On the *Social* Aspect of SoLiD
探索SoLiD的社交与社会技术属性

2024-01-11
赵睿

中文版

rui.zhao@cs.ox.ac.uk

Contents

- Background
    - About me and EWADA, in relation to SoLiD
- On the *Social* aspect of SoLiD
- Summary and future vision

Background

About me and EWADA,
in relation to SoLiD

How I got to know SoLiD

1. When exploring different decentralized/federated systems many years ago
    - Others including Mastodon, GNU Social, Friendica, HubZilla, etc
    - They serve different purposes:
        - Mastodon (etc) are dedicated social media systems
            - Each system stores and manages its own data
        - SoLiD is a (general) federated Personal Data Store (PDS)
            - It expects data from all Apps
2. After joining EWADA project
    - Exploring the user, autonomy and social aspects of SoLiD

EWADA and SoLiD

- EWADA: Ethical Web And Data Architecture in the Age of AI
  - 粗译：在AI时代中有道德的网络（万维网）和数据架构
- Led by Prof. Sir Nigel Shadbolt & Prof. Sir Tim Berners-Lee
- SoLiD as context and experimental ground

UNIVERSITY OF OXFORD

OXFORD MARTIN SCHOOL

https://ewada.ox.ac.uk/

SoLiD explorations by EWADA team

- Solidflix https://github.com/OxfordHCC/solid-media/
  - Movie watching tracking and rating system + Movie recommender
  - Data in everyone's Pods, not in a central platform (e.g. Douban, Netflix)
- KNoodle (+Orchestrator) https://github.com/oxfordhcc/calendar-orchestrator
  - Calendar information, and meeting arrangement
  - Similar to Doodle, but decentralized
- Libertas https://github.com/OxfordHCC/libertas
  - Collective privacy-protecting data usage in SoLiD-like contexts with user autonomy
  - **Effective** use of MPC in SoLiD-like contexts
- DToU (Data Terms of Use)
  - Attempt to improve *the Biggest Lie on the Internet*: "I have read and agree to the Terms of Service"
  - Facilitating user autonomy in decentralized contexts
  - Formal modelling user preferences and application needs, and performing automated reasoning
- Solid + smart band
- Solid + children

Explorations of the Social aspect of SoLiD

SoLiD, Mechanisms, Social

- SoLiD: Social Linked Data
- Linked Data (LD), or RDF, as the main data type / model
  - LD is from Semantic Web technologies, but emphasising on data modelling and reasoning
- WebID
- Access Control
  - WAC, the mainstream access control standard
    - Read, Write, Append, Control; Individual (WebID), Origin, Ground, Public
  - ACP is planned as the next generation
    - More patterns; supporting Verifiable Credential
- Inbox

- Social: Data, Sharing (Authorization), Response, Linkage

Explorations of the Social aspect of SoLiD

Existing attempts:

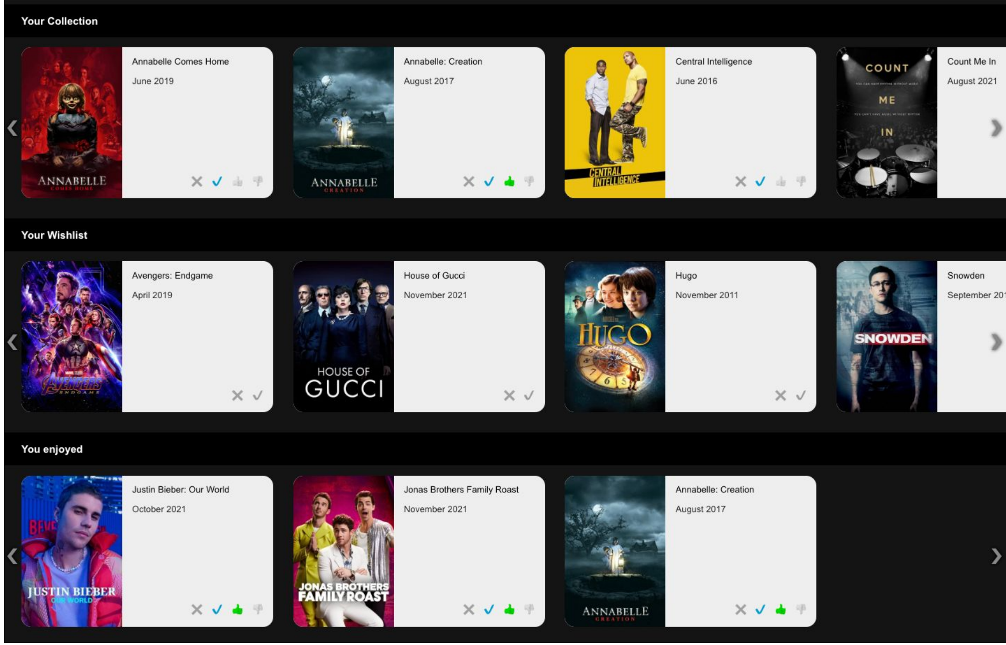
- Instant Messaging
  - Solid Chat、Liquid Chat
- Micro-blogging
  - ActivityPods (ActivityPub protocol)
- Collaborative video watching
  - BBC Watch Party
- Digital twins + Group Management + Intelligent PDF indexing
  - GraphMetrix
- Citizen Documents / Certificates
  - My Citizen Profile (Belgium Flanders)
- Medical data
  - Manchester NHS, XFORM

