Towards Continuous Consistency Checking of DevOps Artefacts

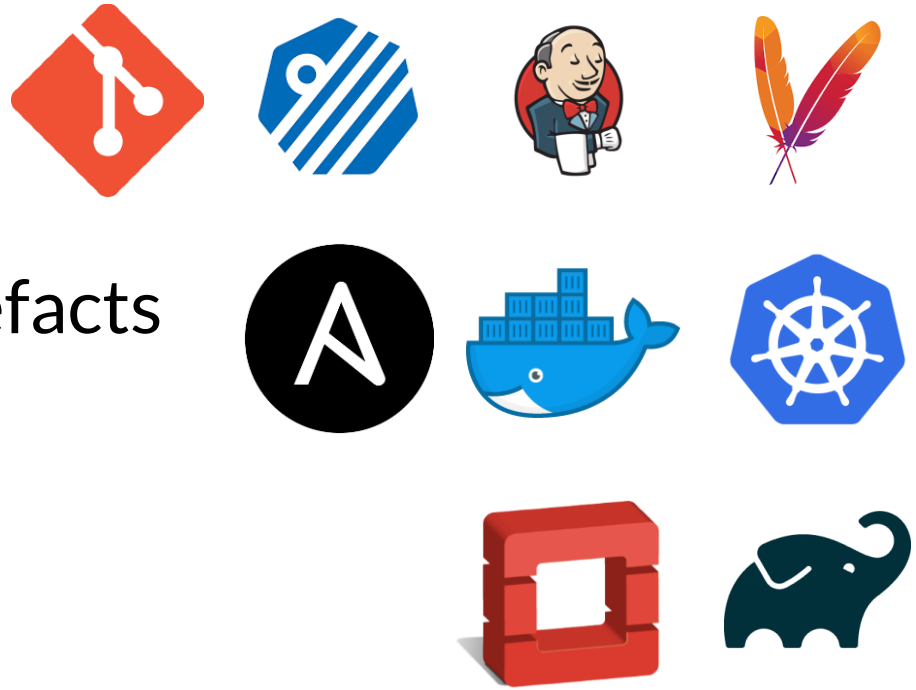
Alessandro Colantoni[1], **Benedek Horváth**[1,2], Ákos Horváth[2], Luca Berardinelli[1], Manuel Wimmer[1]

[1] Johannes Kepler University Linz, Linz, Austria
[2] IncQuery Labs cPlc, Budapest, Hungary

Contact: Benedek.Horvath@incquerylabs.com

Motivation

- More than 400 DevOps tools
- Complex configuration scripts, textual artefacts
- Semantically related configuration files
- Manual effort to check their consistency



Goal: Continuous consistency checking of DevOps artefacts

Example: Keptn

- Open-source cloud-native application lifecycle orchestration project
- Orchestrate continuous delivery (CD) pipelines with quality gate criteria
- Keptn stores configuration artefacts on Git for version control

Service Level Indicator (SLI)

