



COMPARISON-BASED FEATURE LOCATION IN ARGOUML VARIANTS



**Gabriela K.
Michelon**

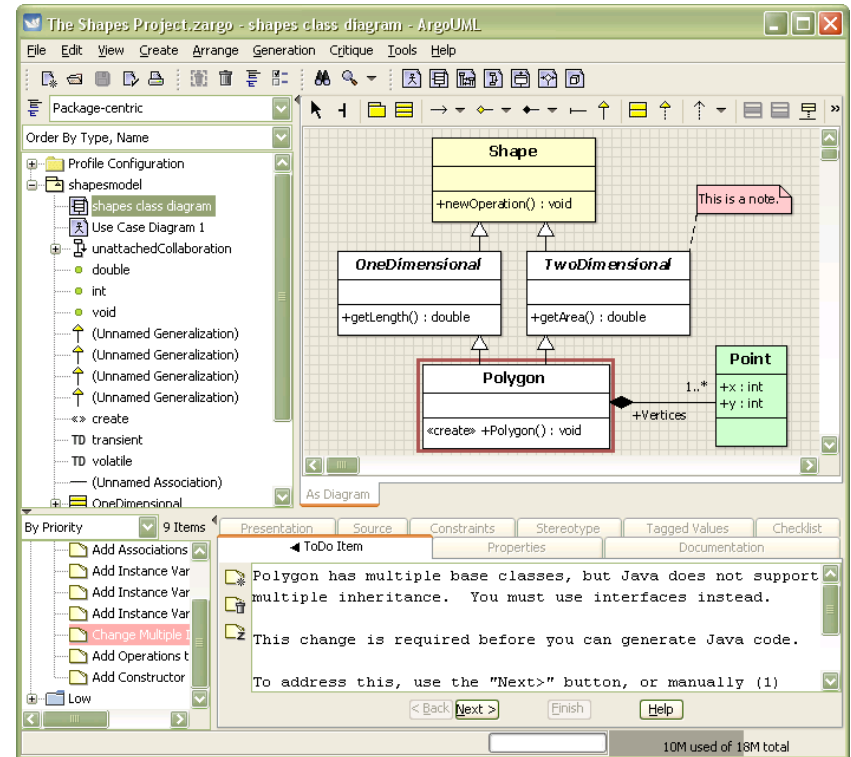
**Lukas
Linsbauer**

**Wesley K.
G. Assunção**

**Alexander
Egyed**

ARGOUML

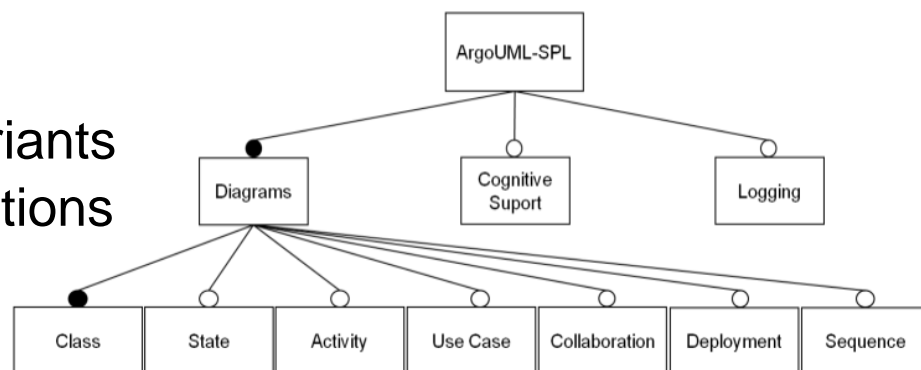
- 1999: initial release
- Java-based open source tool
- Supports UML 1.4 diagrams
- Case study to extract SPL
- 2011: ArgoUML SPL
 - Allows to derive variants from a set of optional features



<http://argouml.tigris.org/>

FEATURE LOCATION IN ARGOUML SPL

- Feature: software system or systems characteristic, quality or user-visible aspect
- Feature location: mappings/traces between features and their respective implementation
- Challenge³:
 - Eight features
 - 15 predefined scenarios
 - Ranging from one to 256 variants
 - Composed of feature interactions and feature negations
 - Granularity varies from Java Classes to statements inside methods



³Jabier Martinez, Nicolas Ordoñez, Xhevahire Tërnavá, Tewfik Ziadi, Jairo Aponte, Eduardo Figueiredo, and Marco Tulio Valente. Feature Location Benchmark with ArgoUML SPL. SPLC 2018.

