

An Analysis Plugin Toolbox for Jira Software

Bachelor's Thesis

Supervisor: Prof. Dr. Alexander Pretschner

Advisor: Valentin Zieglmeier

Email: {pretschn, zieglmev}@in.tum.de

Phone: +49 (89) 289 - 17834

Starting date: immediately



Fakultät für Informatik

Lehrstuhl 4

Software & Systems Engineering

Prof. Dr. Alexander Pretschner

Boltzmannstraße 3

85748 Garching bei München

Tel: +49 (89) 289 - 17834

<https://www4.in.tum.de>

Context

The research project “Inverse Transparency” examines a new form of data privacy. In short, all data accesses are allowed, but accesses are logged and made visible to so-called data owners. We define a data owner roughly as either the person creating a datum, or the person a datum relates to in content. For example, a dossier written by Sally containing personal information about Frank is owned by both Sally and Frank.

We evaluate different approaches to tracking data accesses in a tamper-proof and persistent way. Different layers of the access are looked at, from analysis tools creating knowledge to the lower-level access.

Goal

This thesis looks at the tracking of data accesses through the data analysis layer. We focus on the tool Jira Software¹ to limit the scope, but the work should aim to be generalizable to other cases of data analyses through similarly designed JavaScript plugins.

Our idea is to control the complete data usage relationship and thus prevent or at least limit misuse of the data. Additionally, as the data accesses are tracked at a high level, the semantics of the accesses should be easier to deduct, enabling more comprehensive tracking.

In this thesis, multiple analysis plugins are to be developed. These can later be employed to track data accesses. Accordingly, the work will comprise three steps: First, a taxonomy of targeted analyses is compiled. This taxonomy will be based on existing work and own research. Important parts of this section are the kinds of data that are accessed, a possible use case and target group for the analysis, as well as the semantics of the access from the perspective of the data owner. Second, a selection of representative analysis plugins is implemented as proof-of-concepts. The implementations should follow a “toolbox” model, as in, they should be pluggable and easy to replace or extend. The implementations need to load actual data through Jira API requests, as well as create and send semantically enriched access information requests to the tracking server. Optionally, an access request clearance step can be added. Finally, the implementations are evaluated based on their functionality (Are the analyses functional in relation to the use case and target group?), code reuse and pluggability, as well as ease of use.

Work Plan

1. Compile a taxonomy of exemplary analyses.
2. Implement selected analysis plugins.
3. (Optional) Add access tracking functionality.
4. (Optional) Add access request clearance functionality.
5. Evaluate the implementation.
6. Document of the work in the thesis.

Deliverables

- Source code of the implementation.
- Thesis written in conformance with TUM guidelines.

References

- [1] Wenxuan Li. “Analyzing sensible metrics for individual’s performance based on issue tracking system data.” *Seminar Inverse Transparency (WS 19/20)*. (unpublished).
- [2] Leonard Scheidemantel. “Aspects and graphical representations of team health for an issue tracking system overview dashboard.” *Seminar Inverse Transparency (WS 19/20)*. (unpublished).

¹<https://www.atlassian.com/software/jira>