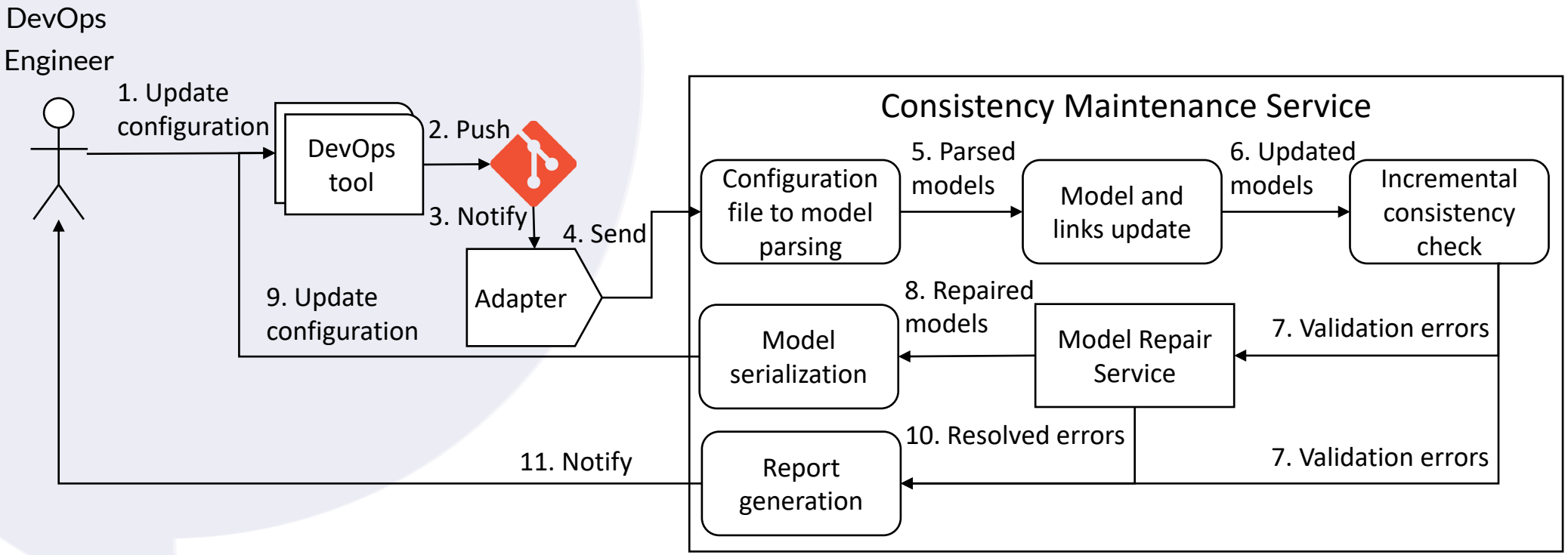
```
{
   "spec_version": "1.0",
   "indicators": {
      "response_time_p95": "response.time:percentile(95)?
      "response_time_p90": "response.time:percentile(90)?
   }
}
```

Service Level Objective (SLO)

```
{
   "spec_version": "1.0",
   "objectives": [
      {
         "sli": "response_time_p95",
         "pass": [
            {
               "criteria": [ "<=+10%" ]
            }
         ]
      },
      {
         "sli": "response_time_p90",
         "pass": [
            {
               "criteria": [ "<600" ]
            }
         ]
      }
   ],
```

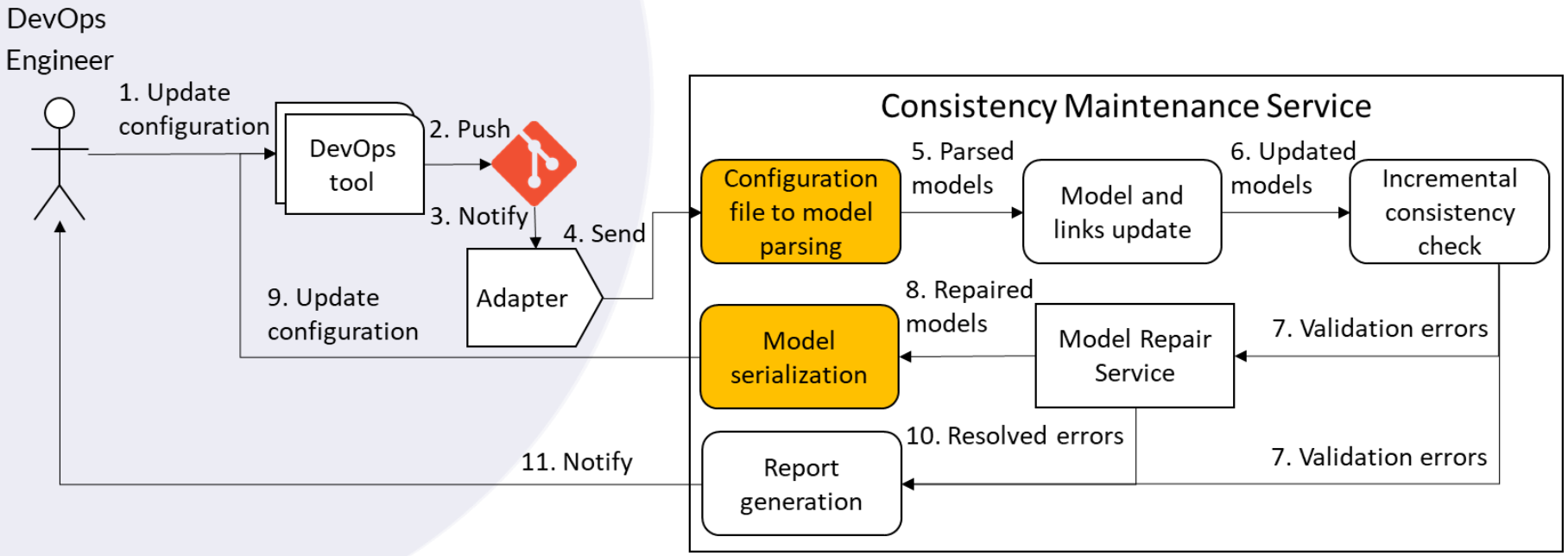
Goal: Ensure the consistency of the indicator names

