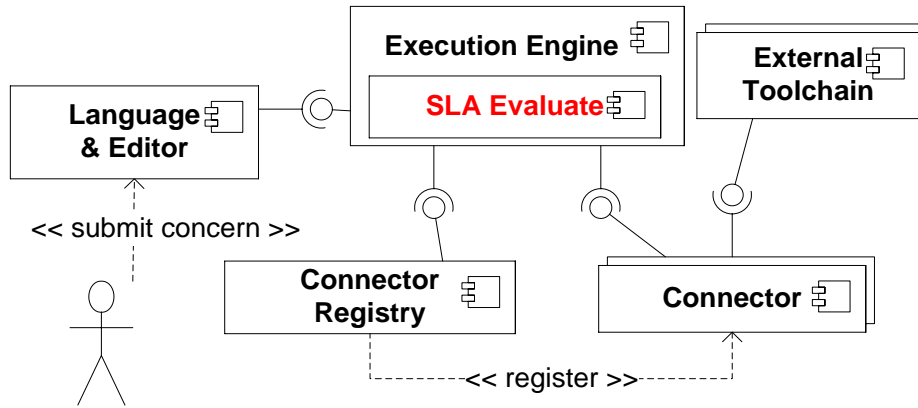
slo2 : service1 . responseTime < 0.2ms ,

FUZZINESS 0.1 PENALTY 6.0 EUR PER 3 VIOLATIONS

FOR SERVICE " service1Identifier " AS service1

CONSTRAINED AS fast

USING connector@'domain_access ';