ARGOUML SPL DATA SET

■ Scenarios

- 15 scenarios.
- Optional features: State Diagram, Activity Diagram, Use Case Diagram, Collaboration Diagram, Deployment Diagram, Sequence Diagram, Cognitive Support, and Logging.

Scenario	Size	Description
Original	1	Original ArgoUML variant containing all features.
Traditional	10	Variants with no, all, and combinations of 7 features.
PairWise	9	Set of variants that covers all pairwise feature combinations.
2-10 Random	2-10	Randomly selected subsets of variants.
50 Random	50	Randomly selected subset of variants.
100 Random	100	Randomly selected subset of variants.
All	256	All possible variants with 8 optional features.

ARGOUML SPL DATA SET

■ Ground Truth

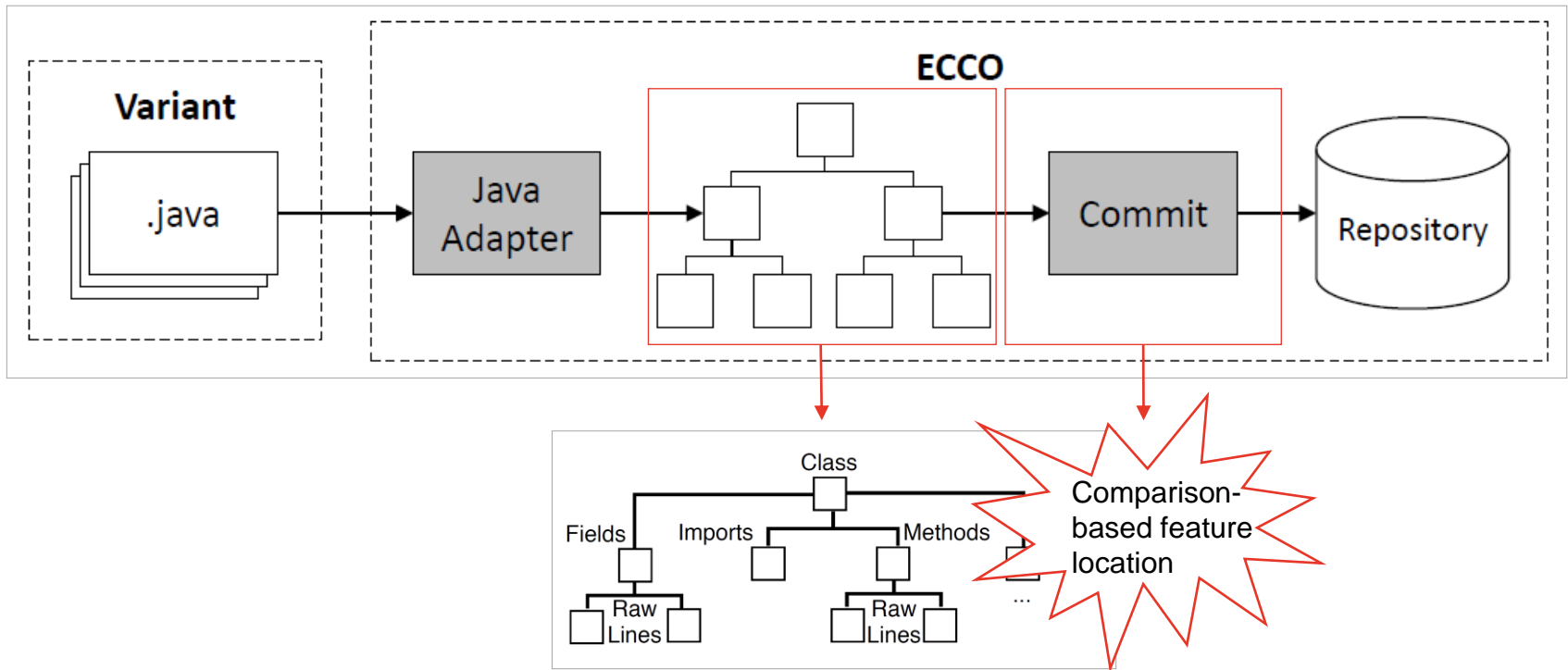
□ 24 traces

- One trace for each of the eight individual features
- Two traces with a single negative feature
- 13 traces with a conjunction of two features
- One trace with a conjunction of three features

A trace T is a pair (F, A) that maps a propositional logic formula F whose literals are features to a set of implementation artifacts A

FEATURE LOCATION TECHNIQUE

- ECCO tool^{1,2}
 - Applied a new Java Adapter



¹Stefan Fischer, Lukas Linsbauer, Roberto E. Lopez-Herrejon, and Alexander Egyed. The ECCO Tool: Extraction and Composition for Clone-and-Own. ICSE.2015.