Proposed Solution



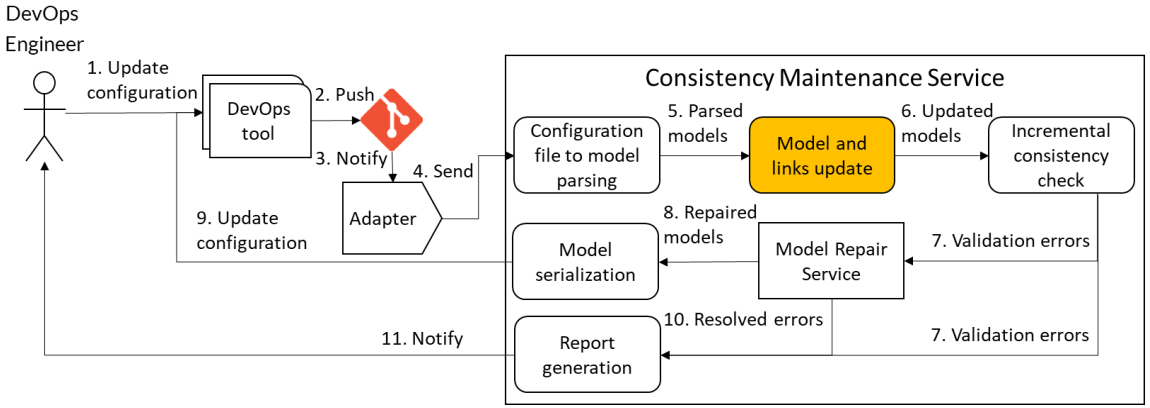
How to check the consistency of artefacts that belong to different languages?