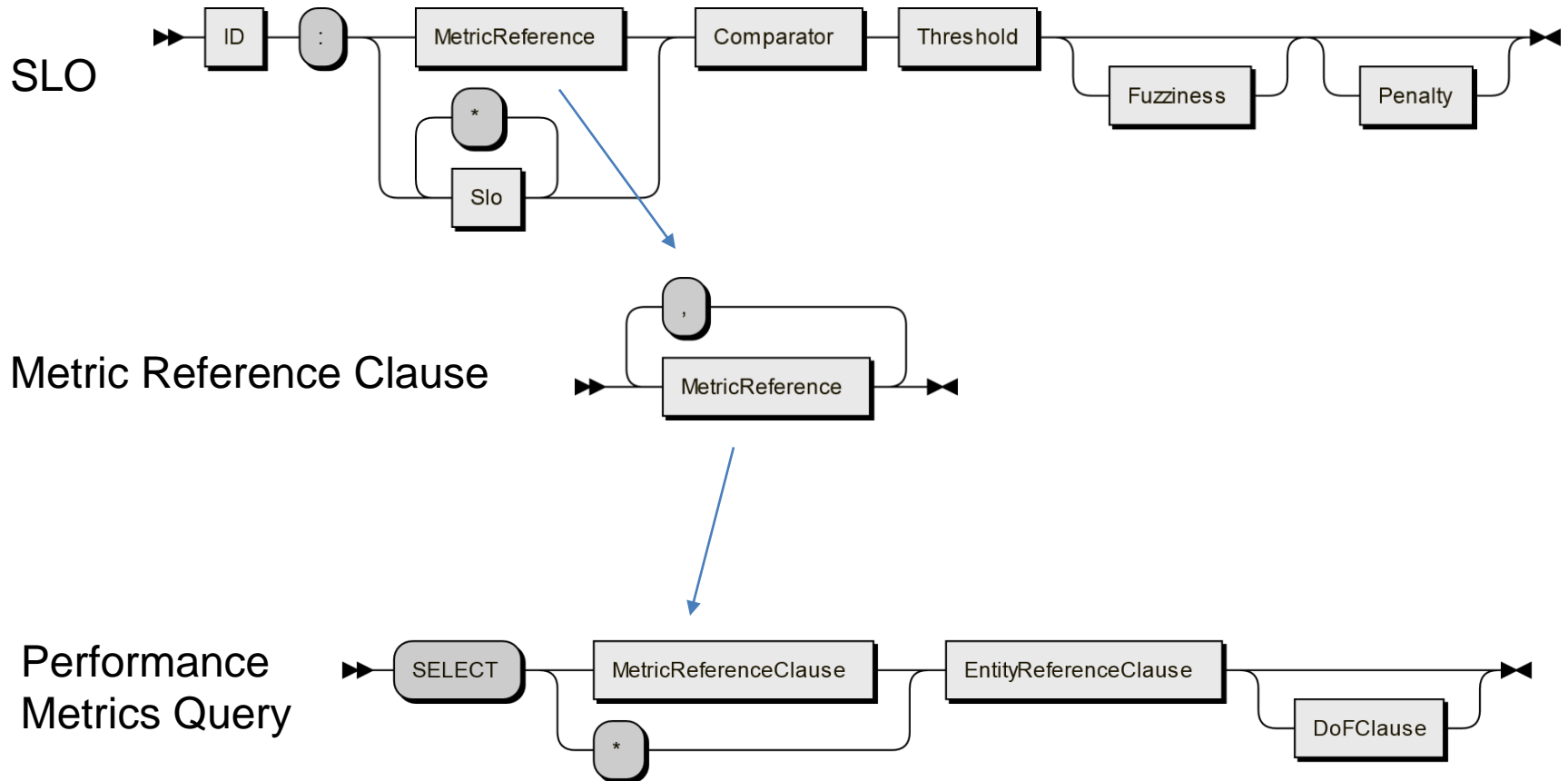


- Evaluation of samples and aggregated values
- Calculation of violations and penalties
- Procedures can be applied for all connectors

```

1: function EVALUATESLAS(List<SLA> slas, boolean
   isQualitativeEvaluation)
2:   for all sla in slas do
3:     for all slo in sla do
4:       if slo.isCompliant == null then
5:         slo.metric ← run query (slo)
6:       end if
7:       if evaluateSLO(slo, metric) then
8:         slo.isCompliant ← false
9:         if isQualitativeEvaluation then
10:          return false
11:        else
12:          sla.penalty += slo.penalty
13:        end if
14:      else
15:        slo.isCompliant ← true
16:      end if
17:    end for
18:  end for
19:  return true
20: end function
  
```

Mapping of SLOs to queries



EVALUATE QUANTITATIVELY AGREEMENTS

sla1 CONTAINS (slo1 , slo2 *2, slo3)

GOALS

slo1 : service1 . availability > 90%

slo2 : service1 . responseTime < 0.2ms

FUZZINESS 0.1 PENALTY 6.0 EUR PER 3 VIOLATIONS

slo3 : (slo1 * 3, slo2) >= 97% PENALTY 7.0 EUR

FOR SERVICE " service1Identifier " AS service1

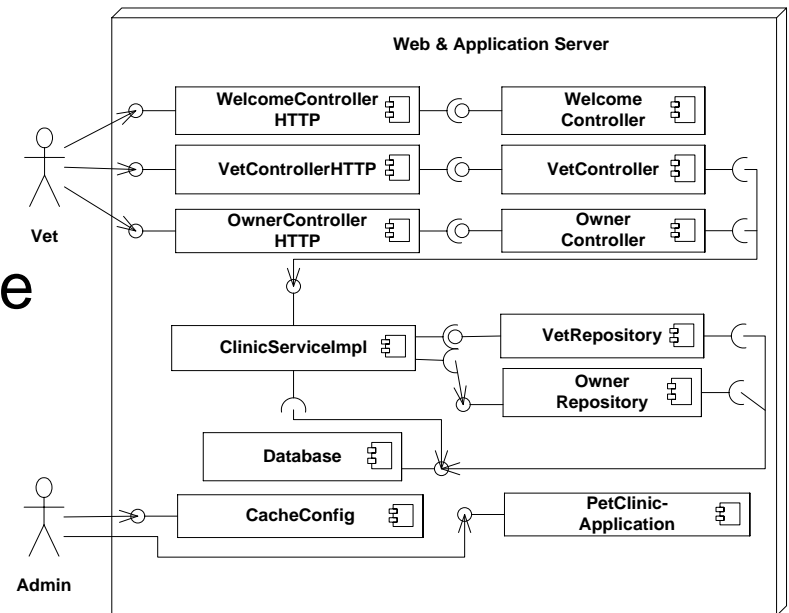
CONSTRAINED AS fast

USING connector@ ' domain_access ' ;

```
SELECT service1.availability
FOR SERVICE "service1Identifier" AS service1
CONSTRAINED AS fast
USING connector@'domain_access' ;
```

```
SELECT service1.responseTime
FOR SERVICE "service1Identifier" AS service1
CONSTRAINED AS fast
USING connector@'domain_access' ;
```

