

The ARDoCo Tool Landscape: REST API, TraceView, and TraceViz for Architecture Traceability

Jan Keim, Dominik Fuchß, Sophie Corallo, Tobias Hey, Julian Winter, Kevin Feichtinger

KASTEL / KIT – ASE 2026, Munich, Tools and Datasets Track

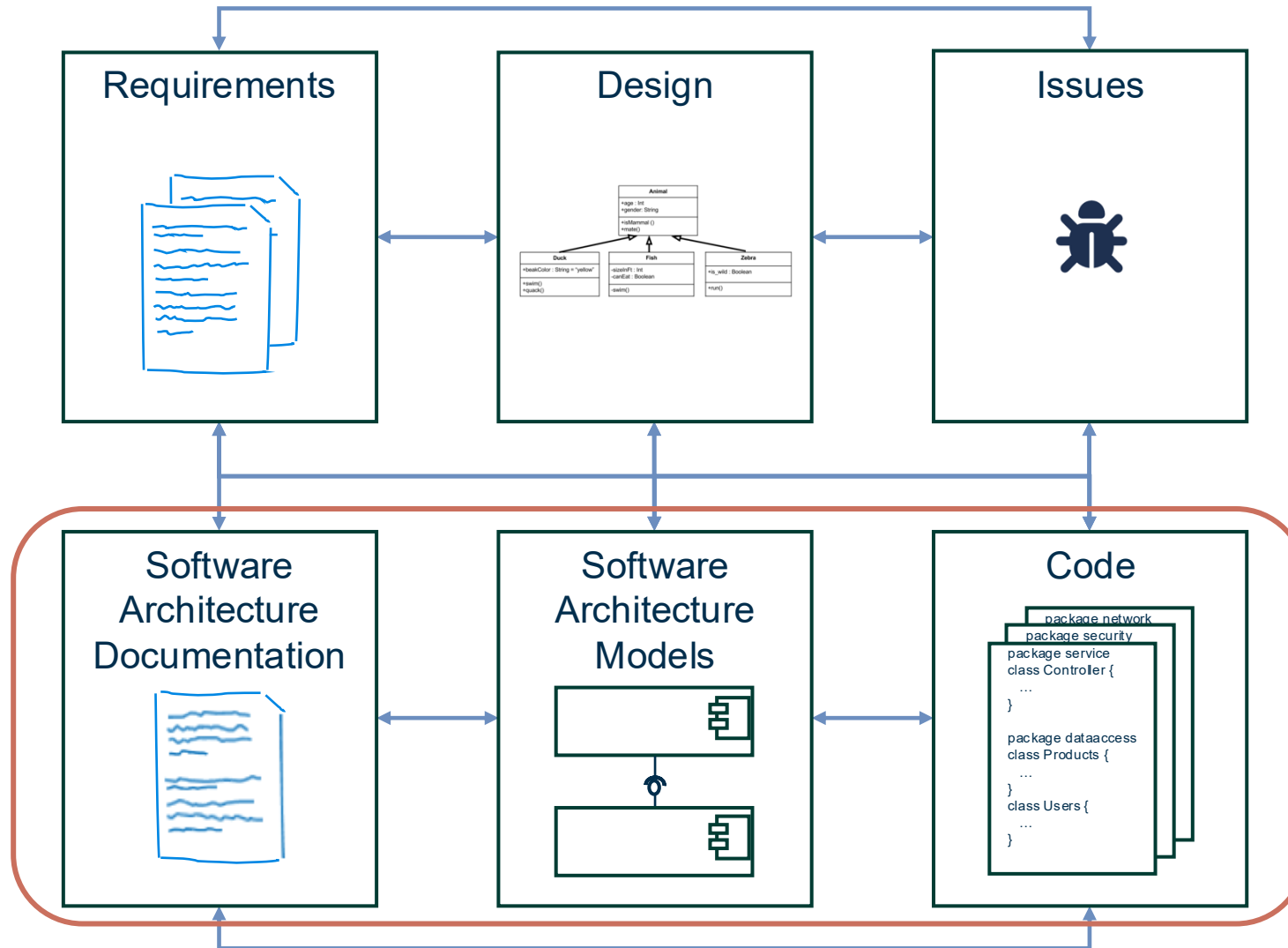


REST API

TraceView

TraceViz

What makes Trace Links important?