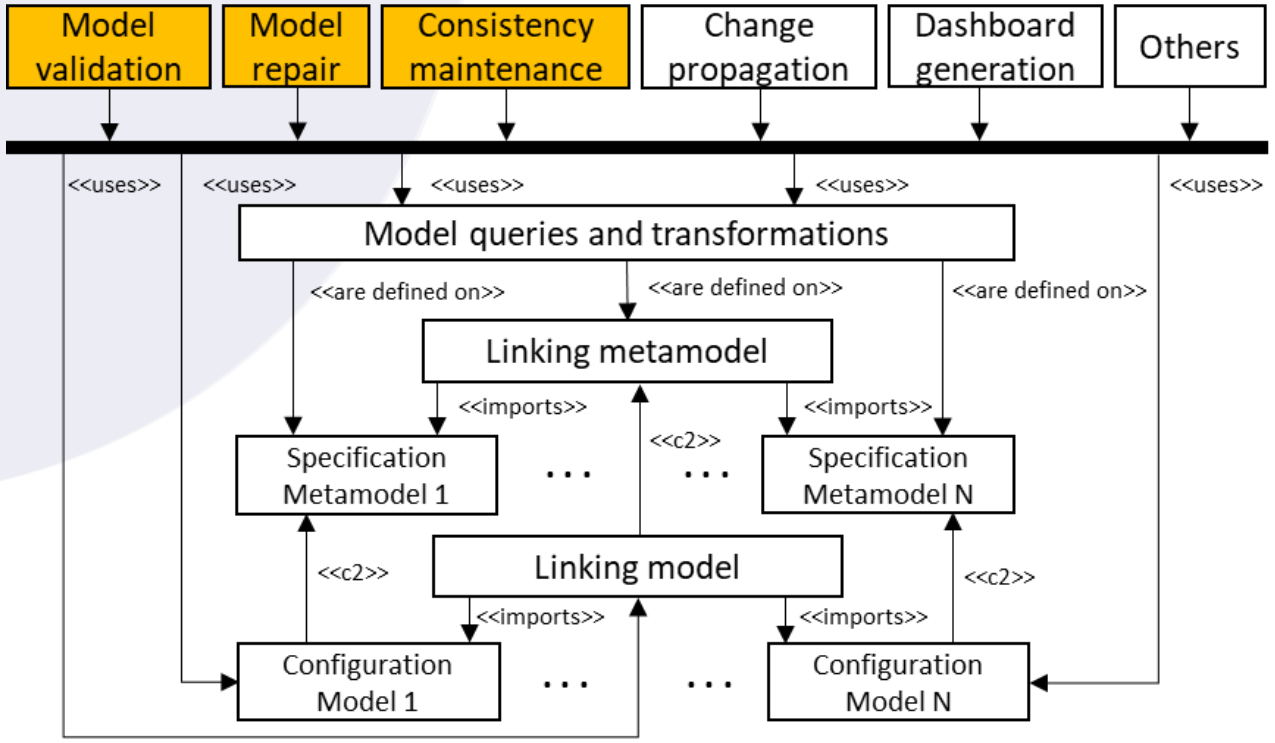
JSON-EMF Bridge



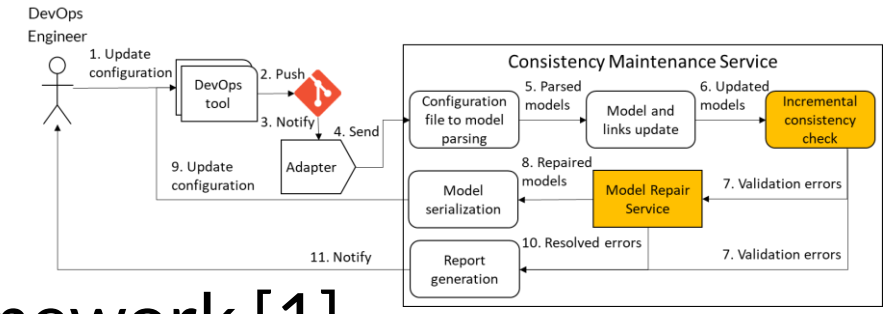
- Alessandro Colantoni, Antonio Garmendia, Luca Berardinelli, Manuel Wimmer, and Johannes Bräuer: *Leveraging Model-Driven Technologies for JSON Artefacts: The Shipyard Case Study*
- **Detailed talk on *Thursday (14.10)* from *5 PM (GMT+2)***

MDE Use Cases





Viatra



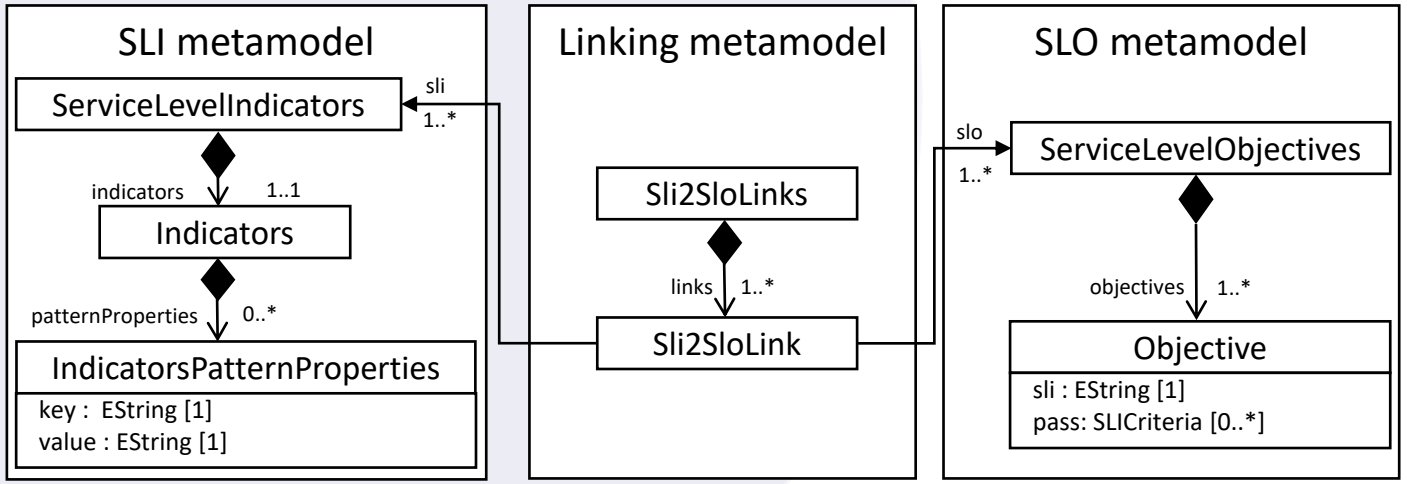
- Model query, transformation, validation framework [1]
- Validation rules as graph patterns
- Model repairs as reactive model transformations

```
@Constraint(
  message = "Objective's SLI field must refer to an indicator with
the same key.",
  severity = "error"
)
pattern objRefersToWrongProperty(obj: Objective, name: java String,
   property: IndicatorsPatternProperties){
  Sli2SloLink.sli(link, sliRoot);
  Sli2SloLink.slo(link, sloRoot);

  ServiceLevelObjectives.objectives(sloRoot, obj);
  Objective.sli(obj, name);

  ServiceLevelIndicators.indicators(sliRoot, indicator);
  Indicators.patternProperties(indicator, property);
  neg Indicators.patternProperties.key(indicator, name);
}
```

