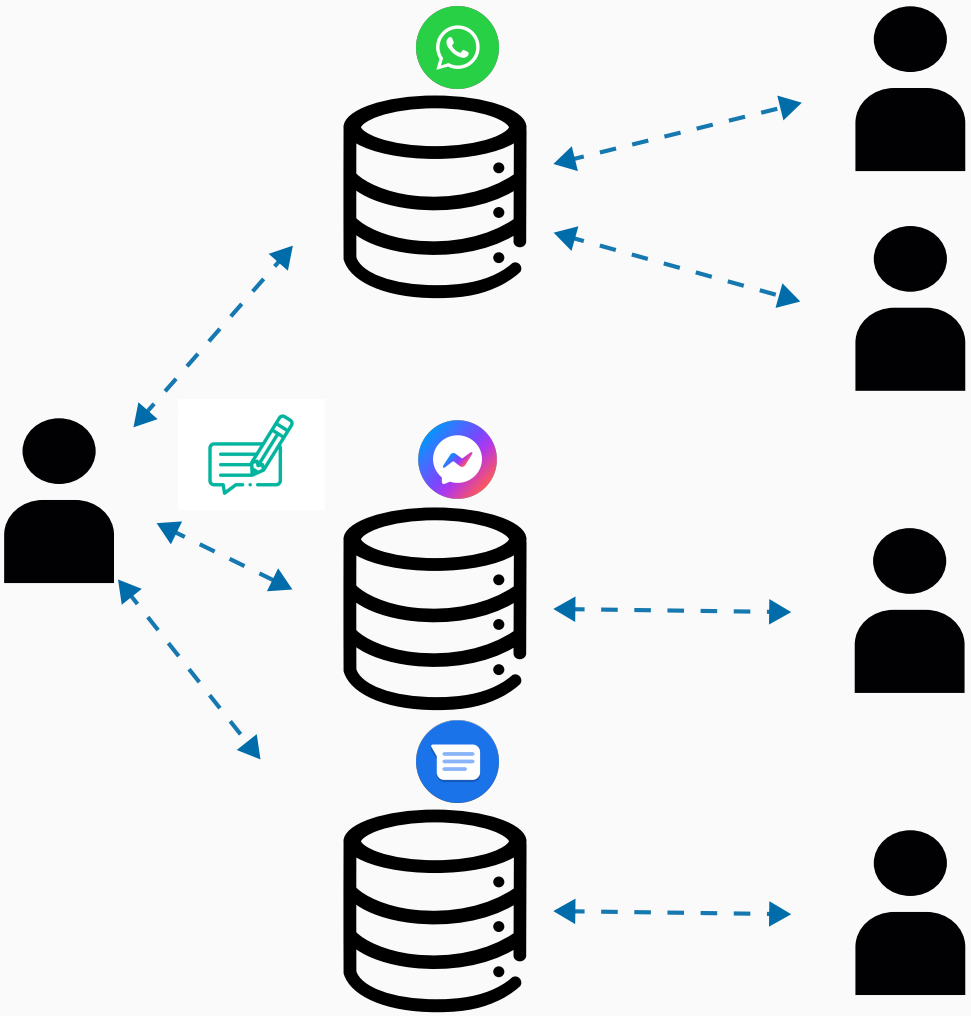
Revisiting Link Prioritization for Efficient Traversal in Structured Decentralized Environments