- PetClinic application, deployed on a VM using 42 cores
- Measurements using Kieker analysis framework
- Model-based analysis using Descartes Modeling Language
- Automated performance model learning using PMX



EVALUATE QUANTITATIVELY AGREEMENTS

sla CONTAINS (slo1 ,slo2 ,slo3 ,slo4 , slo5)

GOALS

slo1 : welcomeGET . responseTime.mean <1.3 ms , PENALTY 1.0 EUR

slo2 : showVetListGET . responseTime.mean <1.7 ms , PENALTY 1.0 EUR

slo3 : initFindFormGET . responseTime.mean <1.3 ms , PENALTY 1.0 EUR

slo4 : processFindFormGET . responseTime.mean <4ms , PENALTY 1.0 EUR

slo5 : showOwnerGET . responseTime .mean <2.6 ms , PENALTY 1.0 EUR

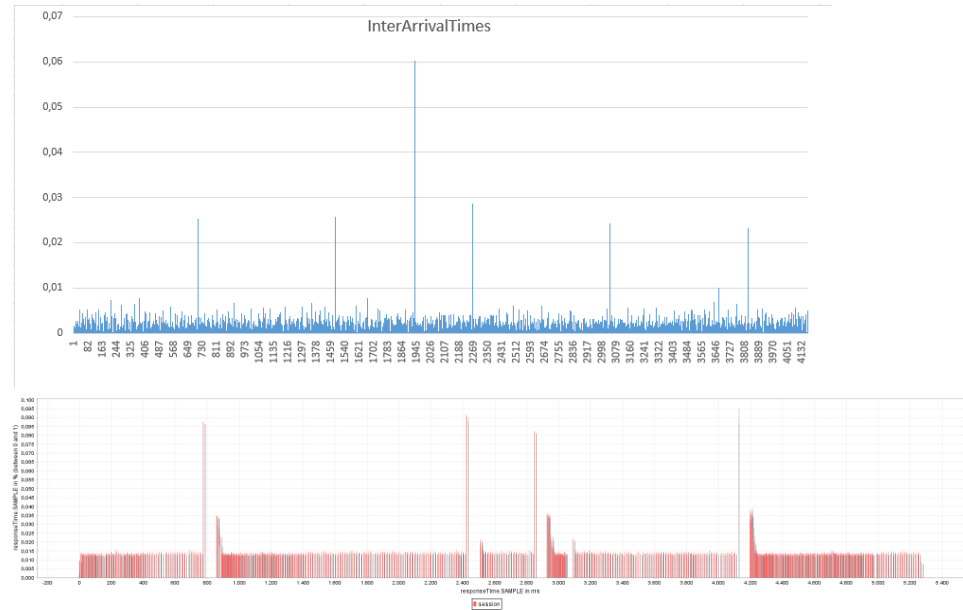
USING dml@'petclinic.properties'

Entity	Evaluation	Analysis Method			
		measurements		model-based	
customers		732	1300	732	1300
slo1	Violating Operations	false	true	false	true
slo2		false	true	false	true
slo3		false	true	false	true
slo4		false	true	false	true
slo5		false	true	false	true
sla1	Conformance	true	false	true	false
	Penalty	0	5	0	5

Table 1: SLA Evaluation Results.

Methodology works ...

- Challenge: Models have to depict borderline scenarios accurately
- Prediction of resource utilization and response time mean accurate, but ...
- Workload driver fits configured arrival rate but creates spikes ...



Not discussed in the paper

- We provide an interface between an SLA language and performance evaluation mechanisms
- Our approach reduces the effort required for applying different performance evaluation methods in different SLA life-cycle stages

- Integrate generic strategies for SLA ensurance into DQL framework
- Combine SLA evaluation with design space exploration and system variation
- Discuss/cope with different counting of violations
 - **Absolute Number** of violations. Penalties can be defined independent of an analysis interval, e.g., 1 Euro, per 5 violations
 - **Time Interval** Count violations within an interval, e.g., 1 Euro per 5 violations within 1 HOUR
 - **Proportional**, e.g., 1 Euro if 10% of all operations violate the threshold
- Automatically determine appropriate analysis run length

Thank You!

Jürgen Walter*, Dušan Okanovic**, and Samuel Kounev*

* University of Würzburg – Chair of Software-Engineering

** University of Stuttgart – Institute of Software Technology

April 22, 2017

WOSP-C 2017 L'Aquila, Italy