Trace links are evidently useful for

Software Maintenance

Bug Localization

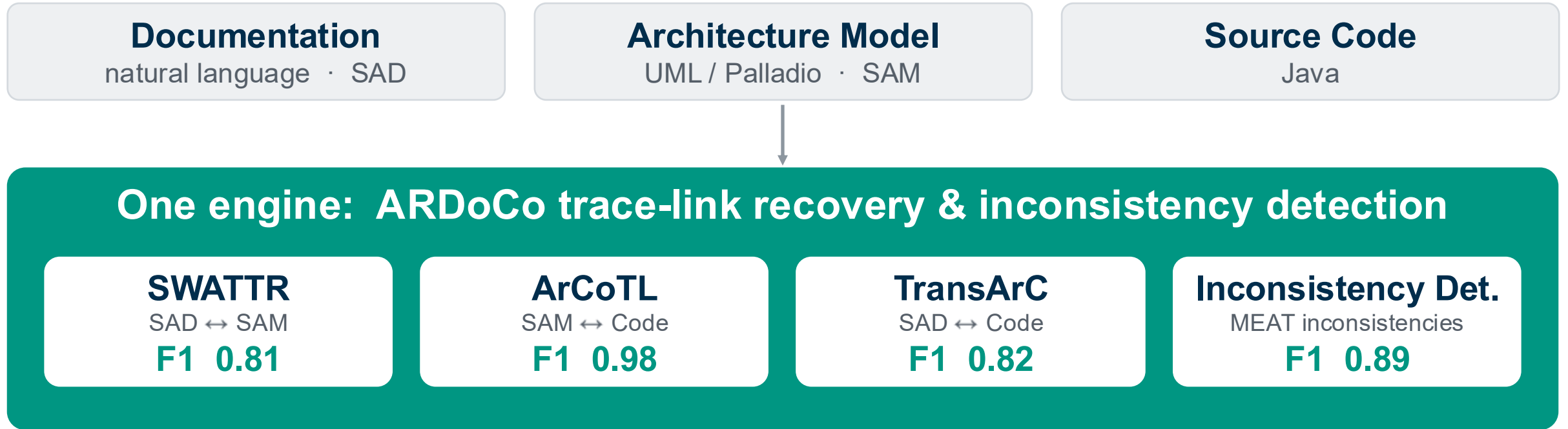
Change Impact An.

System Security

...

□ Artifact
↔ Relation

Automating Requirements and Documentation Comprehension



Three Interfaces of ARDoCo

SWATTR · ArDoCode · ArCoTL · TransArC

ARDoCo REST API for tool integrators

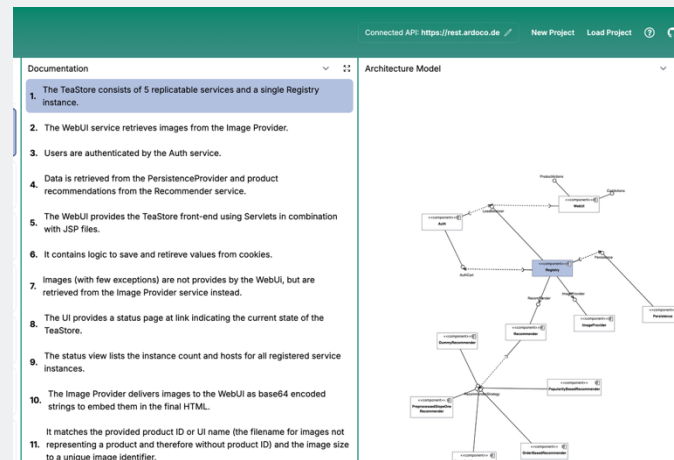
```
POST /api/swattr/start
  inputText, inputArchitectureModel

202 { "requestId": "MediaStore-5c1c..." }

GET .../wait-for-result/{id}
200 { "traceLinks": [
  { "sentence": 17,
    "modelElement":
      "MediaManagement" } ],
  "cached": true }
```

- Four TLR pipelines, one HTTP call
- Async: start · wait-for-result
- Cached in Redis, JSON, Swagger UI

TraceView for architects



- Zero install
- All three artifacts side by side
- Click once; see related elements.

