Hello, my name is Noel

- Developer & Entrepreneur.
- 4-day workweek @ Moodle.
- Making Solid Apps on the side.
- Working in the Open.

noeldemartin.com | noeldemartin.social | @NoelDeMartin
What is Solid?

- Created by Tim Berners-Lee.
- Technologies:
  - Web Browsers
  - HTTP
  - HTML
The Semantic Web (2001)

• An extension to the original web with Linked Data.
• Technologies:
  – RDF
  – JSON-LD, Turtle, etc.
The Solid Protocol (2016)
solidproject.org

• Brings decentralized storage to the web.
• Web 3.0, by the creators of the web.
• Not to be confused with Web3.
  “Web3 is not the web at all”
  – Tim Berners-Lee @ Web Summit 2022
The Solid Protocol (2016)

- Apps & Services store data in your POD. **Personal Online Datastore**
The Solid Protocol (2016)

**centralized:** single app & back-end
- backend X
- specific to X
- app X
- specific to Y
- backend Y

**decentralized:** multiple apps & back-ends
- data pod
- app X
- data pod
- app Y
- data pod

Ref: Ruben Verborgh @ FOSDEM 2019
From Zero to Hero

Ref: Superman

noeldemartin.com/fosdem
From Zero to Hero?

Ref: Superlópez
About this talk

• There are infinite ways to use Solid, just like the web.
• We’ll focus on Solid Apps.
• Lessons learned from 4 years of side-projects.
• Broad strokes, but we’ll get into the weeds.
Solid Focus

November 2018 – August 2019
Create list

- Go shopping
- Give FOSDEM talk
- Prepare FOSDEM talk
Solid 101 – Authentication
Solid 101 – Authentication
Solid OIDC (OpenID Connect)

1. Token + Identity

2. Data exchange

App

Identity Provider

POD
import { login, handleIncomingRedirect } from '@inrupt/solid-client-authn-browser';

// When you log in...
login({
  oидcIssuer: 'https://pod.alice.com',
  redirectUrl: window.location.href,
});

// After redirect...
const session = await handleIncomingRedirect();

alert(`Hello, ${session.webId}!`);
Solid 101 – RDF

Plain JSON (not RDF)

{
  "id": 1,
  "description": "Go shopping",
  "done": false
}
{"@context": "https://schema.org/",
"@id": "https://pod.alice.com/tasks/1#it",
"@type": "Action",
"description": "Go shopping",
"actionStatus": {
    "@id": "https://schema.org/PotentialActionStatus"
}
Solid 101 – RDF

JSON-LD

Vocabulary / Ontology: describes what the properties mean

```json
{
    "@context": "https://schema.org/",
    "@id": "https://pod.alice.com/tasks/1#it",
    "@type": "Action",
    "description": "Go shopping",
    "actionStatus": {
        "@id": "https://schema.org/PotentialActionStatus"
    }
}
```
Solid 101 – RDF

JSON-LD

Id: URIs (usually document URLs)

```json
{
    "@context": "https://schema.org/",
    "@id": "https://pod.alice.com/tasks/1#it",
    "@type": "Action",
    "description": "Go shopping",
    "actionStatus": {
        "@id": "https://schema.org/PotentialActionStatus"
    }
}
```
### JSON-LD

```
{
   "@context": "https://schema.org/",
   "@id": "https://pod.alice.com/tasks/1#it",
   "@type": "Action",
   "description": "Go shopping",
   "actionStatus": {
      "@id": "https://schema.org/PotentialActionStatus"
   }
}
```
Some properties have literals values (string, number, etc.)

```json
{
   "@context": "https://schema.org/",
   "@id": "https://pod.alice.com/tasks/1#it",
   "@type": "Action",
   "description": "Go shopping",
   "actionStatus": {
      "@id": "https://schema.org/PotentialActionStatus"
   }
}
```
Other properties reference other resources

```json
{
  "@context": "https://schema.org/",
  "@id": "https://pod.alice.com/tasks/1#it",
  "@type": "Action",
  "description": "Go shopping",
  "actionStatus": {
    "@id": "https://schema.org/PotentialActionStatus"
  }
}
```
Solid 101 – RDF

Turtle

@prefix schema: <https://schema.org/> .

