Capturing Provenance Information in the File System

✉ Lars Gleim[1], ✉ Leon Müller[1], Florian Brillowski[2], and Stefan Decker[1,3]

[1] Databases and Information Systems, RWTH Aachen University, Germany
[2] Institute of Textile Technology, RWTH Aachen University, Germany
[3] Fraunhofer FIT, Sankt Augustin, Germany

Abstract. As business processes are increasingly complex, agile, and specialized, provenance information can improve interpretability and contextualization of process data. While individual process steps frequently employ digital computer files, their relationships within the overall process are rarely captured. To address this issue, we extend the *factFUSE* system for managing versioned Web resources in the file system to capture provenance relationships. By introducing an extensible commit system, we enable recording the relations between digital files and resources in process steps (activities), which are then captured as RDF metadata using the W3C PROV standard. Our evaluation shows that users without prior experience in provenance management successfully employ the system to capture semantic process provenance, attesting to excellent usability and promising utility. factFUSE is available for practical use under open source GNU AGPLv3 license.

Keywords: Semantic Data Management · Version Control System · FAIR Data · Desktop Computing · FactDAG · FactStack · FUSE · Linked Data Platform

1 Introduction

Today's business processes frequently involve handling data in the form of *computer files*. As decision-making processes are increasingly data-driven, the provenance and semantic context of these files and the contained information can improve interpretability and explainability. Inspired by the global success of the Git [6] system as unified data and version management system in software development (across vendors, programming languages, operating systems, and geographical borders), we explore how to integrate provenance management directly and transparently into the file system.

Design Goals. We strive to fulfill the following design goals with our system:

- **G1** *Interoperability* with existing desktop applications and workflows, which provide almost universal support for interfacing with local files and file systems, enabling the semantic management of arbitrary computer files.
- **G2** *Transparent adoption* of the fundamental *FAIR principles* [9] of scientific data management to ensure reusability and utility of the collected metadata.
- **G3** Implementation of best practices throughout the data life-cycle, notably including resource *versioning*, revision *immutability*, and *unique referenceability* [3].
- **G4** *Semiautomatic semantic data annotation*, combining automatic metadata and provenance collection with contextual user prompts if applicable.
- **G5** *Good usability* for users that are already familiar with hierarchical data management in the file system.

Contributions. Addressing these goals, we propose a platform-independent concept for provenance capturing in the file system and provide a corresponding open-source implementation for both Linux and macOS. We further provide an evaluation of the system w.r.t. the design goals defined above. Sec. 2 introduces the main concept and presents the implementation of an extensible commit system for *factFUSE*. Sec. 3 discusses its quantitative and qualitative evaluation. We conclude our work in Sec. 4.

2 Concept & Realization

To capture provenance information on *computer files* and to enable context embedding in traditional data management environments, a bridge between the classic hierarchical file system and semantic data management is fundamental. We extend upon the recently proposed *factFUSE* [7] system for the joint management of computer files and semantic data and metadata in the file system. The solution is based on FactStack [4], a unified semantic data management system integrating RDF, arbitrary data types, and computer files in a fundamentally provenance-linked knowledge graph through a combination of open Web standards according to the FAIR principles [9]. *factFUSE* maps Linked Data Platform (LDP) [8] resources into the local file system and vice versa through a FUSE file system driver. This allows users to interact with semantic data in the same way as with regular *computer files* and enables the drag-and-drop integration of files into semantic graphs. Each resource is versioned using the HTTP Memento protocol and augmented with a dedicated metadata record linked via the HTTP `rel="describedby"` `Link` header as specified by the LDP standard [8].

Concept. Extending upon this foundation provided by the *factFUSE* system, we embed provenance management into the traditional workflow of managing files in the file system by introducing (i) a *Commit system* inspired by the distributed version control system GIT [6], which may expose user interfaces for metadata collection during the process of persisting resource modifications, and (ii) extensible *Metadata Interfaces* that may be opened directly from the file system explorer (either through context menus or by adding buttons to the operating system's file system explorer itself) to manage and display metadata of the selected resources, as illustrated in Fig. 1). Subsequently, the

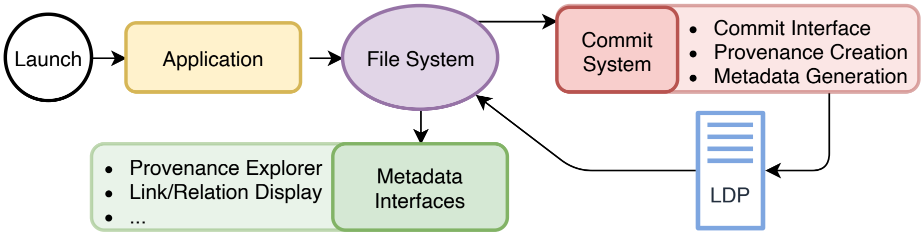


Fig. 1. The factFUSE concept, extended to capture computer file provenance through a commit system and corresponding metadata interfaces in the file system.

