Solid Proof of Concept in an Enterprise Loan Request Use Case

Daniel Henselmann[1,*], Karina Kolinsky[1], Sebastian Schmid[2], Daniel Schraudner[2], Andreas Both[3,4] and Andreas Harth[1,2]

[1]*Fraunhofer Institute for Integrated Circuits IIS, Nürnberg, Germany*

[2]*Friedrich–Alexander University Erlangen-Nürnberg, Nürnberg, Germany*

[3]*DATEV eG, Nürnberg, Germany*

[4]*Leipzig University of Applied Sciences, Leipzig, Germany*

Abstract

Social Linked Data (Solid) origins from the idea of a decentralized social Web with separated data, identities, and applications. This demo provides a proof of concept to show the usability of the Solid specifications in an enterprise data exchange scenario. The loan request of a company via a tax accountant serves as the use case.

Keywords

Solid, WebID, Read/Write Linked Data, Enterprise Data Exchange

1. Introduction

Most companies face the challenge to exchange data between enterprises. Currently, common solutions are proprietary cloud services provided by corporations like Microsoft, Google, Dropbox, Nextcloud, and ownCloud. However, these don't offer the decentralization and trustworthiness needed for some use cases with more confidential enterprise data.

Solid (derived from "Social Linked Data") is a collection of technological specifications for read/write Linked Data supported by authentication and authorization [1]. It builds upon the RESTful HTTP service specification of the Linked Data Platform (LDP)[1]. Solid uses established (Semantic) Web standards [2] and is therefore easy to integrate and use. It is also independent of any centralized element. Furthermore, Solid has lightweight specifications that make no restrictions regarding the backend database used. The physical *Solid Pod*[2] location and identity provider can be freely chosen or self-hosted. Additionally, Solid provides detailed and flexible access control, which is crucial for enterprise use cases.

✉ daniel.henselmann@iis.fraunhofer.de (D. Henselmann); karina.kolinsky@iis.fraunhofer.de (K. Kolinsky); sebastian.schmid@fau.de (S. Schmid); daniel.schraudner@fau.de (D. Schraudner); andreas.both@datev.de (A. Both); andreas.harth@fau.de (A. Harth)

🆔 0000-0001-6701-0287 (D. Henselmann); 0000-0001-7211-6385 (K. Kolinsky); 0000-0002-5836-3029 (S. Schmid); 0000-0002-2660-676X (D. Schraudner); 0000-0002-9177-5463 (A. Both); 0000-0002-0702-510X (A. Harth)
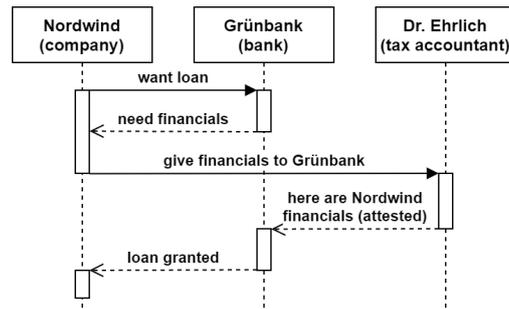
[1]cf., https://www.w3.org/TR/ldp/

[2]A data pod is a place for storing documents, with access control (cf., https://solidproject.org/TR/protocol).

Figure 1: Scenario of the loan request use case

Therefore, this demo shows the usage of Solid for the data exchange process between multiple enterprises. For confidential enterprise data, the Solid specifications build a foundation we use for this proof of concept and will enhance in future work to ensure a level of trust for the data exchange process. For an actual use case (see Section 2), we partnered with DATEV eG, a German software company particularly providing business software (e.g., for tax accounting) on large scale.

While the development of Solid was motivated by a decentralized social Web for personal data [2], Solid has been adapted to a few other use cases[3] showing the possibilities of the Solid specifications. Noteworthy published Solid use cases are Building Information Modelling (BIM) in the construction industry [3], interconnecting Internet of Things (IoT) devices [4], sharing public transportation data [5], and machine-to-machine sales contract conclusion [6].

The International Data Spaces (IDS)[4] and GAIA-X[5] initiatives aim for a complete ecosystem regarding data sovereignty and trustful data sharing. IDS is further developed at this point in time, but still just starting with implementations. IDS and GAIA-X partially utilize Semantic Web standards like LDP and ontologies [7]. Solid could very well supplement IDS and GAIA-X and act as a puzzle piece in their ecosystems.

2. Use Case

We investigate the data-sharing scenario of a loan request for tax accountants. Figure 1 shows the process, which is abstracted but based on real life. A company named Nordwind asks a bank named Grünbank for a loan. Grünbank needs to check Nordwind's finances to grant a loan. Therefore, Nordwind instructs its tax accountant, Dr. Ehrlich, to provide an attested summary of its finances to Grünbank. With that information, Grünbank can decide on the loan to Nordwind. This scenario serves as our use case for data sharing in an enterprise environment following the Solid approach.