²Stefan Fischer, Lukas Linsbauer, Roberto E. Lopez-Herrejon, and Alexander Egyed. Enhancing Clone-and-Own with Systematic Reuse for Developing Software Variants. ICSME 2014.

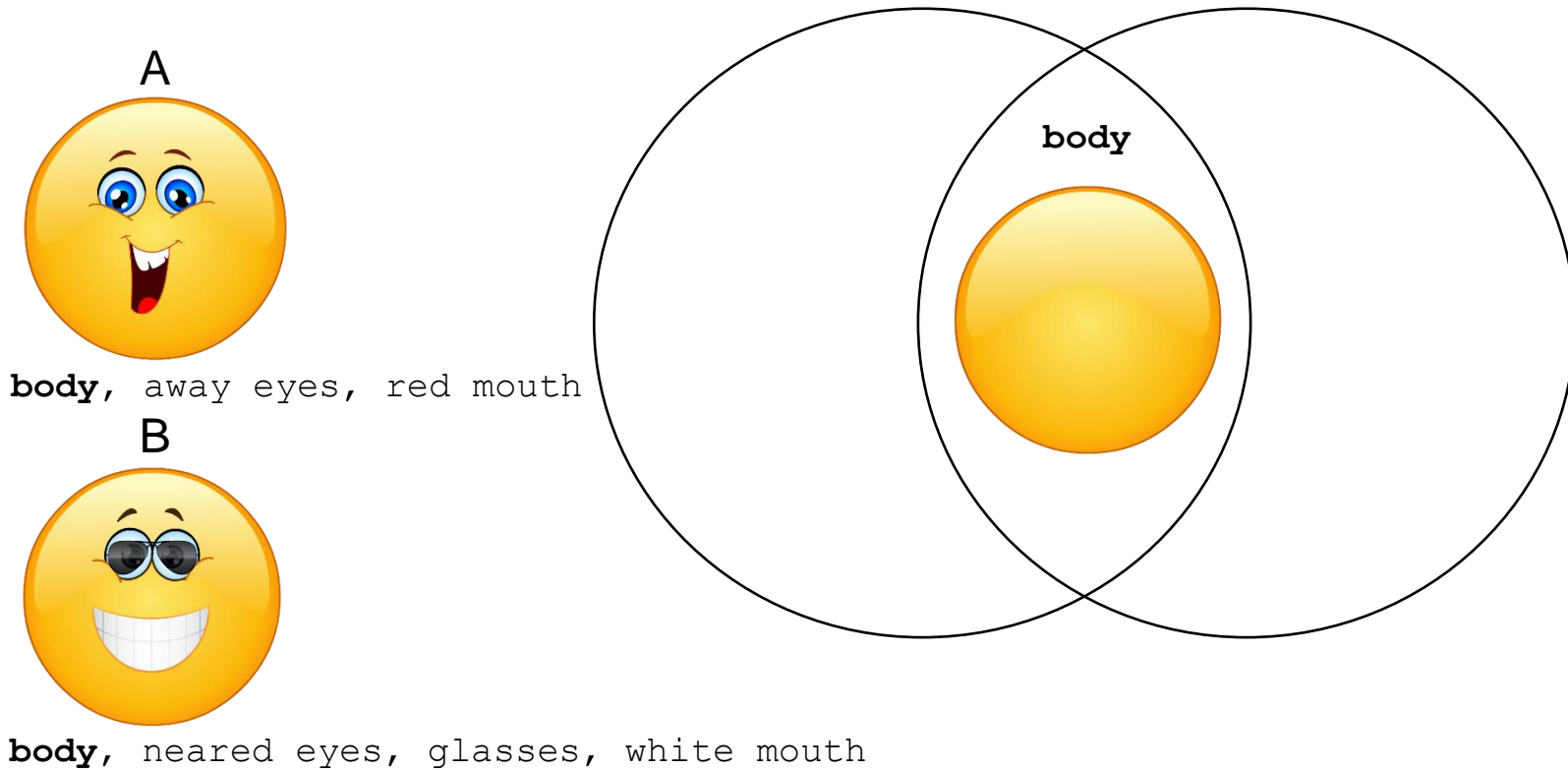
COMPARISON-BASED FEATURE LOCATION

- Commit operation is based on comparison of features and implementation of variants using five rules⁴
- Assume two variants A and B:
 - 1) Common artifacts (in A and B) likely trace to common features (in A and B)
 - 2) Artifacts in A and not B likely trace to features in A and not B, and vice versa
 - 3) Artifacts in A and not B do not trace to features in B and not A, and vice versa
 - 4) Artifacts in A and not B at most trace to features in A, and vice versa
 - 5) Artifacts in A and B at most trace to features in A or B

