

# Web Agent architecture for Incremental View maintenance

Dore Staquet<sup>\*†</sup>

Jan Van den Bussche<sup>\*</sup>

Bart Buelens<sup>†</sup>

August 26, 2024

## Web Agents architecture for Solid

In our research, we envisage both web agents and aggregators to maintain materialized views to swiftly respond to SPARQL queries over knowledge graphs. Figure 1 illustrates our vision of having a web agent for each Solid [1, 2] pod which acts as a layer between trusted partners and the pod. This architecture provides a framework for a more efficient way of retrieving information from the Solid pods, certainly in decentralized models. These materialized views allow for efficiently responding to repeated querying. With these web agents we envision to introduce a layer between the Solid pod and the trusted partner or user, most Solid technologies do not have a dedicated SPARQL endpoint available, which means that when querying a Solid pod the data is firstly downloaded locally and then the query is executed, this introduces some privacy concerns which is the firstmost idea behind the Solid technology. Which also introduces concerns when querying from different pods if all data gets pulled to the same physical location and is processed there. The architecture has a Web Agent for each Solid pod, which encapsulates the pod. Having an agent at each pod will then also allow for querying to happen physically at the pod and not at the place of querying, providing a way to lighten the load of potential big queries regarding multiple pods as well. The architecture also allows for Aggregator Web Agents which then provide a way of having materialized views available which query multiple pods at the same time so trusted partners can access these materialized views at the aggregator agent instead of having to query all pods each time.

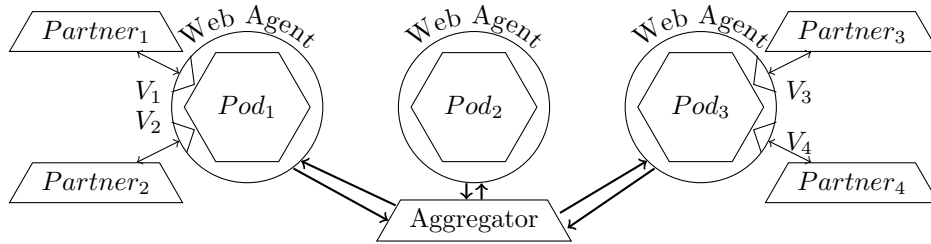


Figure 1: Three pods together with their web agents which handle which trusted partners, such as the aggregator in the illustration, can access which parts of the data. Web Agents provide views  $V_i$  which trusted partners can utilize.

## Incremental view maintenance

This architecture on its turn leans very well into the optimizing of the querying of Solid pods and other linked data technologies. With the Web Agent it allows for a way of storing materialized views built upon the data of the respective Solid pod. When data in one of the pods get updated, this get incrementally updated to the view in the Web Agent, this allows for trusted partners to retrieve the already calculated

---

<sup>\*</sup>Data Science Institute, UHasselt

<sup>†</sup>VITO

materialized view without having calculate the query each time it gets fetched. This view is built from a query which is typically accessed multiple times by multiple partners.

## References

- [1] Andrei Vlad Sambra, Essam Mansour, Sandro Hawke, Maged Zereba, Nicola Greco, Abdurrahman Ghanem, Dmitri Zagidulin, Ashraf Aboulnaga, and Tim Berners-Lee. Solid: a platform for decentralized social applications based on linked data. *MIT CSAIL & Qatar Computing Research Institute, Tech. Rep.*, 2016.
- [2] The Solid Team. Solid project. <https://solidproject.org/>, 2024. Accessed: 2024-04-05.