---

[3]See https://solidproject.org/apps for an overview of Solid apps with use cases.
[4]cf., https://internationaldataspaces.org
[5]cf., https://www.data-infrastructure.eu

Table 1
Entities in the setup for the loan request use case

Solid App	An Angular app that provides a user interface to the employees of the involved organizations to execute the use case. The app reads/writes RDF data on the Solid Pods using HTTP requests.
Nordwind App	An instance of the Solid App. It runs in the web browser of a Nordwind employee. The app offers features adapted to Nordwind users.
Ehrlich App	An instance of the Solid App. It runs in the web browser of an Ehrlich employee. The app offers features adapted to Ehrlich users.
Grünbank App	An instance of the Solid App. It runs in the web browser of a Grünbank employee. The app offers features adapted to Grünbank users.
Nordwind Pod	A Solid Pod that contains Nordwind's financial data.
Ehrlich Pod	A Solid Pod that contains Nordwind's financial documents.
Identity Provider	The identity providers host Solid servers that contain the profile documents belonging to a WebIDs. We use multiple existing identity providers to create WebIDs for employees of the three organizations.

3. Implementation

To implement the use case with Solid, we set up a *Solid App*, multiple Solid Pods for data, and several identities represented as WebIDs[6] used for access control [8]. Table 1 lists the deployed entities. It shows three distinct instances of our *Solid App* that might run in the web browser of an employee at one of the three involved organizations. Our developed *Solid App* offers varying features to the user depending on their association with the three organizations. The app is deployed on a generic web server. This server also hosts a Community Solid Server[7], an openly available implementation of a Solid server. On the Solid server, we instantiated two Solid Pods. The *Nordwind Pod* contains the financial data of Nordwind using the file system of the web server as storage backend. The *Ehrlich Pod* contains financial documents created by the Dr. Ehrlich tax accountants on an AWS S3 bucket. From the app, users log in to their *Identity Provider* to enable authentication and authorization at the Solid Pods. This demonstrates that multiple identity providers work with the same *Solid App* and the Pods used for data storage can be independent of the identity provider. All communication happens through RESTful HTTP requests.

The sequence diagram in Figure 2 shows the usage of the setup for the use case. The actors in the diagram are three instances of the Solid App with corresponding features of a logged-in user, the *Nordwind Pod*, and the *Ehrlich Pod*. We omit here authentication and authorization steps for simplicity, both are provided by the Solid server. The presented sequence starts at the *Nordwind App*, from where accounting transaction data is posted to the *Nordwind Pod*. After this data is pulled with a GET request to the *Ehrlich App*, an audited Cash Basis Accounting[8] document (CBA) is created as a summary of the financial situation of Nordwind. The CBA is then posted to the *Ehrlich Pod*. Afterwards, its Access Control List[9] (ACL) is successively

---

[6]cf., https://www.w3.org/wiki/WebID

[7]cf., https://github.com/CommunitySolidServer/CommunitySolidServer

[8]German: "Einnahmenüberschussrechnung" (EÜR)

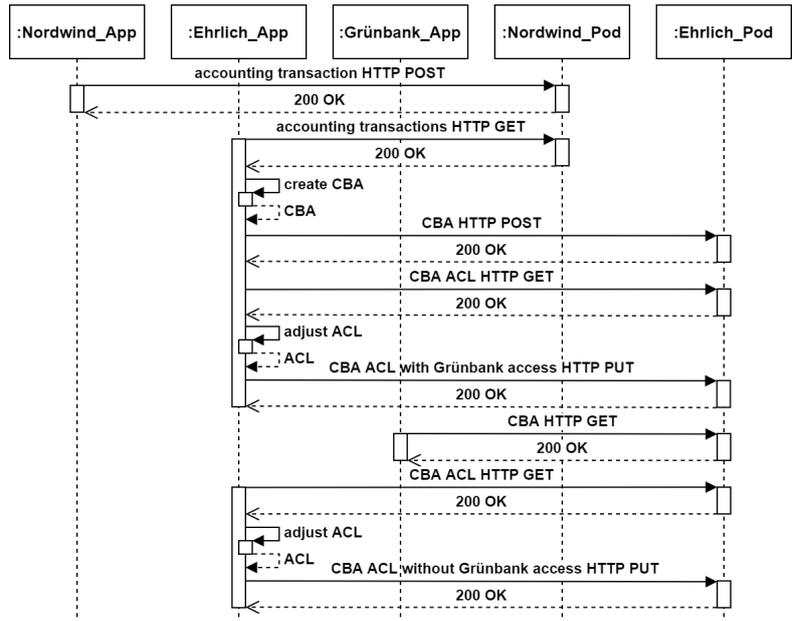[9]cf., https://solid.github.io/web-access-control-spec/#introduction

Figure 2: Sequence diagram of the loan request use case

retrieved, adjusted, and overwritten to change the authorization to allow access to the CBA for Grünbank employees. Afterwards, the CBA can be investigated with the *Grünbank App* to decide about the loan grant. Finally, the access for Grünbank to the CBA is removed through the *Ehrlich App* with another adjustment of the ACL.