Ruben Eschauzier, Ruben Taelman, Ruben Verborgh

October 9, 2025

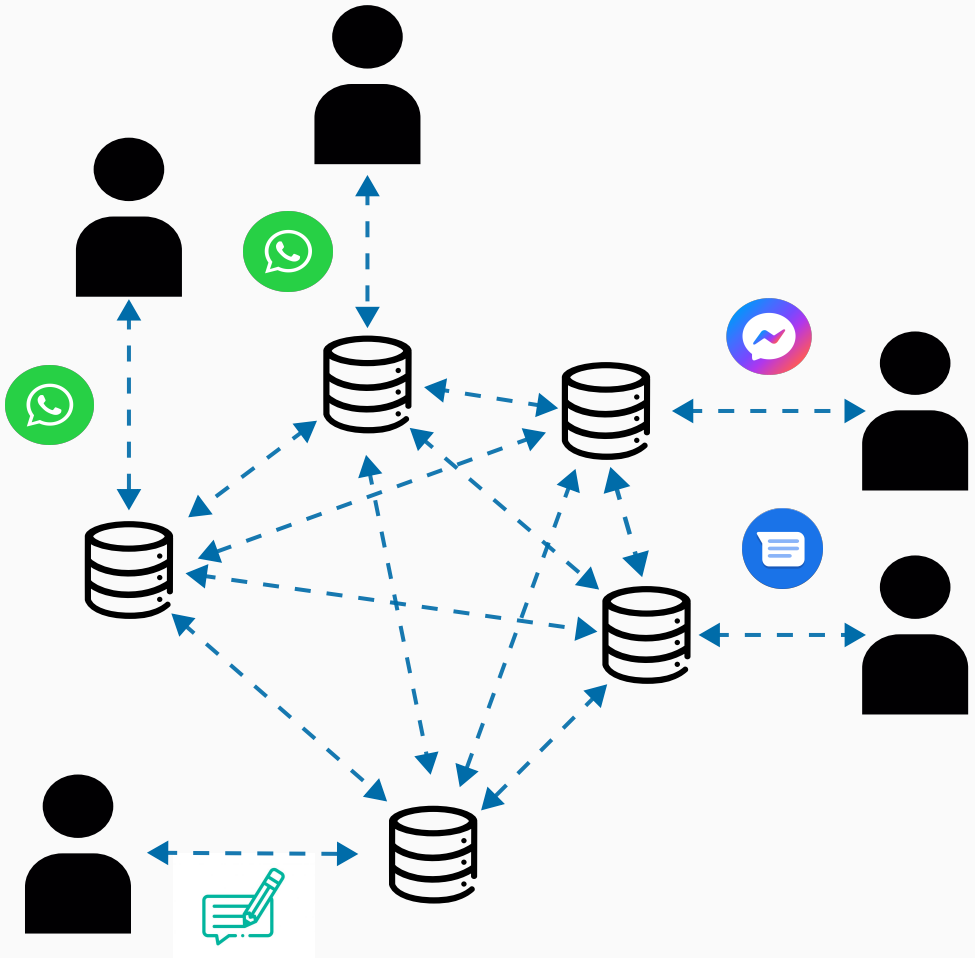
Department of Electronics and Information Systems, Ghent University – Imec

The Need for Decentralized Personal Data Storage



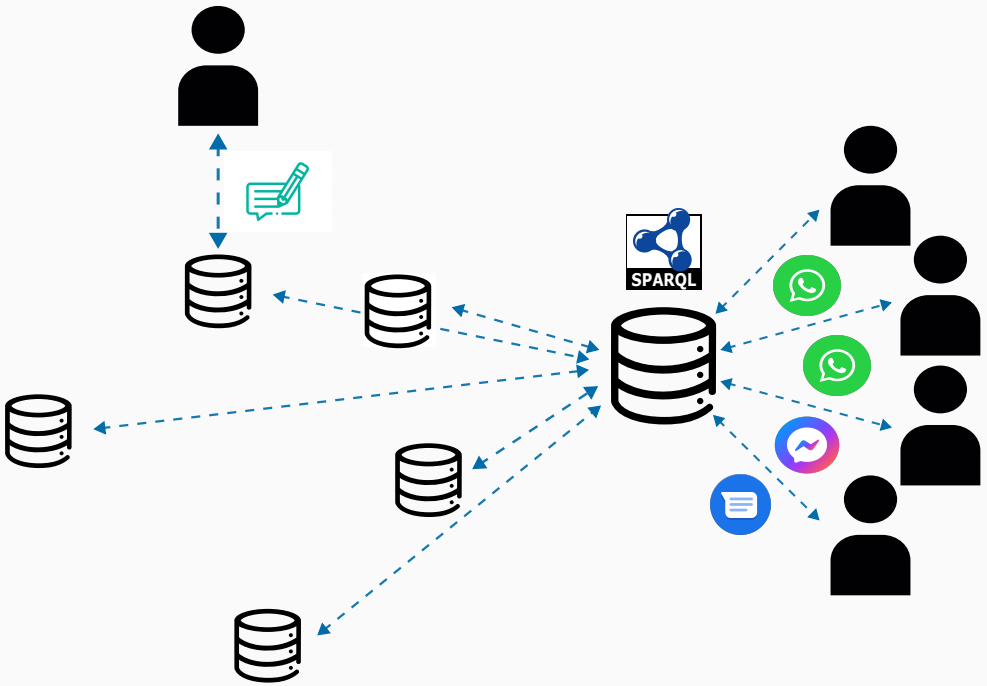
- Each application has its own data
- Stiffles innovation
- Causes vendor lock-in