⁴Lukas Linsbauer, Roberto Erick Lopez-Herrejon, and Alexander Egyed. Variability extraction and modeling for product variants. Software & Systems Modeling, 2017.

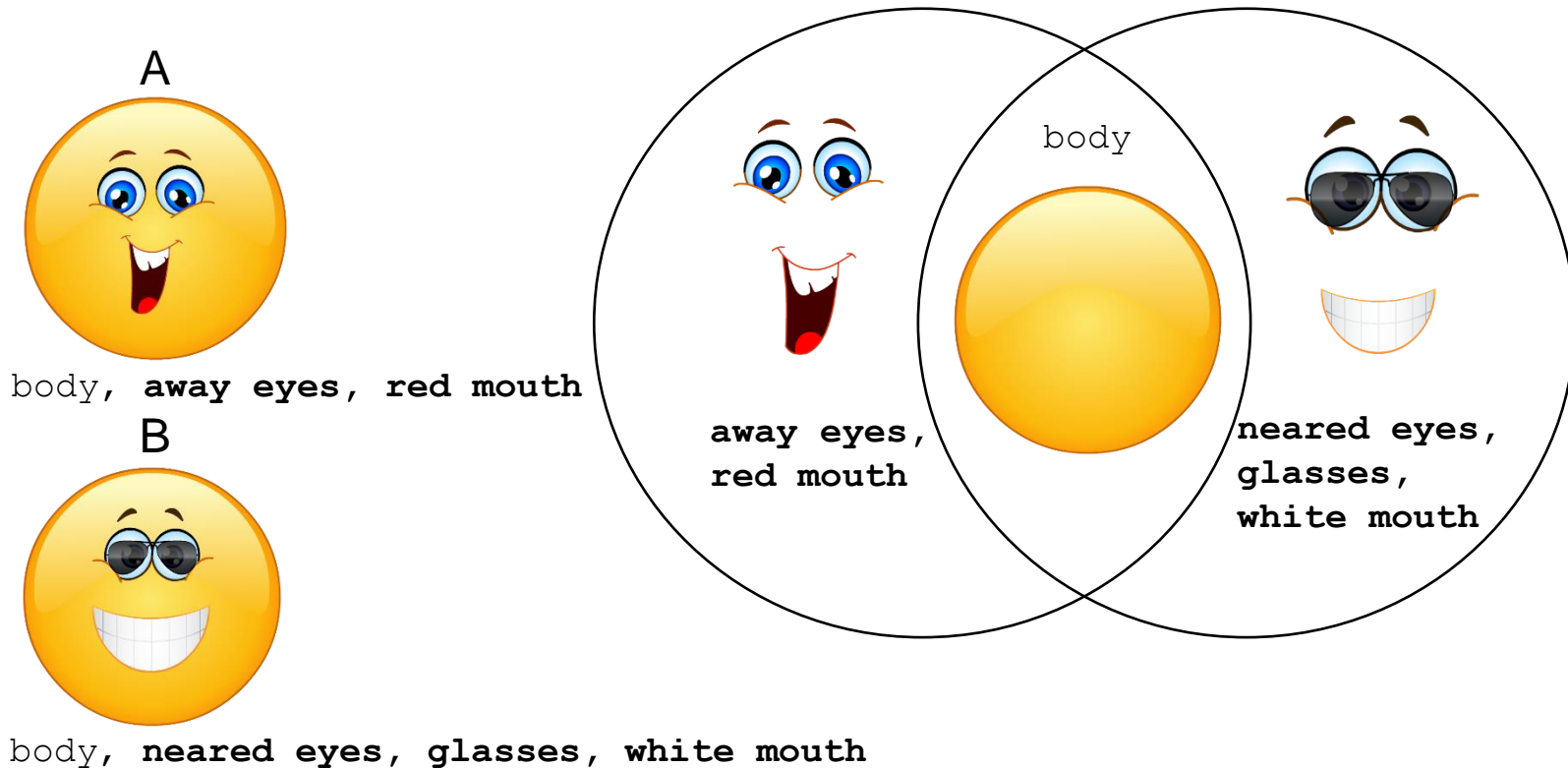
COMMIT OPERATION