Case Study

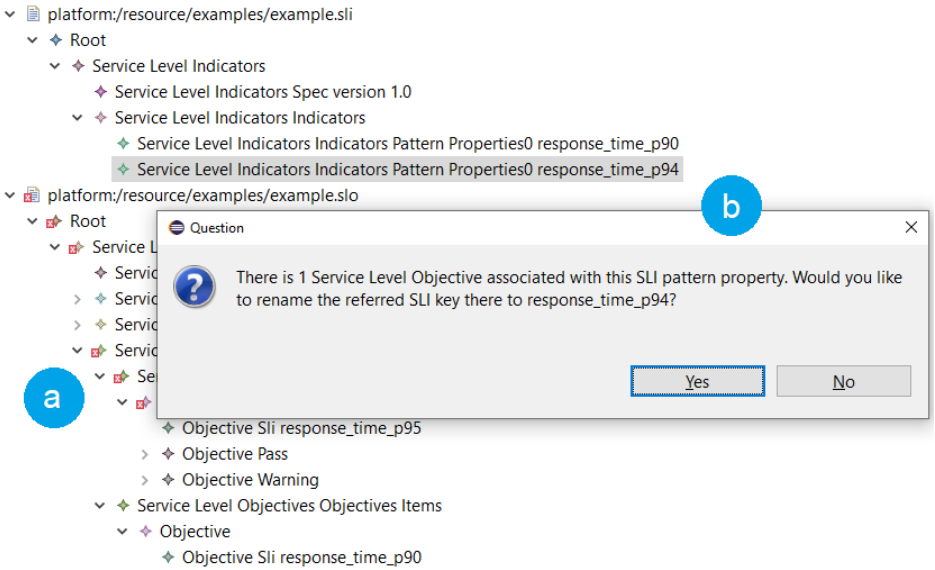



```
{ ...
  "indicators" : {
    "response_time_p95": "builtin:service.response.time:merge(0):percentile(95....",
    "response_time_p90": "builtin:service.response.time:merge(0):percentile(90...."
  }
  ...
}
```