Our exploration:

- Movie sharing + Privacy-friendly recommendation
  - Solidflix
- External sync + Multi-user async collaboration (including offline)
  - KNoodle + Orchestrator
- Collective computation + data protection + user autonomy
  - Libertas

Solidflix

- Movie-watching library
  - Recording
  - Rating
  - Friends’ libraries
- Personalized Movie recommendati
  - Content-based recommender
  - Collaborative filtering



Your Collection

Annabelle Comes Home
June 2019

Annabelle: Creation
August 2017

Central Intelligence
June 2016

Count Me In
August 2021

Your Wishlist

Avengers: Endgame
April 2019

House of Gucci
November 2021

Hugo
November 2011

Snowden
September 20

You enjoyed

Justin Bieber: Our World
October 2021

Jonas Brothers Family Roast
November 2021

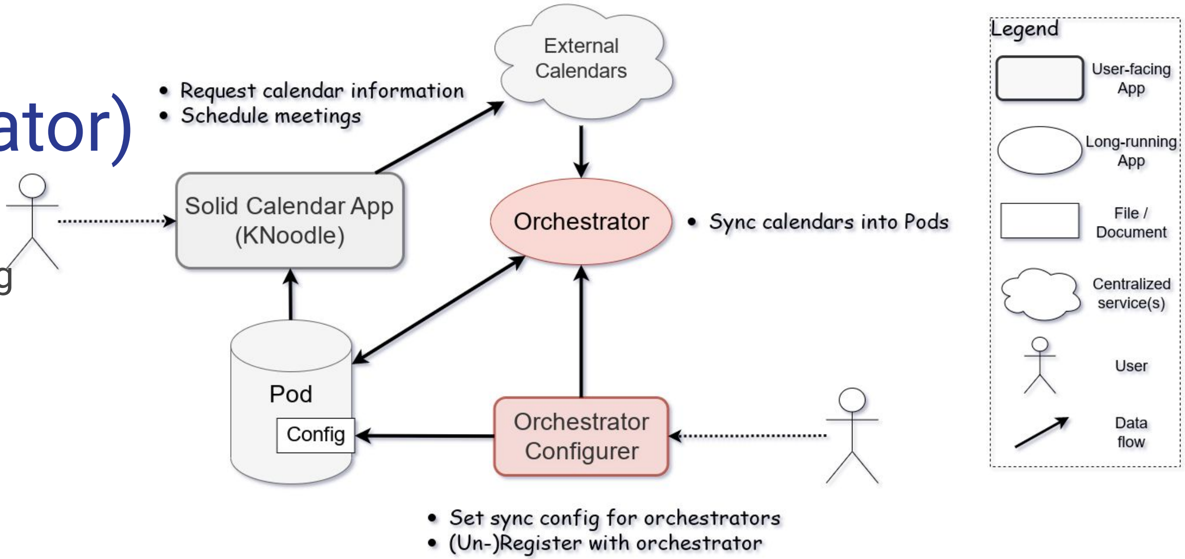
Annabelle: Creation
August 2017

Considerations:

- Permission control?
- Privacy-friendly recommender?

KNoodle (+Orchestrator)

- Calendar (busy/free) state sharing
- Handy meeting scheduling
- External calendar



- Request calendar information
- Schedule meetings

- Sync calendars into Pods

- Set sync config for orchestrators
- (Un-)Register with orchestrator

Considerations:

- Synchronization?
- Offline users?