<#it>
  a schema:Action ;
  schema:description "Go shopping" ;

noeldemartin.com/fosdem
Solid 101 – CRUD

Solid POD

Container

RDF Document

RDF Resource

Binary (image, video, etc.)

...
Solid 101 – CRUD

https://pod.alice.com/tasks/1#it

- Solid POD
- Container
- RDF Document
- RDF Resource
Solid 101 – CRUD

Just use HTTP verbs!

- Create one task → POST https://pod.alice.com/tasks/1

```
@prefix schema: <https://schema.org/> .

<#it>
  a schema:Action ;
  schema:description "Go shopping" ;
```
Solid 101 – CRUD

Just use HTTP verbs!

• Create one task → POST  https://pod.alice.com/tasks/1
• Get one task → GET  https://pod.alice.com/tasks/1
• Delete one task → DELETE  https://pod.alice.com/tasks/1
• Get list of all tasks → GET  https://pod.alice.com/tasks/
• ...
You just built your first Solid App!
Learn more

- Introduction to Solid (FOSDEM 2019).
- Solid Hello World: With a task manager!
- Other specs:
  - RDF Primer
  - RDF Schema
  - Turtle
  - LDP Primer (containers/documents)
Read the full story

• Implementing a Task Manager using Solid
• Improving Solid Focus Task Manager
• Working on Solid Focus Task Manager
Takeaways

• Learned Solid basics.

• No server headaches.
  Because the app lives in the frontend.
Challenges

• Onboarding UX.
• Page speed.
• Interoperability.
  – How do I structure my data (which vocabulary)?
  – Where do I store my data (which container)?
Media Kraken

January 2020 – July 2020
Onboarding UX

Media Kraken
Keep track of your movies and create your own collection!

Where do you want to store your data?
- Use Solid
- Use browser storage

Help me decide what to use
Onboarding UX

Collection (36)

Import Movies

You can import movies from the following sources:

- IMDb
- JSON-LD
- TVISO
- Good Films

If you can't import your movies with any of these options, please let me know and I'll help you.

In the meantime, you can probably find your movies using the search in the header.
Page Speed

Movies container

GET /movies/

Movie #1
GET /movies/1

Movie #2
GET /movies/2

Movie #3
GET /movies/3

Movie #4
GET /movies/4

Movie #5
GET /movies/5

Movie #6
GET /movies/6

Movie #7
GET /movies/7

...

Total HTTP requests: N+1 😲
Page Speed

- Movies cache (IndexedDB).
- GET only updated movies. using http://purl.org/dc/terms/modified
- Still not great for big collections.

Loading movies metadata... 😄
Interoperability

• How do I structure my data (which vocabulary)?
• Where do I store my data (which container)?

PS: Interoperable Serendipity
How do I structure my data?

• Use an existing vocabulary.  
  https://lov.linkeddata.es
How do I structure my data?

- Use an existing vocabulary. [https://lov.linkeddata.es](https://lov.linkeddata.es)
- Create your own vocabulary.
- Mix and match.
  - See “Bag of Chips” ([text](#), [video](#))
Where do I store my data?

Use the Type Index

https://alice.com/#me

WebId document

- Name
- Avatar
- Public Type Index
- Private Type Index
- ...

Private Type Index

- Tasks container
- Movies container
- Recipes container
- ...

- https://pod.alice.com/movies/
- https://pod.bob.com/films/
- ...

noeldemartin.com/fosdem
Where do I store my data?

Use the Type Index

- If it doesn’t exist, just create it!
- Disclaimer: The [Type Index spec](#) is still a draft.
  
  Client-Client standard

- See also the [Solid Application Interoperability spec](#).
  
  Client-Server standard
Data sources

TMDB
https://themoviedb.org
Read the full story

- Implementing a Media Tracker using Solid
- Housekeeping
- Media Kraken @ Solid World (video)
Takeaways

• Type Indexes are nice.
• Caching is nice.
Challenges

• ~Onboarding UX.
• ~Page speed.
December 2020 – January 2023
Here's your cookbook

Avocado Quiche
Baba ganoush
Houmous
Pizza
Ramen
Vegan Shepherd's Pie
Onboarding UX

With *Umai* you'll be able to collect and share all your precious recipes, start now your culinary adventure!