The Need for Decentralized Personal Data Storage



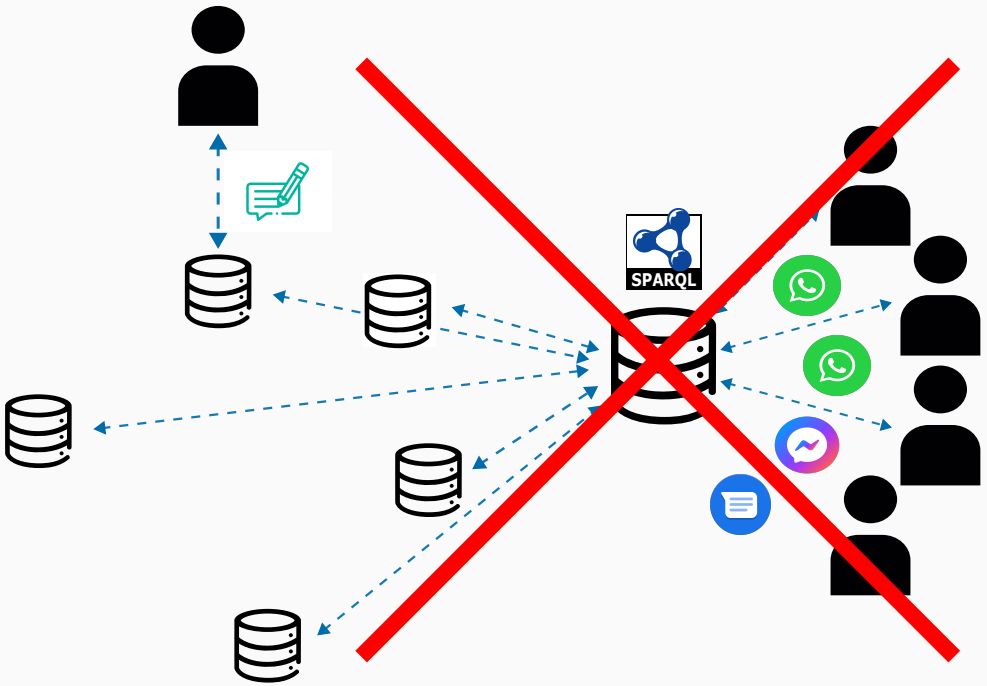
- Each application uses common data storage
- Easy to switch vendors
- Promotes innovation

The Problem with Centrally Aggregating and Querying



- Why not aggregate data and query it?
- Impossible in case of personal data due to privacy concerns

The Problem with Centrally Aggregating and Querying

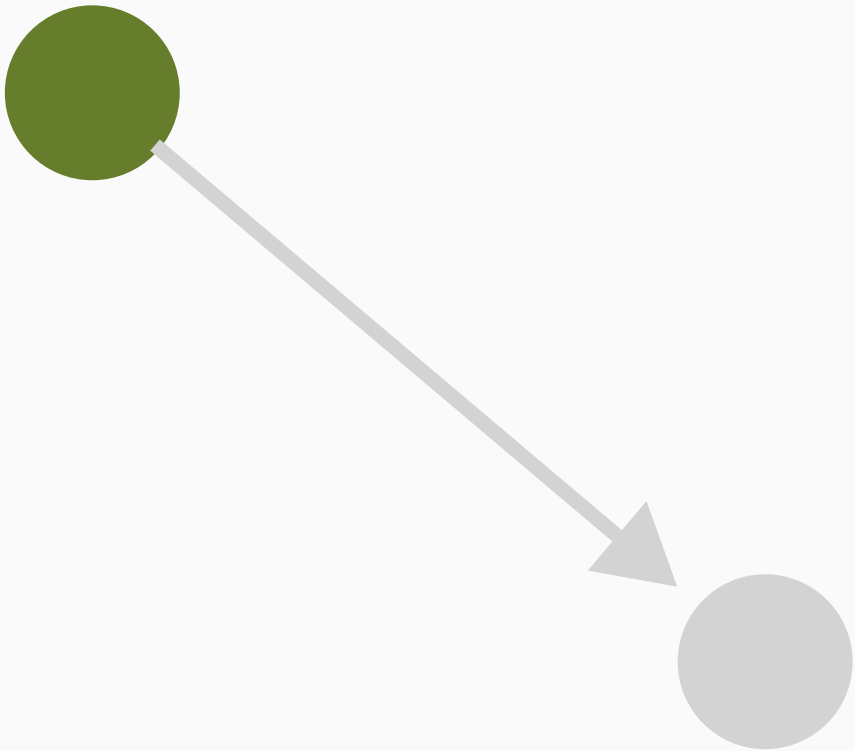


- Why not aggregate data and query it?
- Impossible in case of personal data due to privacy concerns

