Solid Data Modules

Using Solid as a Platform for Data Portability

Michiel de Jong, software engineer and founder of Ponder Source.

APP DEVS SOLVE PROBLEMS

DATA DESIGN

IS PART OF

APP DESIGN

RIGHT?

“Single-app” Solid

=

RWW

Your webserver

machine read/writable

and linkable to others

HTML & RDF

“data browsers”

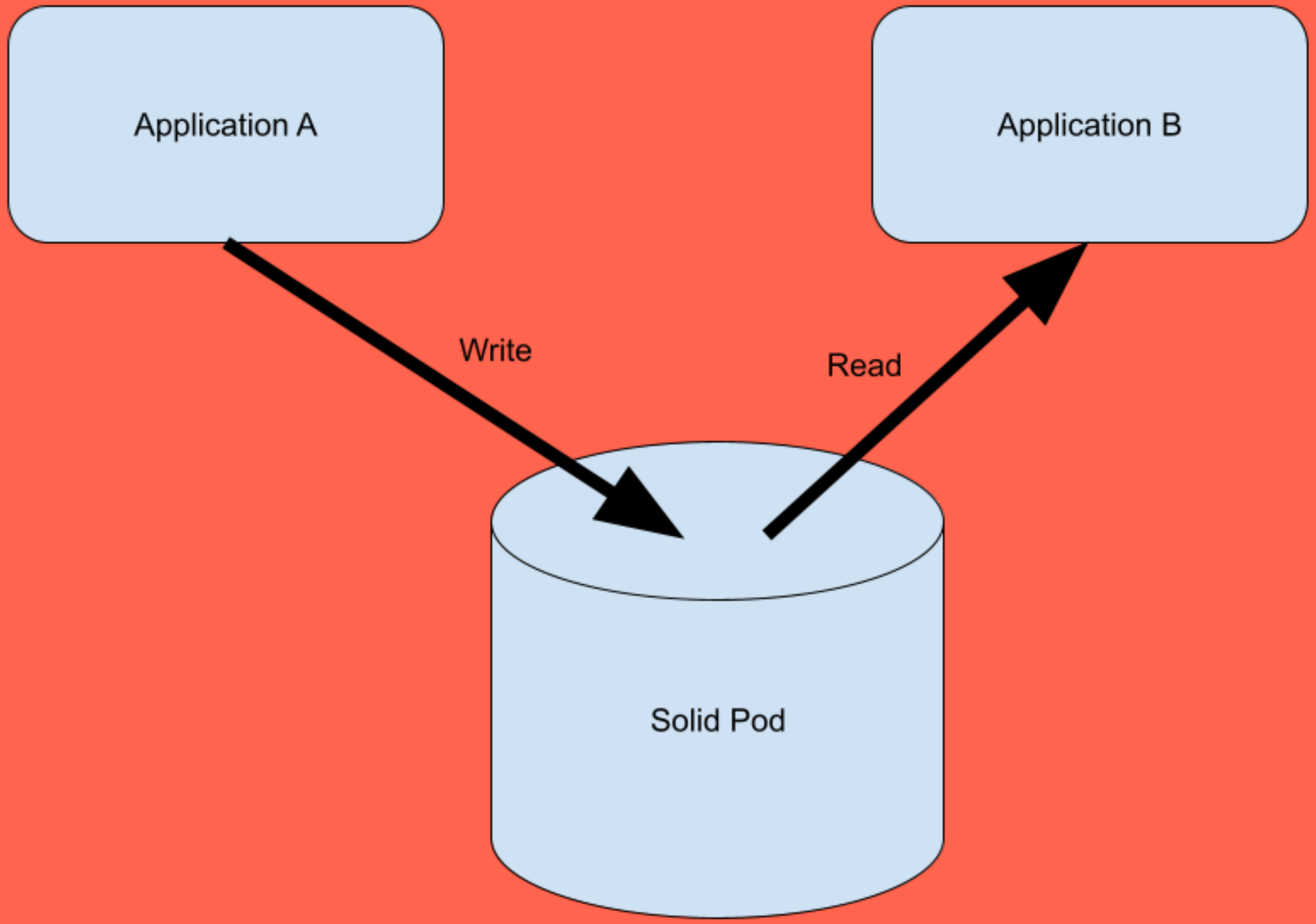
form generators

“Multi-app” Solid

=

Data pivot

THE POD AS A DATA PIVOT



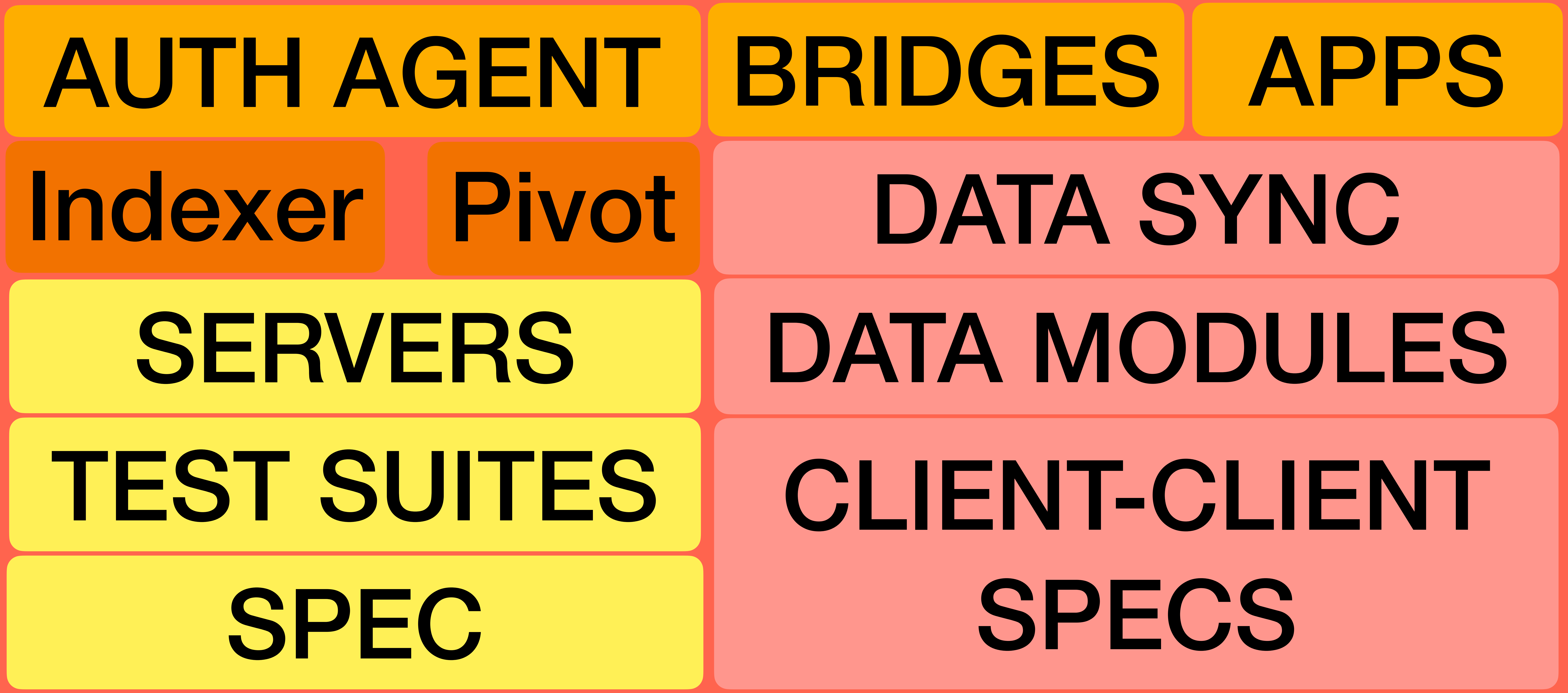
Interop + exit

=

Data Portability

PONDER SOURCE

MY SOLID ROADMAP



AUTH AGENT

BRIDGES

APPS

Indexer

Pivot

DATA SYNC

SERVERS

DATA MODULES

TEST SUITES

CLIENT-CLIENT SPECS