TraceViz for developers

Trace-Links (blue) · 26 | Trace-Links (red) · 14

```
27 The Recommender is used to generate inc
28 It is trained using all existing orders
29 Recommendations are generated based on
30 The item rating of the users are based
31 If the current user is unknown (e.g., i
32 If the user is known, Slope One as item
33 We implemented two versions of the algo
34 One CPU-intensive, calculating the item
35 Furthermore, one order-based nearest-ne
36 Its recommendation time is dependent or
37 The Registry provides information about
```

ready Ln 27, Col 86 Spaces: 4

- Trace links as gutter dots
- One click: sentence ↔ code.
- REST API, CSV or local LiSSA approach (LLM)

ARDoCo REST API

for tool integrators rest.ardoco.de · github.com/ardoco/rest

- Spring Boot service, publicly deployed
- Four pipelines, one controller each
 - **sad-sam** – SWATTR + inconsistencies
 - **sam-code** – ArCoTL
 - **sad-code** – ArDoCode
 - **sad-sam-code** – TransArC
- Four endpoints each: start, start-and-wait, get-result, wait-for-result
- **Transparent caching**: the run ID is a hash of the request → Redis, 24 h

```
POST /api/sad-sam-code/start
    inputText, inputArchitectureModel,
    inputCode
202 { "requestId": "MediaStore-5c1c02dc..." }

GET /api/sad-sam-code/wait-for-result/{id}
200 { "traceLinks": [
    { "sentence": 17, "modelElement":
      "MediaManagement" },
    { "sentence": 19, "modelElement":
      "Packaging" } ] }
```

- *identical input → served from cache*
- *Swagger UI at /swagger-ui*

TraceView: Trace Links in the Browser

for architects tv.ardoco.de · no installation, four-step wizard

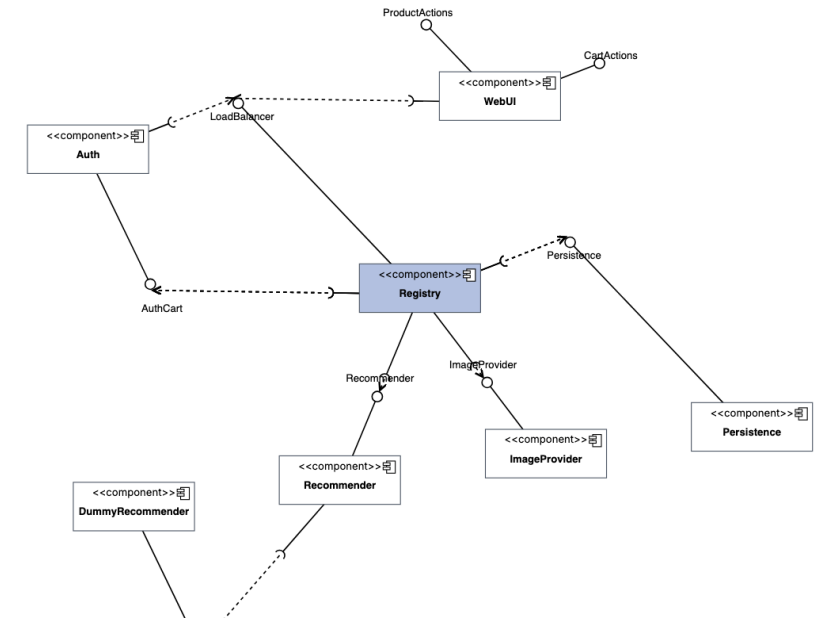
Documentation (SAD)

Documentation



1. The TeaStore consists of 5 replicatable services and a single Registry instance.
2. The WebUI service retrieves images from the Image Provider.
3. Users are authenticated by the Auth service.
4. Data is retrieved from the PersistenceProvider and product recommendations from the Recommender service.
5. The WebUI provides the TeaStore front-end using Servlets in combination with JSP files.
6. It contains logic to save and retrieve values from cookies.