A video showing the proof of concept as a demo is available at https://purl.org/solid-poc-app/demo. The repository of the developed Solid App is available on GitHub at https://github.com/wintechis/DATEV-Solid-PoC-App. The basic ontology used to model the RDF data of the accounting transactions and the CBAs is available at https://purl.org/solid-poc-app/ontology.

4. Conclusion & Outlook

The proof of concept shows the usage of Solid in an enterprise loan request use case. With the usage of multiple Solid Pods as well as different identity providers and data backends, Solid displays its strengths to share data in an enterprise scenario while still keeping control over it.

Our next steps to advance the use case are additions to our technology stack besides the Solid specifications towards a mature solution for confidential enterprise data. These include features regarding data integrity, security against manipulation, confirmability of identities, and limits of authority. Another topic is more complex: the role/group hierarchies for access control.

Further possibilities lie in the automated processing of data from Solid Pods. The creation of the CBA in our use case could happen automatically whenever the accounting transitions are updated. Consequently, Grünbank could receive an automatic notification with access to the new CBA and automatically provide a fitting loan offer to Nordwind using that data. Solid could thus work as an enabler for real-time economy.

Another application for Solid in the enterprise environment is automated data aggregation, e.g., in supply chains. With increasingly strict regulations, enterprises need to unfold their supply chains to the very bottom. To make statements about the sustainability of an entire supply chain, data must be aggregated vertically over multiple layers of suppliers.

Acknowledgments

References

[1] Sarven Capadisli, Tim Berners-Lee, Ruben Verborgh, Kjetil Kjernsmo, Solid Protocol, 2021. URL: https://solidproject.org/TR/protocol.

[2] E. Mansour, A. V. Sambra, S. Hawke, M. Zereba, S. Capadisli, A. Ghanem, A. Aboulnaga, T. Berners-Lee, A Demonstration of the Solid Platform for Social Web Applications, in: Proceedings of the 25th International Conference Companion on World Wide Web - WWW '16 Companion, ACM Press, 2016, pp. 223–226. doi:10.1145/2872518.2890529.

[3] Jeroen Werbrouck, Pieter Pauwels, Jakob Beetz, Léon van Berlo, Towards a decentralised common data environment using linked building data and the solid ecosystem, in: Advances in ICT in Design, Construction and Management in Architecture, Engineering, Construction and Operations (AECO) : Proceedings of the 36th CIB W78 2019 Conference, 2019, pp. 113 – 123.

[4] F. Komeiha, N. Cheniki, Y. Sam, A. Jaber, N. Messai, T. Devogele, Towards a Privacy Conserved and Linked Open Data Based Device Recommendation in IoT, in: Service-Oriented Computing – ICSOC 2020 Workshops, volume 12632 of *Lecture Notes in Computer Science*, Springer, Cham, 2021, pp. 32–39.

[5] W. Zhao, B. Zhou, C. Zhang, Heterogeneous Social Linked Data Integration and Sharing for Public Transportation, Journal of Advanced Transportation 2022 (2022). doi:10.1155/2022/6338365.

[6] Xinni Wang, C. H.-J. Braun, Andreas Both, Tobias Käfer, Using Schema.org and Solid for Linked Data-based Machine-to-Machine Sales Contract Conclusion, in: Companion Proceedings of the Web Conference 2022 (WWW '22 Companion), ACM, New York, NY, USA, 2022, p. 4. URL: https://www2022.thewebconf.org/PaperFiles/79.pdf.

[7] Boris Otto, GAIA-X and IDS, Position Paper, International Data Spaces Association, Berlin, 2021. doi:10.5281/ZENODO.5675897, version number: 1.0.

[8] P. Mainini, A. Laube, Access control in linked data using WebID, in: M. Martin, M. Cuquet, E. Folmer (Eds.), Joint Proceedings of the Posters and Demos Track of the 12th International Conference on Semantic Systems - SEMANTiCS 2016 and the 1st International Workshop on Semantic Change & Evolving Semantics (SuCCESS'16), Leipzig, Germany, September 12-15, 2016, volume 1695 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2016, p. 8. URL: http://ceur-ws.org/Vol-1695/paper7.pdf.