example.sli

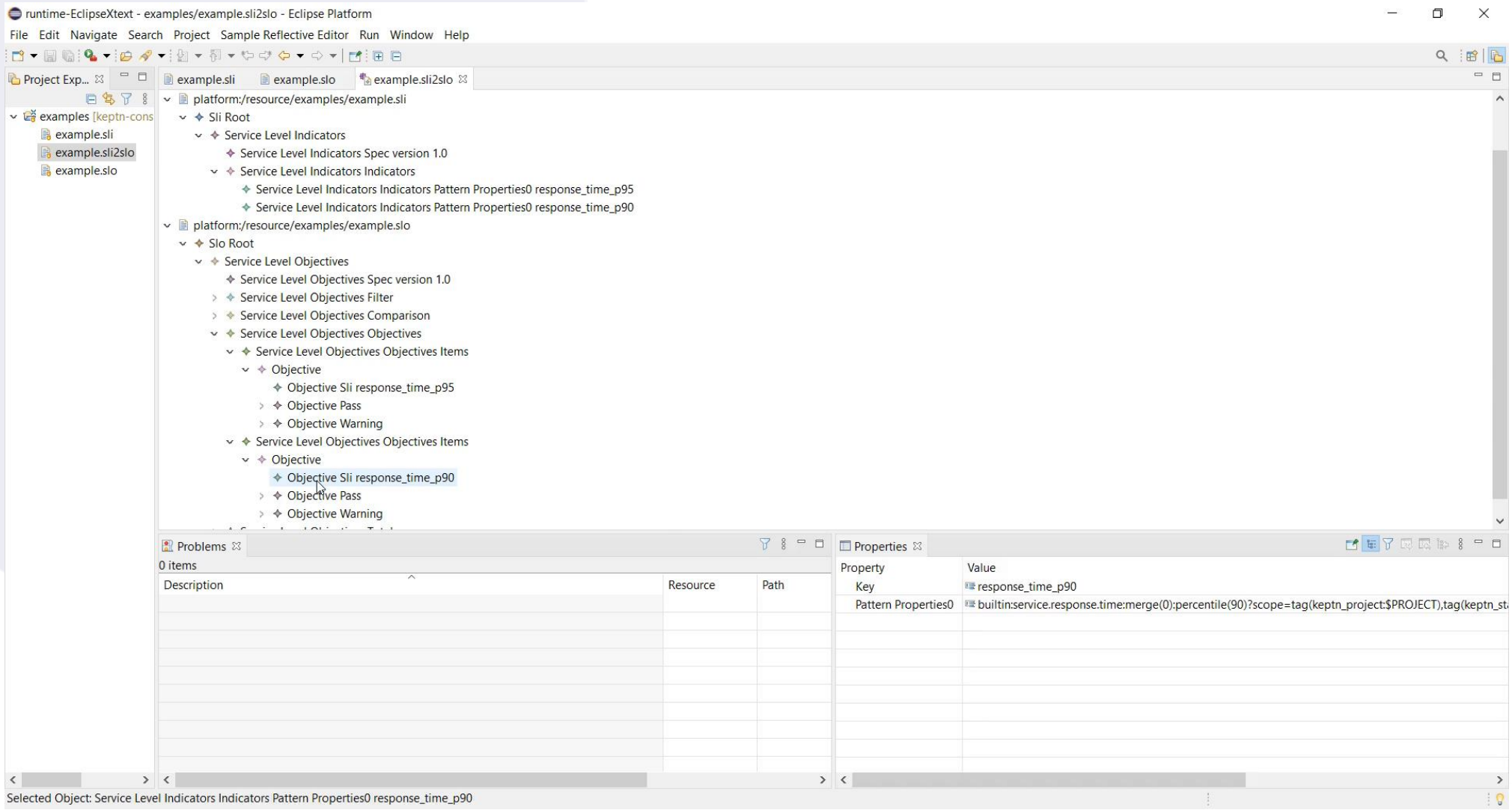
```
{ ...
  "objectives" : {
    "sli" : "response_time_p95",
    "pass" : [
      {"criteria": ["<=+10%"]}
    ]
  }
  ...
}
```

example.slo



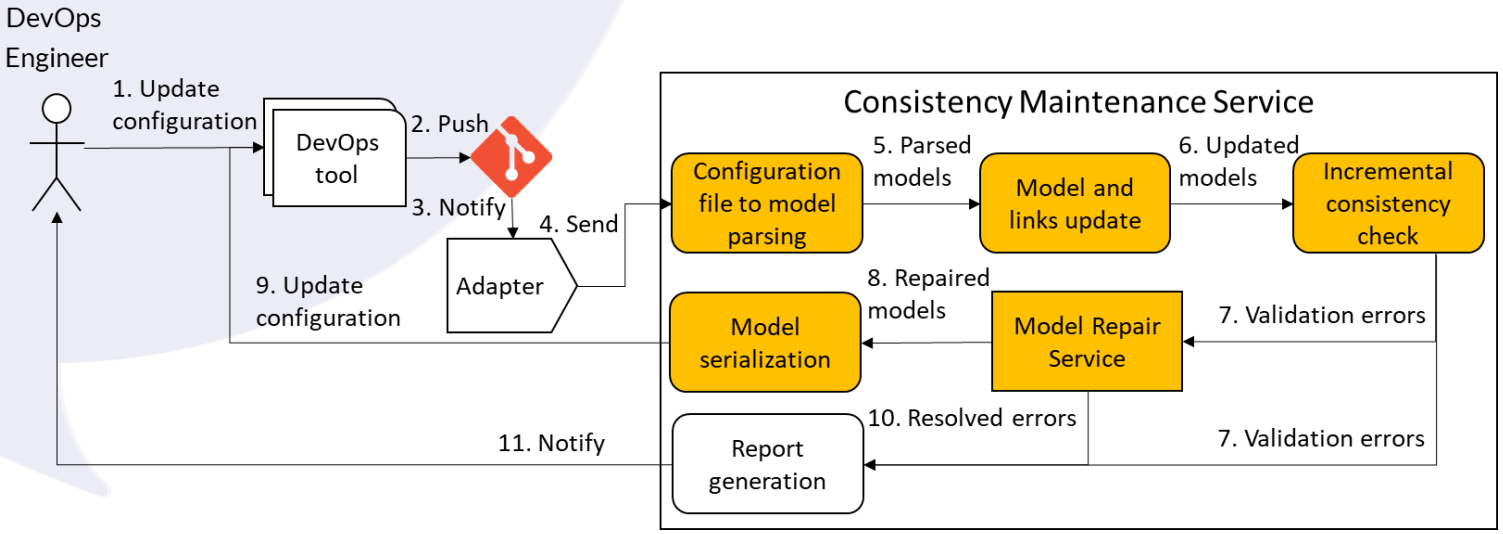
https://github.com/lowcomote/keptn-consistency-maintenance