- Rule 1: Common artifacts *likely* trace to common features



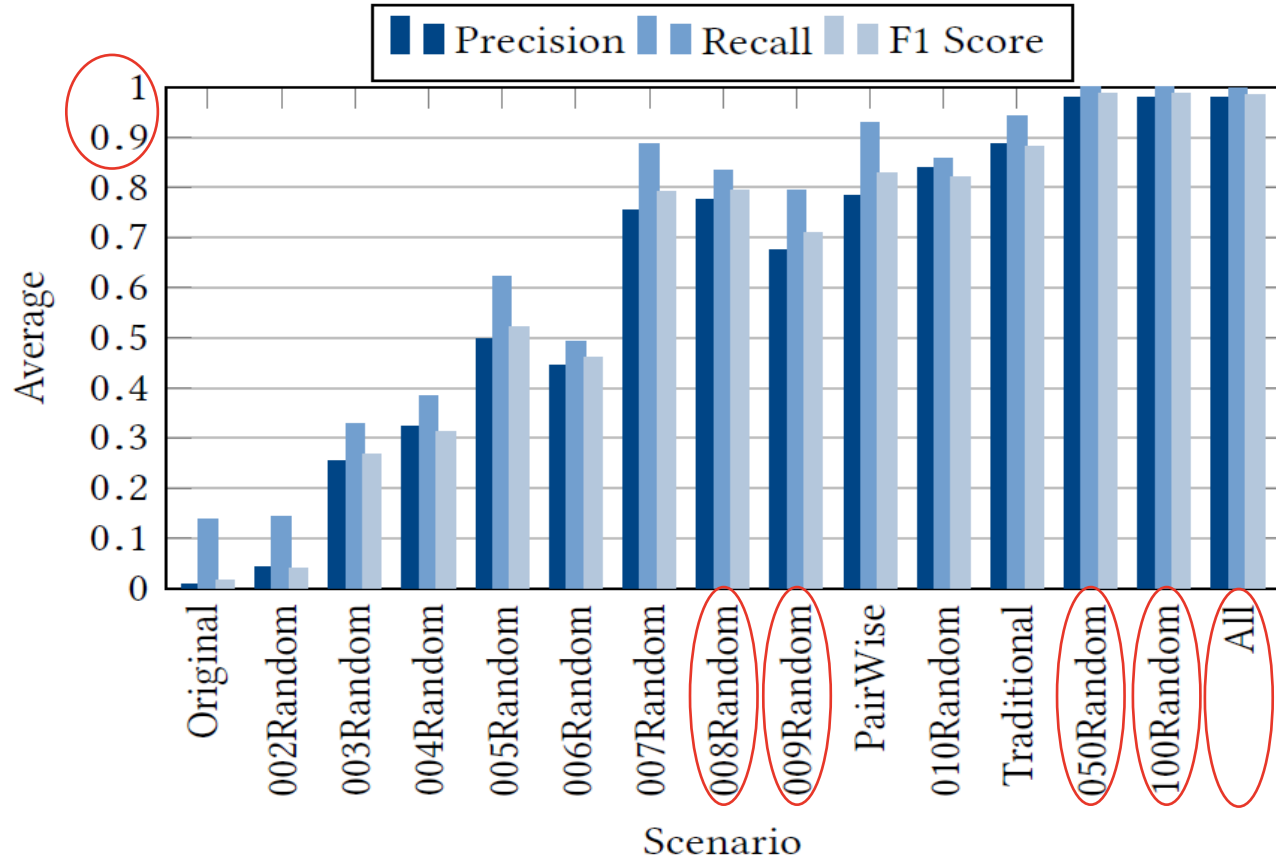
COMMIT OPERATION

- Rule 2: Artifacts in A and not B *likely* trace to features that are in A and not B, and vice versa