SPEC

PONDER SOURCE

Server Behaviour

DATA CONVENTIONS

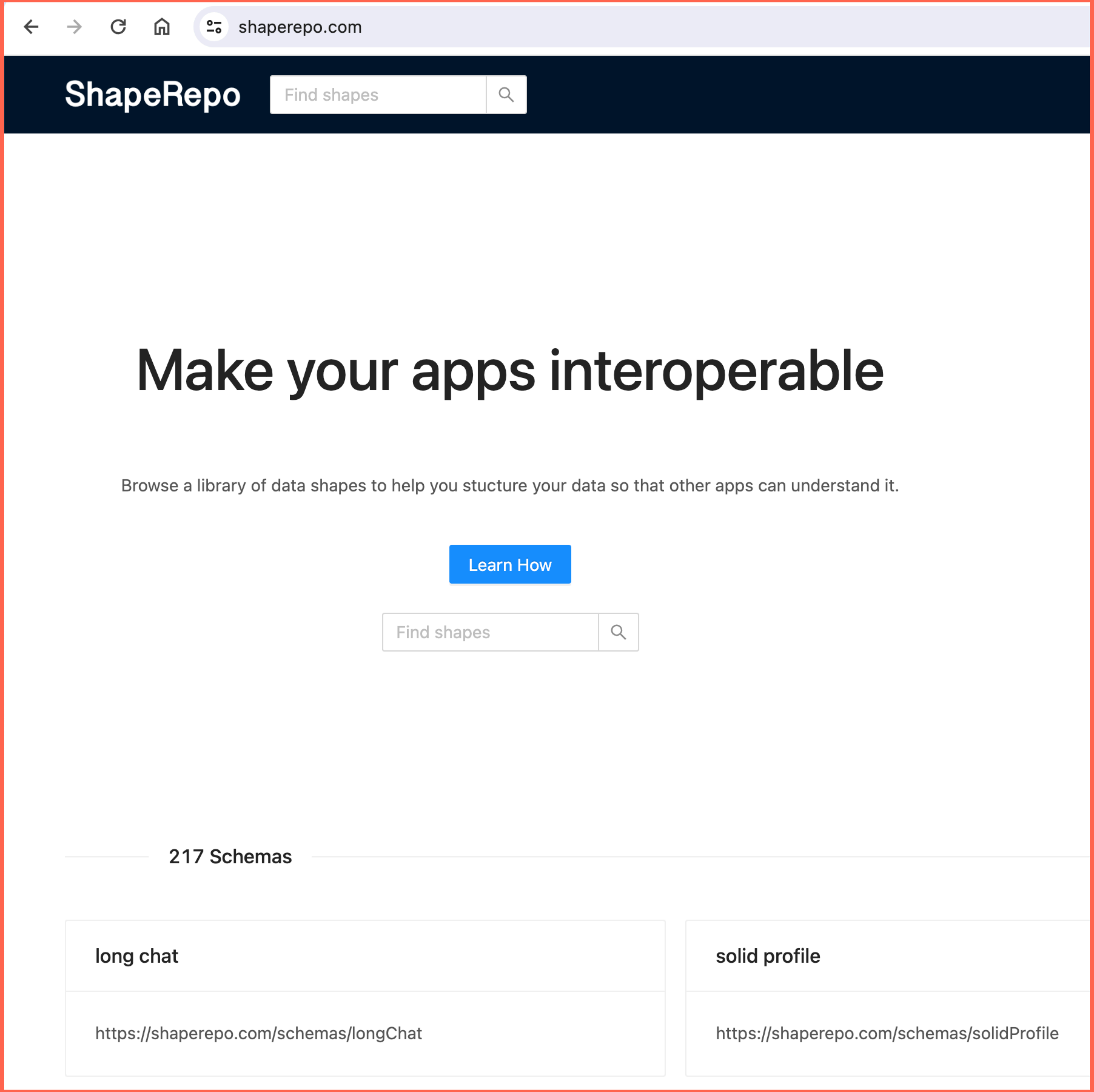
Access Control

PONDER SOURCE

Document

Translate

Reusable Code



shaperepo.com

ShapeRepo

Find shapes

Make your apps interoperable

Browse a library of data shapes to help you stucture your data so that other apps can understand it.