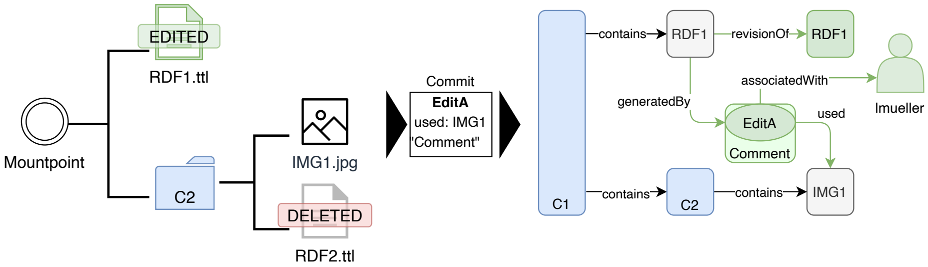
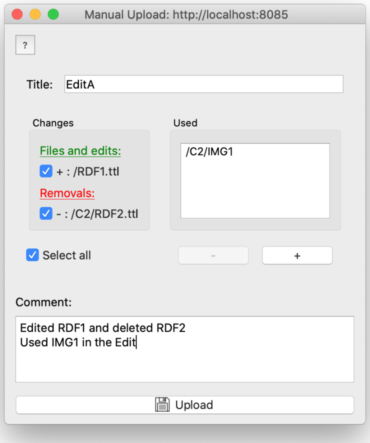


Fig. 2. Modifications in the local file system are enriched with semantic metadata collected through UI extensions. Data and semantic metadata are then jointly committed to the LDP server.

collected provenance information is then persisted to the resource's metadata record to track the changes and revisions that resources go through and capture process knowledge.

Realization. factFUSE is a NodeJS application, providing a custom user-space file system based on the FactStack [4] system which provides primitives for handling resource versioning, network communication, and metadata management. Additionally, it provides helper functions for provenance management and preservation according to the W3C PROV standard [5,1], which expresses data provenance through *entities*, *activities*, and *agents*. factFUSE tracks changes made to resources in the file system in an internal cache before asynchronously synchronizing them with the upstream LDP server. To capture process provenance in the file system, we introduce an extensible commit system.

A commit (modeled as a PROV activity) carries information on the time and content of changed resources (PROV entities), a title, its author (PROV agent), a message, and possibly additional resources that were used (but not modified) in the process. Fig. 2 shows an example in which two resources in the local file system are modified. The CommitGUI illustrated in Fig. 3 is then used to capture provenance and additional metadata of the generating process and to commit the data to the upstream LDP server.


Manual Upload: http://localhost:8085

Title: EditA

Changes

Files and edits:
☑ + : /RDF1.ttl
Removals:
☑ - : /C2/RDF2.ttl

Used

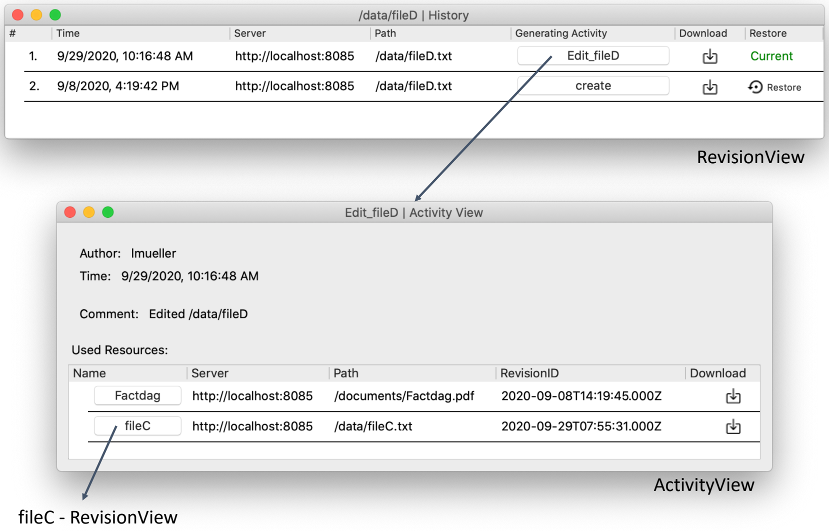
/C2/IMG1

☑ Select all

Comment:
Edited RDF1 and deleted RDF2
Used IMG1 in the Edit

Fig. 3. Configuring changes & metadata to include when creating a process step commit.