How would you like to begin?

- Create your first recipe
- Connect your Solid POD

What is Solid?
Onboarding UX

It’s Offline First!

You can keep your data safe across devices with a **Solid account**, use the input below to log in.

```
https://
```

No thanks, I want to keep my data offline
Offline First – Authentication

- [Authenticating Offline-First Solid Apps](forum thread)
- TLDR:
  - Web Apps shouldn’t store Solid authentication tokens.
  - Logging in requires a browser redirect.
Offline First – Synchronization

Ramen is awesome
Offline First – Synchronization

T0: Recipe exists in both devices and Solid POD

**Mobile**

Ramen
Ramen is awesome
createdAt T0
updatedAt T0

**Solid POD**

Ramen
Ramen is awesome
createdAt T0
updatedAt T0

**Desktop**

Ramen
Ramen is awesome
createdAt T0
updatedAt T0
Offline First – Synchronization

T1: Recipe is updated in Mobile

Mobile

Ramen!!!
Ramen is awesome
createdAt T0
updatedAt T1

Solid POD

Ramen
Ramen is awesome
createdAt T0
updatedAt T0

Desktop

Ramen
Ramen is awesome
createdAt T0
updatedAt T0
Offline First – Synchronization

T2: Recipe is updated in Desktop

Mobile

Ramen!!!
Ramen is awesome

createdAt T0
updatedAt T1

Solid POD

Ramen
Ramen is awesome

createdAt T0
updatedAt T0

Desktop

Ramen
Ramen is life

createdAt T0
updatedAt T2
Offline First – Synchronization

T3: Desktop synchronizes with the POD

**Mobile**
- **Ramen!!!**
  - Ramen is awesome
  - createdAt T0
  - updatedAt T1

**Solid POD**
- **Ramen**
  - Ramen is awesome
  - createdAt T0
  - updatedAt T0

**Desktop**
- **Ramen**
  - Ramen is life
  - createdAt T0
  - updatedAt T2

(no updates)
Offline First – Synchronization

T3: Desktop synchronizes with the POD

Mobile

Ramen!!!
Ramen is awesome
createdAt T0
updatedAt T1

Desktop

Ramen
Ramen is life
createdAt T0
updatedAt T2

Ramen
Ramen is life
createdAt T0
updatedAt T2

(1 update)
Offline First – Synchronization

T4: Mobile synchronizes with the POD

<table>
<thead>
<tr>
<th>Mobile</th>
<th>Solid POD</th>
<th>Desktop</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Ramen!!!</strong></td>
<td><strong>Ramen</strong></td>
<td><strong>Ramen</strong></td>
</tr>
<tr>
<td>Ramen is awesome</td>
<td>Ramen is life</td>
<td>Ramen is life</td>
</tr>
<tr>
<td>createdAt T0</td>
<td>createdAt T0</td>
<td>createdAt T0</td>
</tr>
<tr>
<td>updatedAt T1</td>
<td>updatedAt T0</td>
<td>updatedAt T2</td>
</tr>
</tbody>
</table>

(1 update)
Offline First – Synchronization

CRDTs

- CRDTs = Conflict-free Replicated Data Types.
- Request for Comments: CRDTish approach to Solid (forum thread)
- Custom vocab. https://vocab.noeldemartin.com/crdt/
Offline First – Synchronization

CRDTs

T0: Recipe exists in both devices and Solid POD

Mobile

Ramen
Ramen is awesome
createdAt T0
updatedAt T0

Solid POD

Ramen
Ramen is awesome
createdAt T0
updatedAt T0

Desktop

Ramen
Ramen is awesome
createdAt T0
updatedAt T0
Offline First – Synchronization

CRDTs

T1: Recipe is updated in Mobile

Mobile

Ramen!!!
Ramen is awesome

createdAt T0
updatedAt T1

... T1: update name

Solid POD

Ramen
Ramen is awesome

createdAt T0
updatedAt T0

Desktop

Ramen
Ramen is awesome

createdAt T0
updatedAt T0
Offline First – Synchronization

CRDTs

T2: Recipe is updated in Desktop

**Mobile**

**Ramen!!!**
Ramen is awesome

createdAt T0
updatedAt T1

T1: update name

**Solid POD**

**Ramen**
Ramen is awesome

createdAt T0
updatedAt T0

**Desktop**

**Ramen**
Ramen is life

createdAt T0
updatedAt T2

T2: update description
Offline First – Synchronization

CRDTs

T3: Desktop synchronizes with the POD

Mobile

Ramen!!!
Ramen is awesome
createdAt T0
updatedAt T1

T1: update name

Solid POD

Ramen
Ramen is awesome
createdAt T0
updatedAt T0

(no updates)