Learn How

Find shapes

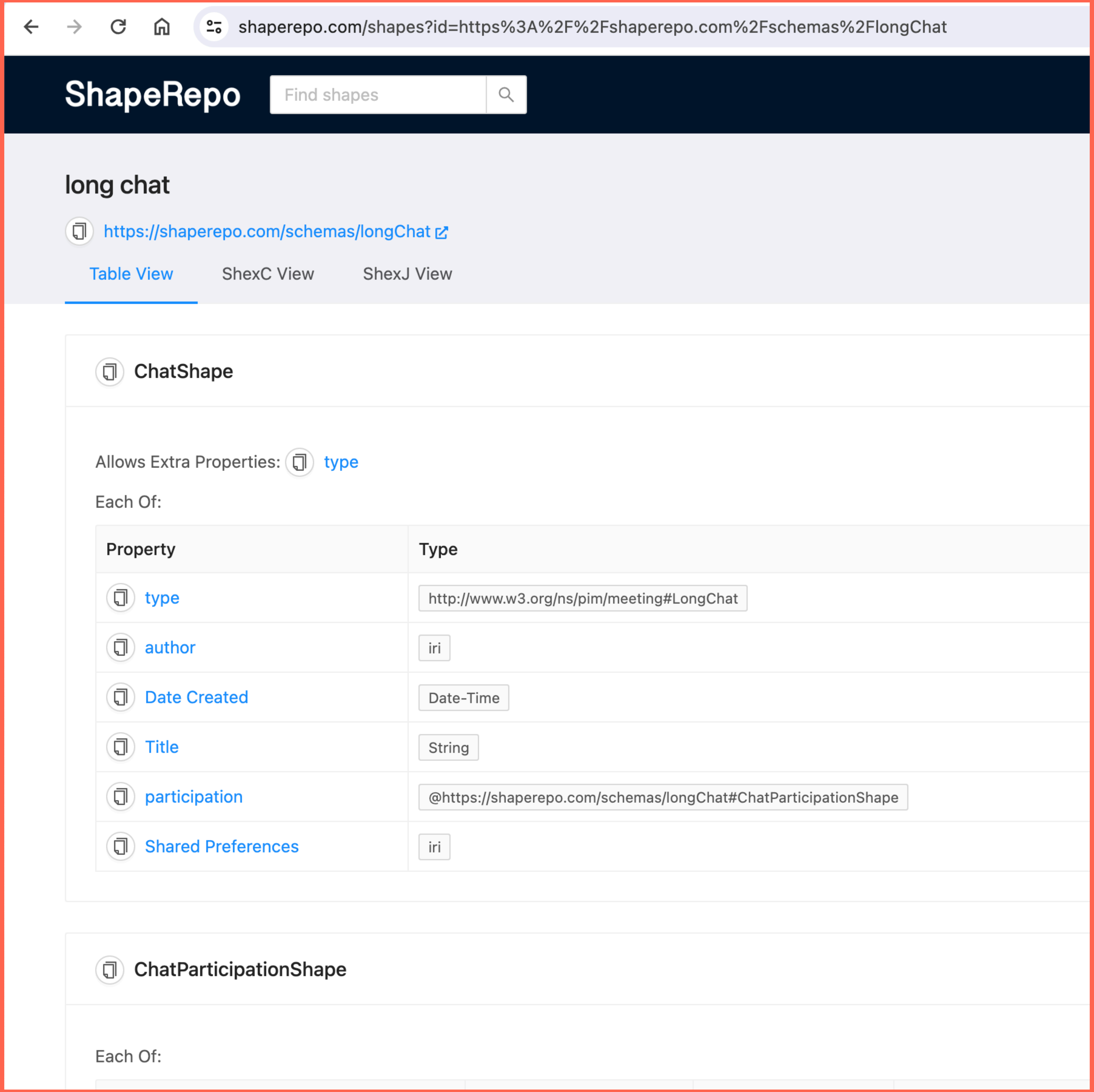
217 Schemas

long chat

https://shaperepo.com/schemas/longChat

solid profile

https://shaperepo.com/schemas/solidProfile



shaperepo.com/shapes?id=https%3A%2F%2Fshaperepo.com%2Fschemas%2FlongChat
ShapeRepo
Find shapes
Table View
ShexC View
ShexJ View

long chat

https://shaperepo.com/schemas/longChat

ChatShape

Allows Extra Properties: type

Each Of:

Property	Type
type	http://www.w3.org/ns/pim/meeting#LongChat
author	iri
Date Created	Date-Time
Title	String
participation	@https://shaperepo.com/schemas/longChat#ChatParticipationShape
Shared Preferences	iri

ChatParticipationShape

Each Of:

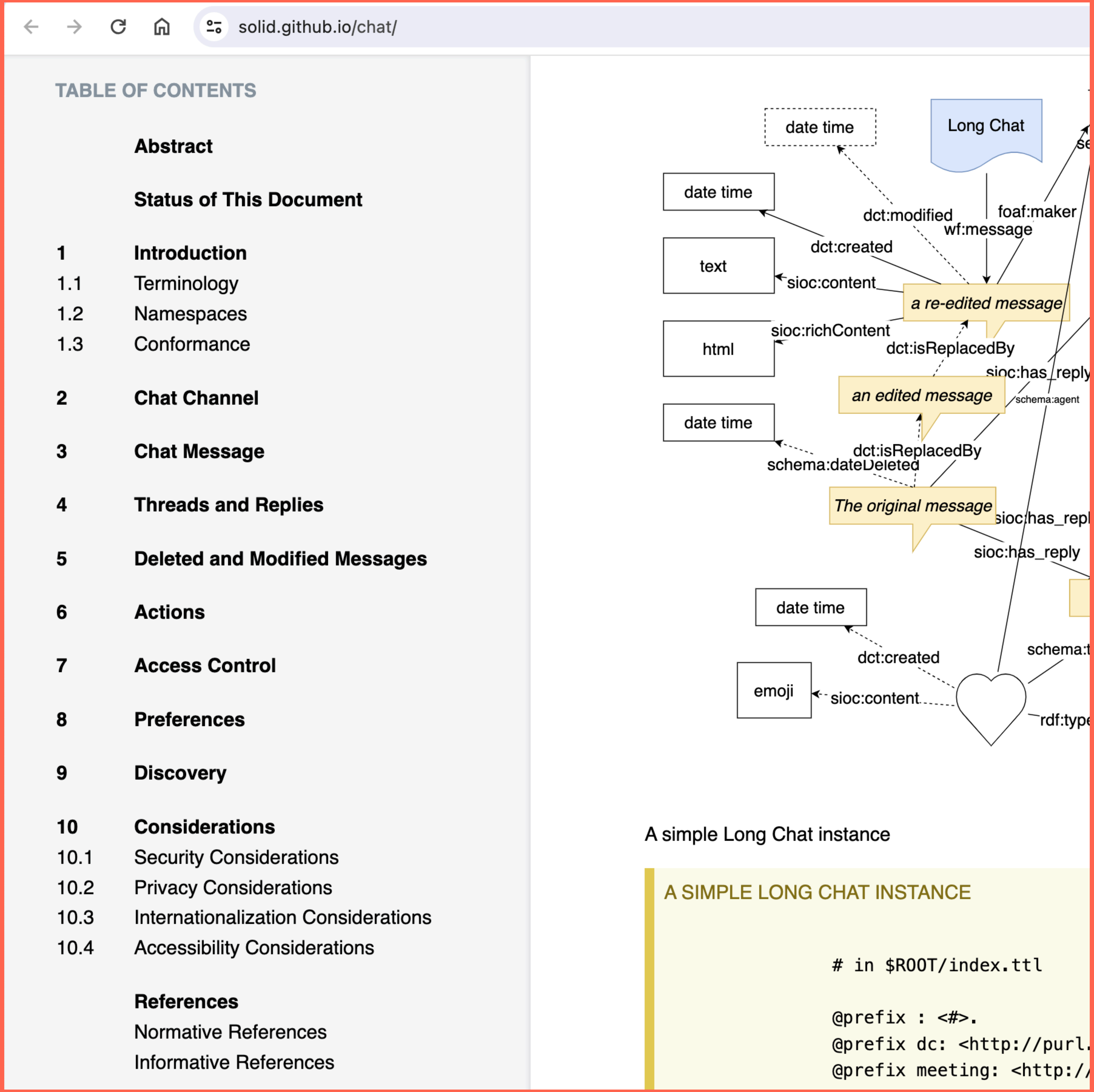