In order to enable the semantic exploration and management of existing resources, a set of user interfaces (see Fig. 4) has been designed and implemented, that are accessible through the right-click context menu, extending the file system's functionality. The *RevisionView* displays a history of all existing revisions of a resource as well as each revisions generating activity. Additionally, UI elements to download or restore revisions are provided. Inspecting a revision's generating activity allows further inspection along the edges of the PROV graph, by opening the *ActivityView*. An activity's *ActivityView* holds information on its responsible Agent, the time of execution, and a list of resources that have been used during the activity. A full overview of the implemented interfaces in in-use scenarios can be found on the projects repository page[4]. This approach of providing context-dependent UI extensions to manage and visualize metadata, in alignment with individual ontologies and best practices of specific application domains, allows for the progressive and configurable adoption of semantic data management to the traditional file system.



#	Time	Server	Path	Generating Activity	Download	Restore
1.	9/29/2020, 10:16:48 AM	http://localhost:8085	/data/fileD.txt	Edit_fileD	⤓	Current
2.	9/8/2020, 4:19:42 PM	http://localhost:8085	/data/fileD.txt	create	⤓	⟲ Restore

Author: lmueller
Time: 9/29/2020, 10:16:48 AM

Comment: Edited /data/fileD

Used Resources:

Name	Server	Path	RevisionID	Download
Factdag	http://localhost:8085	/documents/Factdag.pdf	2020-09-08T14:19:45.000Z	⤓
fileC	http://localhost:8085	/data/fileC.txt	2020-09-29T07:55:31.000Z	⤓

Fig. 4. Exploring the provenance of a resource through file system UI extensions.

3 Evaluation & Discussion

Along with the proposed provenance management extensions, factFUSE provides a tool to manage version-controlled Web resources in the file system which automatically collects provenance information with interfaces to explore and use them. Goal **G1** is met by design since factFUSE allows for the drag-and-drop integration of arbitrary computer files and its representation of Web resources in the file system that can be used and edited in existing desktop applications. Goals **G2** and **G3** are met by employing the FactStack data management system as detailed in [4]. Goal **G4** is addressed by extending factFUSE with a commit system as detailed above. Provenance information is automatically captured and can either be synchronized with the LDP in near real-time or be manually customized and committed by the user. In order to validate the user-friendliness specified by goal **G5**, a user study was conducted evaluating the management and exploration of semantic provenance metadata as well as versioned resources. The participants – all without prior experience in provenance management – were asked to

[4] `https://git.rwth-aachen.de/i5/factdag/factfuse`

complete a set of six tasks that each represented a core functionality and use case of the system. A detailed description of the study, its participants, raw task, result data, and further discussion can be found in the repository. The System Usability Scale (SUS) [2] results – a score of 83.25 – attested excellent usability to the factFUSE system, fulfilling goal **G5**. Summarizing the key results, all participants successfully used the system to solve the provenance and version management-related tasks and shared positive feedback about the system and its utility. The commit system was further identified as an extension point for future metadata collection, possibly depending on the type and context of the modified resources, enabling deeper integration of semantic data management primitives into the file system.

4 Conclusion

In this paper, we presented a solution for the collection and management of provenance information in the file system, by extending the factFUSE system for managing versioned Web resources. Through the implementation of an extensible commit system, we provide a solution for semi-automatic provenance collection combined with manual user prompts for additional metadata that is expressed as RDF, using the W3C PROV standard. This enables the semantic embedment of digital files into processes and subsequently enables a detailed overview of the relations within a process and between different resources. Our user study yielded excellent usability results and showed a quick adoption of principles by new users. As such, factFUSE provides a first step towards the flexible integration of traditional file-based data management with semantic metadata using open Web standards and technologies and is available open-source for community discernment.

Acknowledgments Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC-2023 Internet of Production – 390621612.

References

1. Belhajjame, K., B'Far, R., et al.: Prov-dm: The prov data model. W3C Recommendation (2013)
2. Brooke, J.: SUS: A quick and dirty usability scale. In: Usability Evaluation In Industry, pp. 207–212. CRC Press (1996)
3. Gleim, L., Pennekamp, J., et al.: FactDAG: Formalizing Data Interoperability in an Internet of Production. IEEE Internet Things J. **7**(4) (2020)
4. Gleim, L., Pennekamp, J., et al.: FactStack: Interoperable Data Management and Preservation for the Web and Industry 4.0. In: BTW (2021)
5. Gleim, L., Tirpitz, L., et al.: Expressing FactDAG Provenance with PROV-O. In: MEPDaW @ ISWC (2020)
6. Loeliger, J., McCullough, M.: Version Control with Git: Powerful tools and techniques for collaborative software development. O'Reilly Media (2012)
7. Müller, L., Gleim, L.: Managing Versioned Web Resources in the File System. In: ICWE (2021)
8. Speicher, S., Arwe, J., et al.: Linked Data Platform 1.0. W3C Rec. (2015)
9. Wilkinson, M.D., Dumontier, M., et al.: The FAIR Guiding Principles for scientific data management and stewardship. Sci. Data **3**, 160018 (2016)