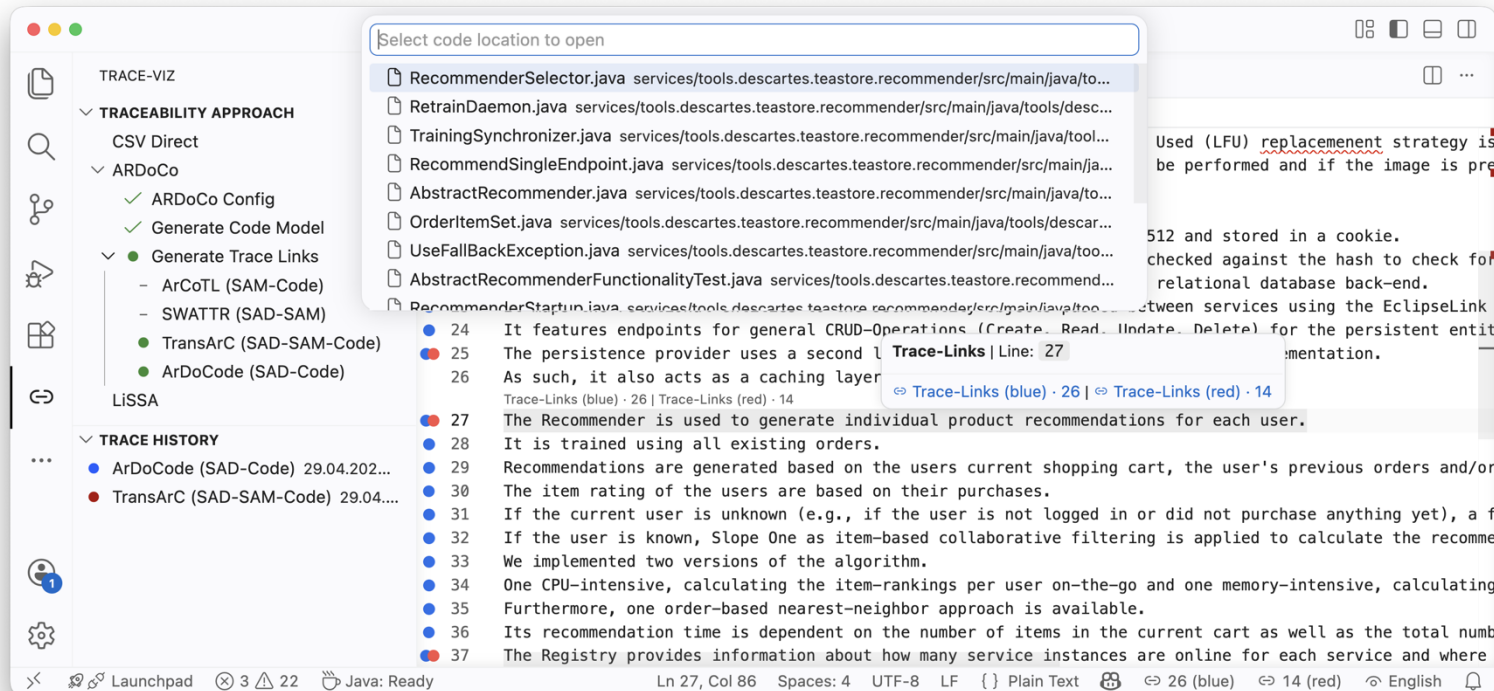
Architecture Model (SAM)



Select the first trace link: sentence 1 names a "single Registry instance", and the Registry component lights up in the model.

TraceViz: Trace Links in the IDE

for developers github.com/ardoco/traceviz · VS Code extension



Sidebar

pick the traceability approach, keep a history of runs

Gutter dots

one per linked documentation line;
hovering line 27 shows blue 26 · red 14

One click

a Quick Pick with the linked code

Approach-agnostic

ARDoCo REST API · CSV · LiSSA (LLMs)

Conclusion

- State-of-the-art architecture TLR:
 - **REST API**: four pipelines over HTTP, async + Redis caching
 - **TraceView**: browser wizard and exploration incl. inconsistencies
 - **TraceViz**: trace links on the gutter, one click into the code
- All three publicly deployed as Docker images / extension package
- Next:
 - LLM-based approaches LiSSA, ExArch behind the REST API
 - Revisit preliminary study (n = 7): in-IDE visualization improves comprehension



paper · tools · data
ardoco.de/c/ase26-tools



screencast
youtu.be/IOTEPZQ3tVs

rest.ardoco.de · tv.ardoco.de · github.com/ardoco/traceviz

replication package: doi.org/10.5281/zenodo.21533985