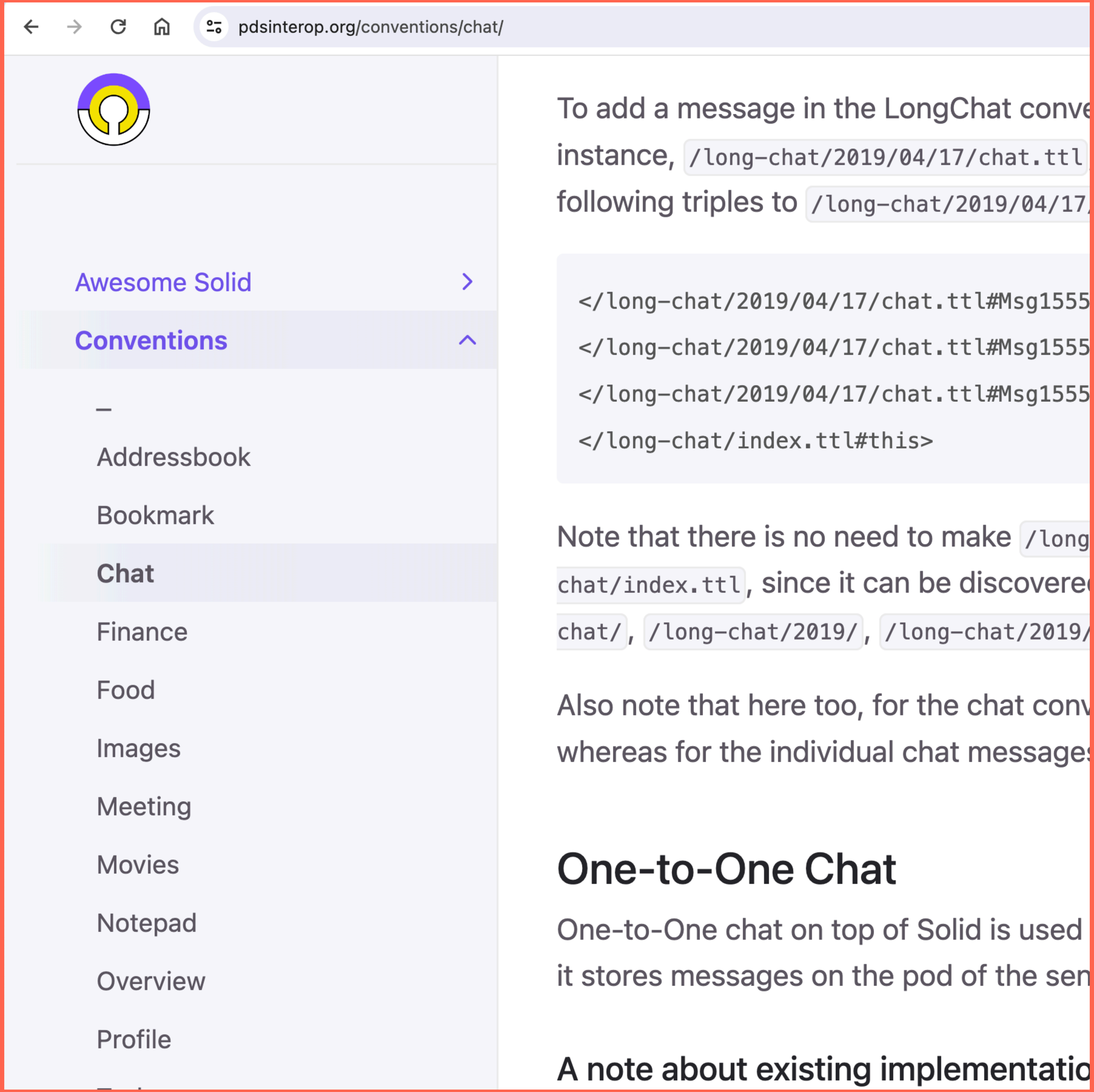