- Introducing Orchestrator
    - Long-running mini-service
    - Limited to specific functionalities (calendar importing)
    - Users to “register (interest)” to use

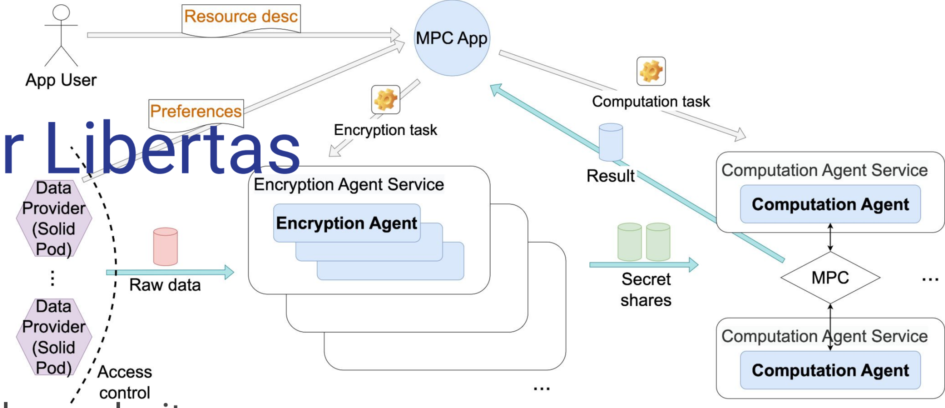
Experience from KNoodle + Orchestrator

- Orchestrator can solve business-layer data maintenance
- SoLiD’s authentication suffices basic requirements
  - Solid-OIDC can verify the authenticity of a *delegated* agent
    - E.g. When the Orchestrator-configuration App instructs Orchestrator to perform actions
- Orchestrator can be generalized
  - https://mellonscholarlycommunication.github.io/spec-orchestrator/
    - E.g. Trigger-based
- Potential risk of explosion of Orchestrator functionalities
  - Another form of centralization?

Libertas

- PDS (SoLiD Pods) + Collective/Crowd + Data(-secret)-protected computation
- MPC (Secure Multi-Party Computation) provides security, but not (decentralized) trust
    - MPC running on two servers deployed by the same company may not be more secure than not using MPC on a single server
- Libertas fills the gap:
    - Decentralized trust settings
        - Set once, Reuse forever
    - No sacrifice of autonomy
        - Computation requestor(s) do not need to repeatedly disturb data providers
        - Data providers can revoke trust and permission from their Pods, without informing computation requestors
    - Resource consumption scales linearly with increment of SoLiD Pods
    - Generic computation, not just for a specific type of tasks
        - E.g. We evaluated Differential Privacy on Libertas, providing input- and output-privacy

Challenges and Solutions for Libertas



- Significant number of data providers
  - Use *delegated*-decentralized computation
  - Naively applying MPC results in polynomial complexity
- No computation capability in SoLiD Pods
  - Use *Agents*
    - *Encryption Agent* for data secret-sharing
    - *Computation Agent* for further MPC with these secretly-shared data
  - Data providers perform (de-)authorization
- Diverse trusts
  - Use *preference document*
  - Set permissions
  - Use Agent-selection algorithm

Experience from Libertas

- It is possible to perform collective secrecy-protecting / privacy-preserving computation in decentralized contexts
- It is possible to respect individual users' autonomy in the meantime
- It is possible to be compatible with existing (SoLiD) protocols
    - By introducing additional nodes and doing user authorization

- Maybe SoLiD Pod can be extended with compassion capacities
    - To replace Encryption Agents?
    - To replace Computation Agents?
- Agent selection algorithm affects security assumptions
    - Semi-honest honest-majority, in the best case
    - Malicious dishonest-majority, in the worst case (with very high probability)

Summary and future vision

Possibilities with SoLiD

- ✓ Flexibility
  - ○ E.g. Single-WebID Multiple-Pods
- ✓ Extensibility
- ✓ Personal data and personal Apps
- ✓ Cross-App data sharing and interoperability
  - ○ RDF + Type Index
- ✓ Team/Friend data-sharing
- ✓ Notification, inbox
- ✓ Authentication of *delegated* agents
- ✓ Social sharing and social network
- ✓ Business-layer data maintenance (using Orchestrator or alike, or if Pod service supports extension)
- ✓ Effective collective data usage with security/privacy protection

Experience and observations

- Business logic of decentralized and centralized Apps can be different
  - E.g. Centralized and decentralized recommendation algorithms
- Emphasis on user autonomy
  - E.g. Meeting scheduling and syncing; Libertas
- Natural self-constraint / conservation of App’s data access
  - App cannot assume access to all data from everyone, so its functionality would not depend on that
- A pity for Pods without computation capacity in certain scenarios
  - Workarounds available

Future vision of SoLiD

- More powerful generic authorization / permission mechanism
  - DToU? Agent? Negotiation?
- Data Views
  - E.g. Pod contains contact data; when using *Contact* App, providing all information; when using *Birthday Reminder* App, providing birthday and name only
- Computation capacity (or similar) in Pods
  - Triggers, delegated computation, transferred computation
  - Virtual documents

Thanks for listening!

https://ewada.ox.ac.uk
https://me.ryey.icu
rui.zhao@cs.ox.ac.uk