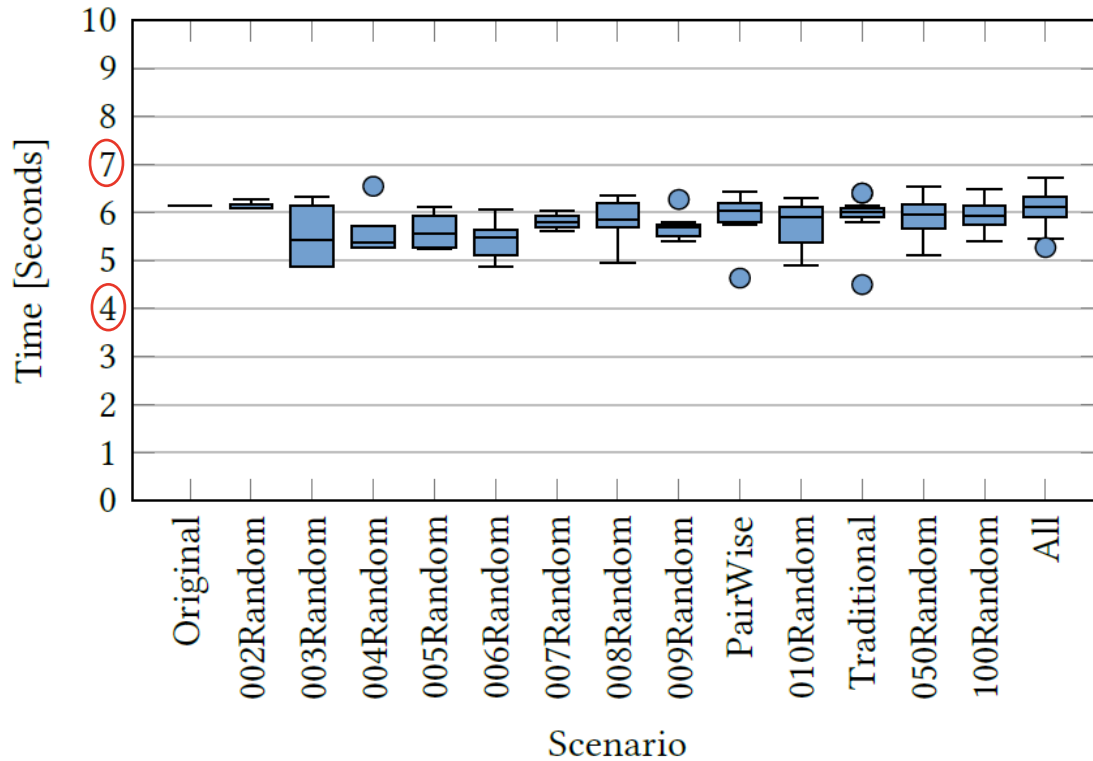
RESULTS

- Precision
- Recall
- F1 Score



RESULTS

■ Time Performance



Measured on a laptop model HP ZBook 14, with Intel(R) Core(TM) i7-4600U processor (2.10GHz, 2 cores), 16GB of RAM and SSD storage.

FINAL REMARKS

- We presented a solution to the ArgoUML SPL feature location challenge
 - Automatic feature location technique
 - Based on the comparison of features and implementation of a set of variants
 - Scenarios with more variants available had better results in the comparison based feature location
 - Runtime increases linearly with the number of variants

DATA AVAILABLE



■ Artifacts for the paper:

- <https://github.com/jku-isse/SPLC2019-Challenge-ArgoUML-FeatureLocation>

■ ECCO tool repository:

- <https://github.com/jku-isse/ecco/tree/develop>

■ ArgoUML challenge repository:

- <https://github.com/but4reuse/argouml-spl-benchmark>

■ Data Input:

- ArgoUML SPL Benchmark scenarios code (set of Java files)
- Configurations of Variants with its respective features

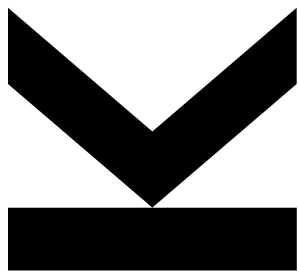
■ Data Output:

- Set of computed traces in plain text
- Computed metrics in % (Precision, Recall and F1 Score)
- Recorded runtimes per commit operation in milliseconds



COMPARISON-BASED FEATURE LOCATION IN ARGOUML VARIANTS

THANK YOU!



Gabriela Karoline Michelon
gabriela.michelon@jku.at

Lukas Linsbauer
lukas.linsbauer@jku.at

Wesley K. G. Assunção
wesleyk@utfpr.edu.br

Alexander Egyed
alexander.egyed@jku.at