Link Traversal-based Query Processing

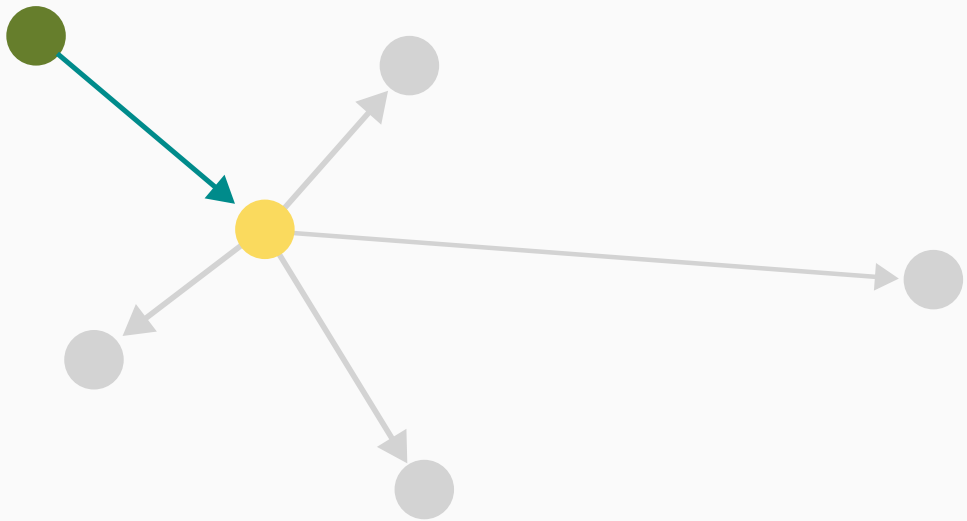
- Link Traversal iteratively dereferences data to query over
- Continuously produces results
- Can enforce fine-grained (document-level) access-control

Link Traversal: Seed Document



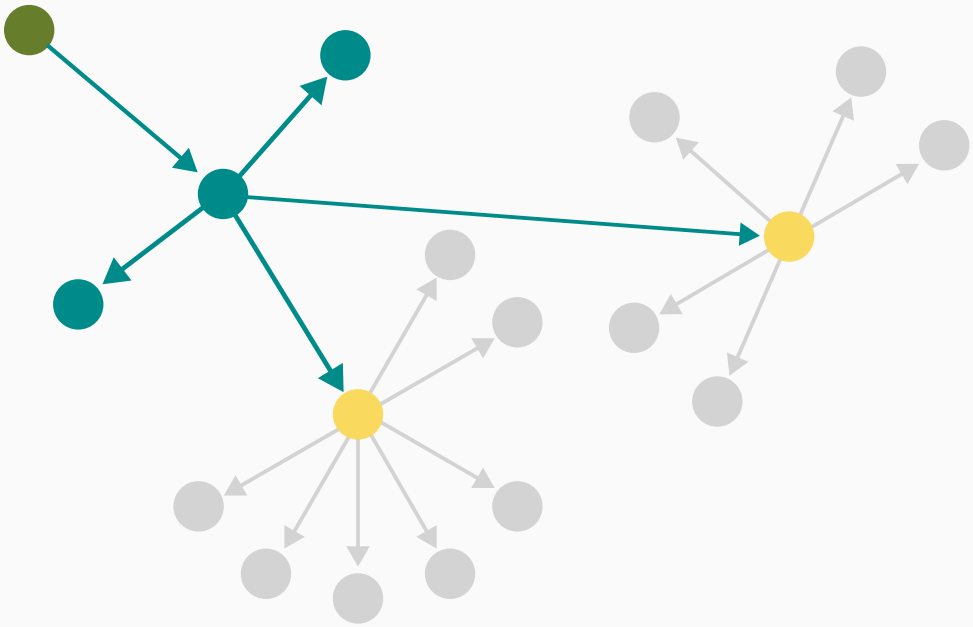
- Link Traversal starts from seed documents (URIs)
- These are provided by the user or in the query.