Case Study


runtime-EclipseXtext - examples/example.sli2slo - Eclipse Platform
File Edit Navigate Search Project Sample Reflective Editor Run Window Help
Project Exp...
examples [keptn-cons
example.sli
example.sli2slo
example.slo
example.sli
example.slo
example.sli2slo
platform:/resource/examples/example.sli
Sli Root
Service Level Indicators
Service Level Indicators Spec version 1.0
Service Level Indicators Indicators
Service Level Indicators Indicators Pattern Properties0 response_time_p95
Service Level Indicators Indicators Pattern Properties0 response_time_p90
platform:/resource/examples/example.slo
Slo Root
Service Level Objectives
Service Level Objectives Spec version 1.0
Service Level Objectives Filter
Service Level Objectives Comparison
Service Level Objectives Objectives
Service Level Objectives Objectives Items
Objective
Objective Sli response_time_p95
Objective Pass
Objective Warning
Service Level Objectives Objectives Items
Objective
Objective Sli response_time_p90
Objective Pass
Objective Warning
Problems
0 items
Description
Resource
Path
Properties
Property
Value
Key
response_time_p90
Pattern Properties0
builtin:service.response.time:merge(0):percentile(90)?scope=tag(keptn_project:$PROJECT),tag(keptn_st
Selected Object: Service Level Indicators Indicators Pattern Properties0 response_time_p90

Case Study Limitations

- Two artefacts of the same DevOps tool (Keptn: SLI, SLO)
- Linking model is created by hand
- The consistency management workflow is partially implemented



- Scalability evaluation is future work

Conclusions and Future Work

- Proposed GitOps workflow to check the artefacts' consistency
- Proof of concept implementation of the workflow
- Several MDE use cases for DevOps configuration artefacts

- Modeling:
  - GitOps adapter to reflect the Git structure as a megamodel
  - Query-driven soft-links to maintain the linking model [2]
- Research-oriented:
  - Incremental parsing to have better integrations with Viatra [3]

Acknowledgements

- This work was funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 813884,
- the AIDOaRt project ECSEL Joint Undertaking (JU) under grant agreement No. 101007350,
- by the Austrian Research Promotion Agency (FFG), program ICT of the Future, project number 867535,
- and contributed to the ITEA3 BUMBLE project (18006).
- The authors are grateful for the valuable feedback of the anonymous reviewers about the paper.

References

[1] Dániel Varró, Gábor Bergmann, Ábel Hegedüs, Ákos Horváth, István Ráth, Zoltán Ujhelyi: Road to a reactive and incremental model transformation platform: three generations of the VIATRA framework. Softw. Syst. Model. 15(3): 609-629 (2016)

[2] Ábel Hegedüs, Ákos Horváth, István Ráth, Rodrigo Rizzi Starr, Dániel Varró: Query-driven soft traceability links for models. Softw. Syst. Model. 15(3): 733-756 (2016)

[3] Thomas Goldschmidt, Steffen Becker, Axel Uhl: Classification of Concrete Textual Syntax Mapping Approaches. ECMDA-FA 2008: 169-184