Desktop

Ramen
Ramen is life
createdAt T0
updatedAt T2

T2: update description
Offline First – Synchronization

CRDTs

T3: Desktop synchronizes with the POD

Mobile

Ramen!!!
Ramen is awesome
createdAt T0
updatedAt T1
---
T1: update name

Solid POD

Ramen
Ramen is life
createdAt T0
updatedAt T2
---
T2: update description

Desktop

Ramen
Ramen is life
createdAt T0
updatedAt T2
---
T2: update description

(1 update)
Offline First – Synchronization

CRDTs

T4: Mobile synchronizes with the POD

Mobile

Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

---

T1: update name
T2: update description

Solid POD

Ramen
Ramen is life

createdAt T0
updatedAt T2

---

T2: update description

Desktop

Ramen
Ramen is life

createdAt T0
updatedAt T2

---

T2: update description
Offline First – Synchronization

CRDTs

T4: Mobile synchronizes with the POD

**Mobile**

**Ramen!!!**
Ramen is life

createdAt T0
updatedAt T2

--
T1: update name
T2: update description

**(1 update)**

**Solid POD**

**Ramen!!!**
Ramen is life

createdAt T0
updatedAt T2

--
T1: update name
T2: update description

**Desktop**

**Ramen**
Ramen is life

createdAt T0
updatedAt T2

--
T2: update description
Offline First – Synchronization

CRDTs

T5: Desktop synchronizes with the POD

Mobile

Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

T1: update name
T2: update description

Solid POD

Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

T1: update name
T2: update description

Desktop

Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

T1: update name
T2: update description
Offline First – Synchronization

CRDTs

T5: Desktop synchronizes with the POD

Mobile

Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

--
T1: update name
T2: update description

Solid POD

Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

--
T1: update name
T2: update description

Desktop

Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

--
T1: update name
T2: update description

(no updates)
Offline First – Synchronization

CRDTs

T6: Everything is up to date!

Mobile
Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

--
T1: update name
T2: update description

Solid POD
Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

--
T1: update name
T2: update description

Desktop
Ramen!!!
Ramen is life

createdAt T0
updatedAt T2

--
T1: update name
T2: update description

noeldemartin.com/fosdem
Offline First – Synchronization

CRDTs
Data Sources

How do you want to create your recipe?

- Import from the Web
- Create from scratch
- Upload JsonLD file
Data Sources

The Web itself!

https://schema.org/Recipe
(Thanks, Semantic Web!)
Data Sources
The Web itself!
Data Sources

The Web itself!

• CORS :/
Sharing


⚠️ This recipe is private, remember to make it public before sharing the link with someone else.
Pizza

Ingredients

- 800g strong white bread flour
- 200g fine ground semolina flour or strong white bread flour
- 85g mozzarella cheese
- 2 x 7 g sachets of dried yeast
- 1 bunch of fresh basil
Sharing

• Visitors don’t need a Solid account.
• You can make your own Viewer App:
  – Cooking Assistant
  – Meal Planner
  – Shopping List
  – ...
Learn more

- offlinefirst.org
- Instant Loading: Building offline-first PWAs
- CRDTs for Mortals
- Local-first software
- WAC & ACP (Solid permissions)
Read the full story

• Implementing a Recipes Manager using Solid
Takeaways

● Offline First is the way!
● Sharing is caring.
● Keep It Simple.
Challenges

• Onboarding UX
  – What is Solid?
  – Where do I get a POD?
  – Why isn’t your app working with my POD?

• CORS
Thank you!

- Follow my work
  noeldemartin.com/now
- Check out my apps (all GPL!)
  noeldemartin.com/projects
- Join the community
  forum.solidproject.org

Ref: Superlópez