Link Traversal: Traversal



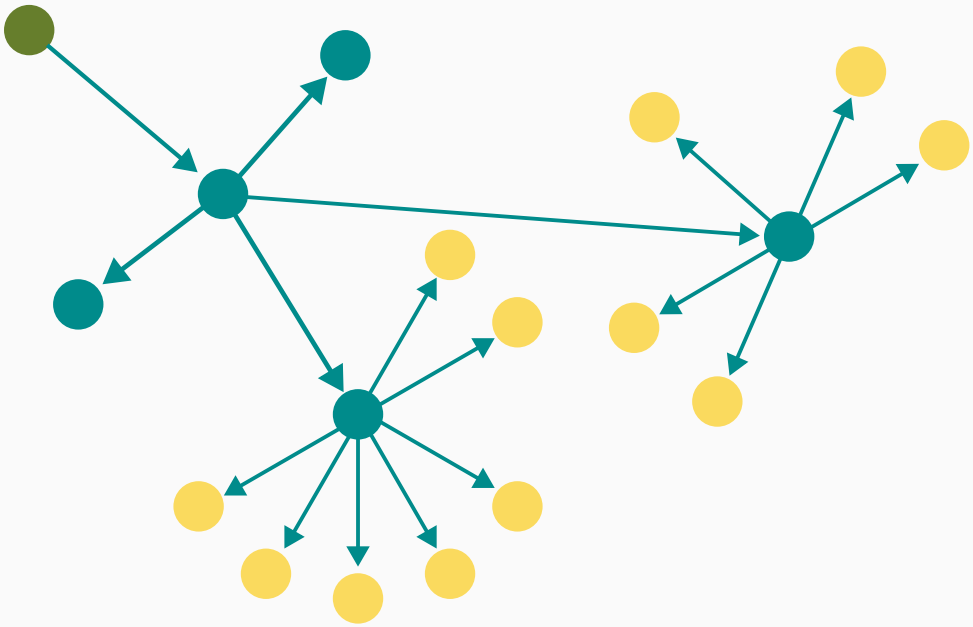
- New URIs are extracted from the seed document
- URIs are extracted in accordance with reachability criterions

Link Traversal: Traversal



- New URIs are dereferenced and the process is repeated

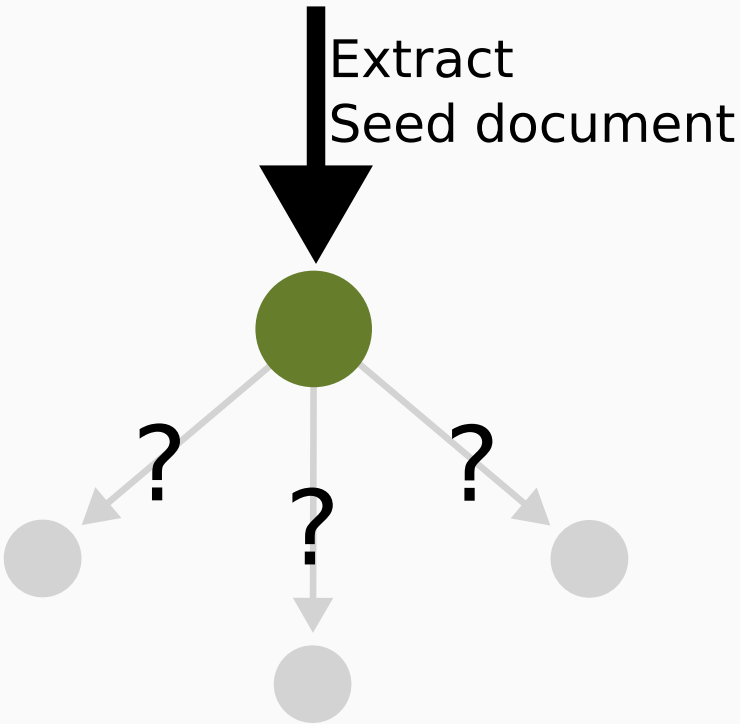
Link Traversal: Termination



- This continues untill all links are dereferenced

Query Optimization for Link Traversal

```
SELECT * WHERE {
  <seedUri> <ex:p1> ?o1.
  <seedUri> <ex:p2> ?o2.
  ...
}
```



The query (partly) determines:

- The queried data
- The topology of the queried data
- The query-relevant documents

Result: limited prior knowledge for query optimization