TABLE OF CONTENTS

A simple Long Chat instance

A SIMPLE LONG CHAT INSTANCE

```
# in $ROOT/index.ttl

@prefix : <#>.
@prefix dc: <http://purl.
@prefix meeting: <http:/
```



To add a message in the LongChat conve instance, `/long-chat/2019/04/17/chat.ttl` following triples to `/long-chat/2019/04/17/`

```
</long-chat/2019/04/17/chat.ttl#Msg1555
</long-chat/2019/04/17/chat.ttl#Msg1555
</long-chat/2019/04/17/chat.ttl#Msg1555
</long-chat/index.ttl#this>
```

Note that there is no need to make `/long-chat/index.ttl`, since it can be discovere `chat/`, `/long-chat/2019/`, `/long-chat/2019/`

Also note that here too, for the chat conv whereas for the individual chat message

One-to-One Chat

One-to-One chat on top of Solid is used it stores messages on the pod of the sen

A note about existing implementatio

SOLID

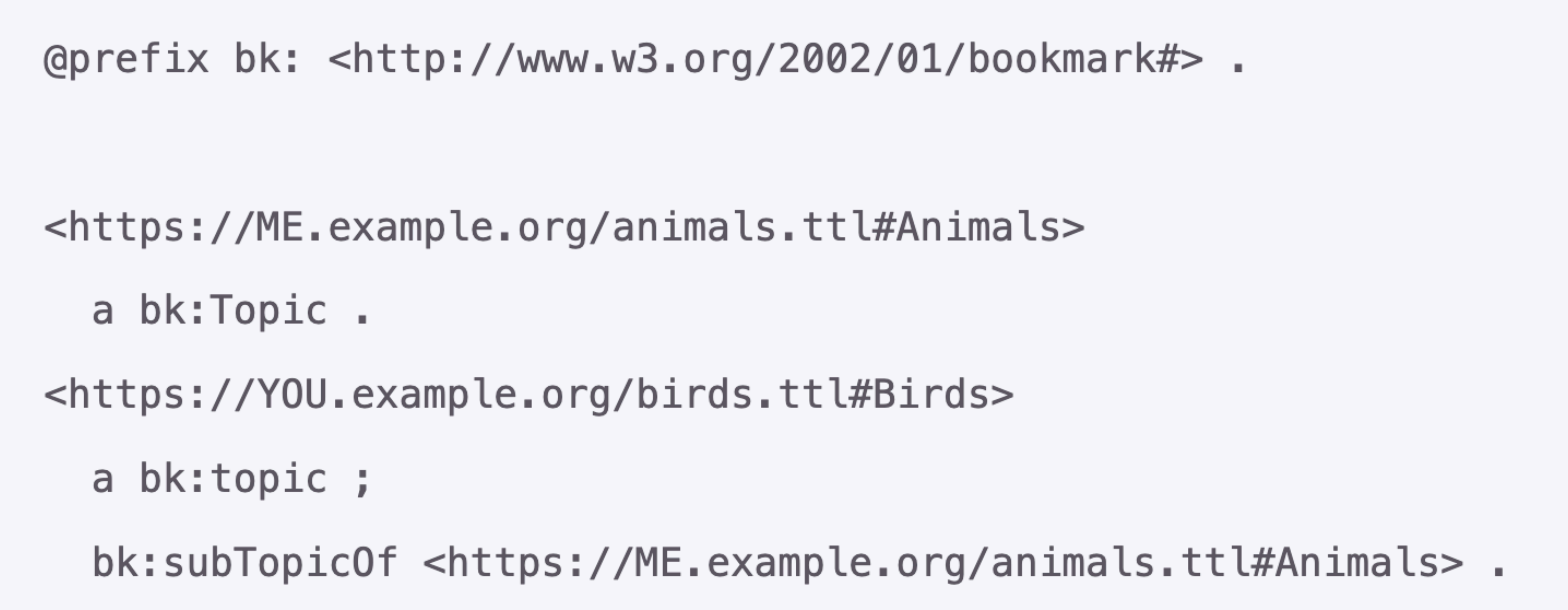
DATA

MODULES

nlnet

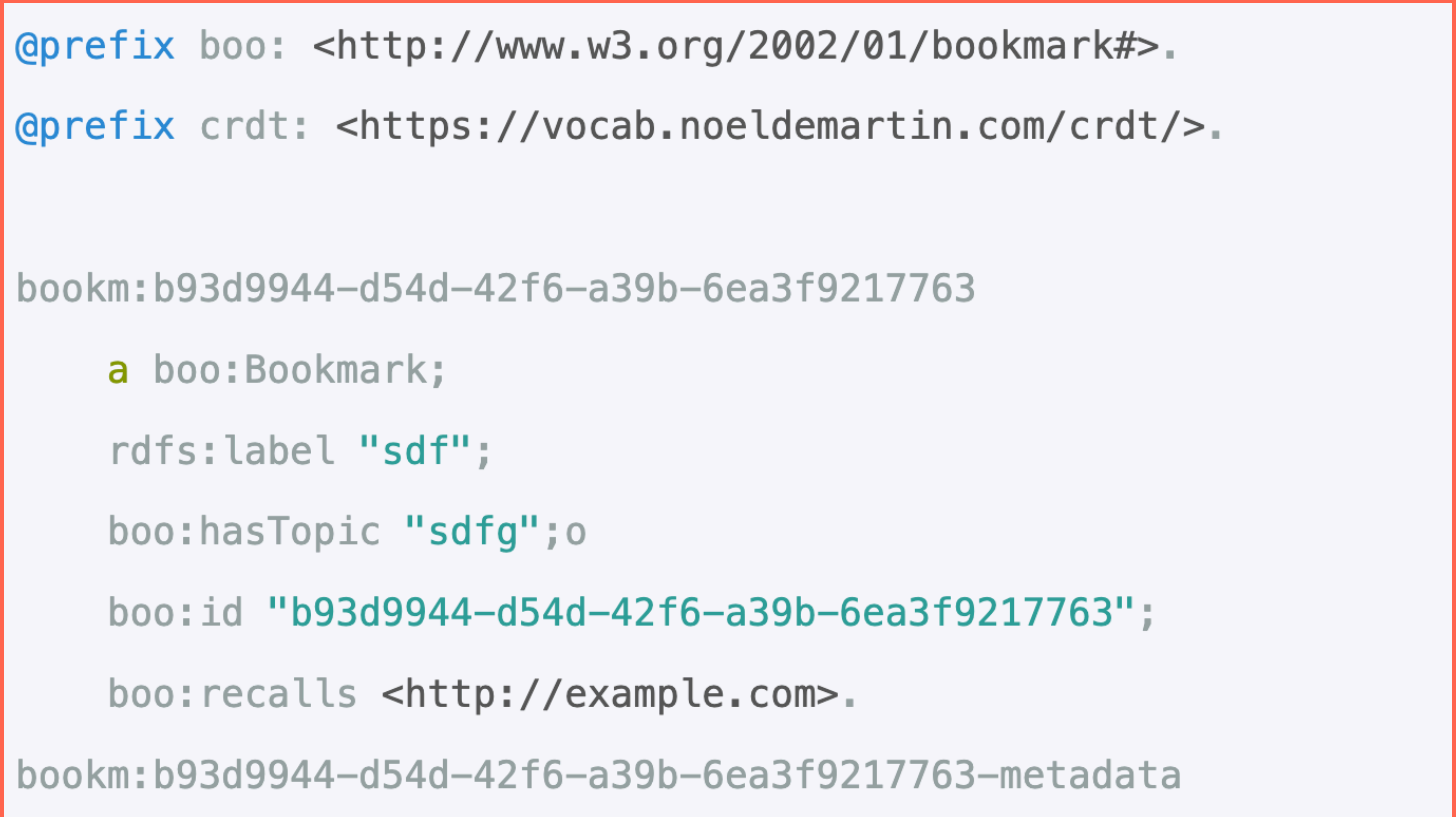
FOUNDATION

A SOLID BOOKMARK



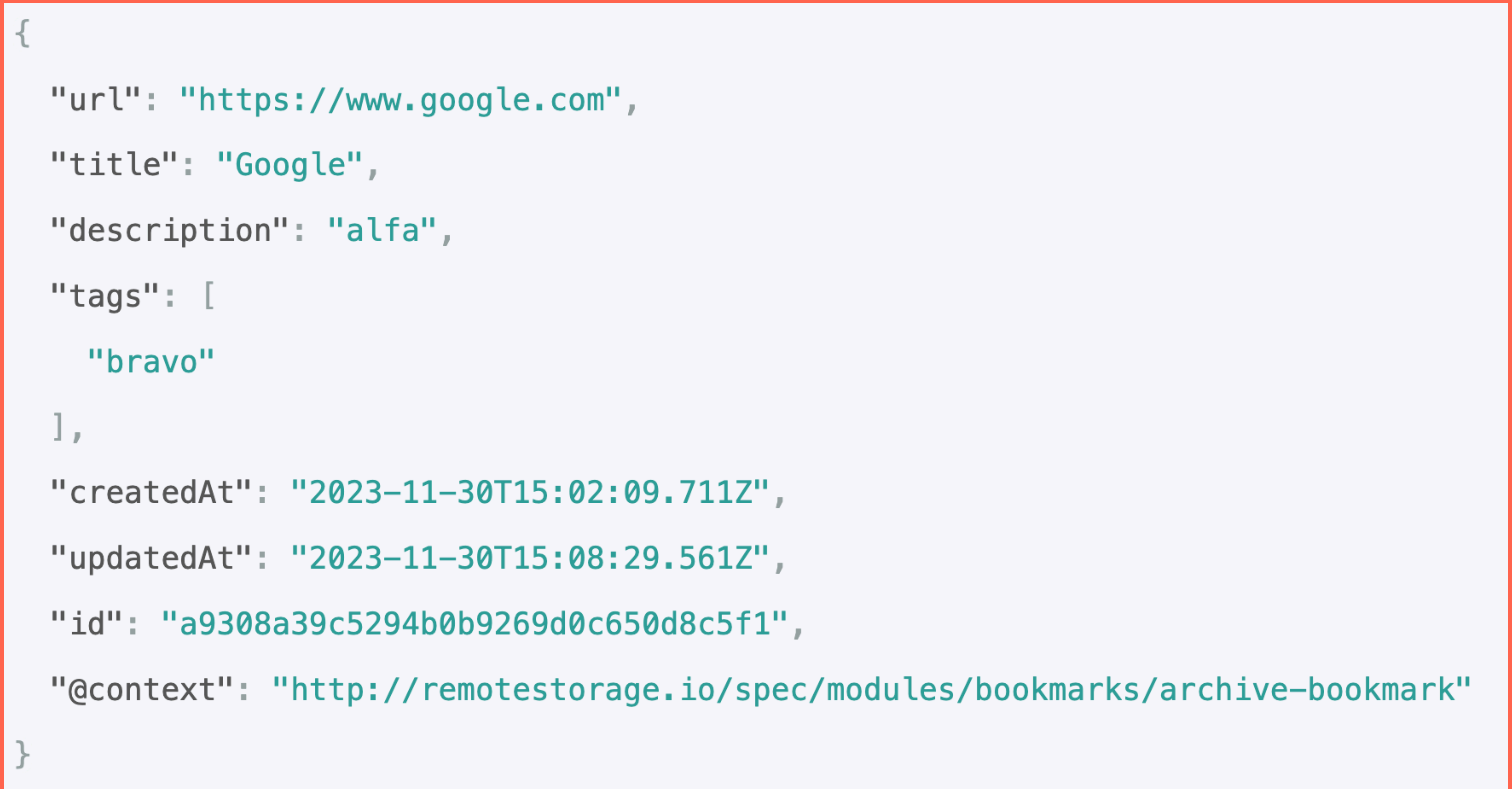
```
@prefix bk: <http://www.w3.org/2002/01/bookmark#> .


<https://ME.example.org/animals.ttl#Animals>
  a bk:Topic .
<https://YOU.example.org/birds.ttl#Birds>
  a bk:topic ;
  bk:subTopicOf <https://ME.example.org/animals.ttl#Animals> .
```

A SOLID BOOKMARK



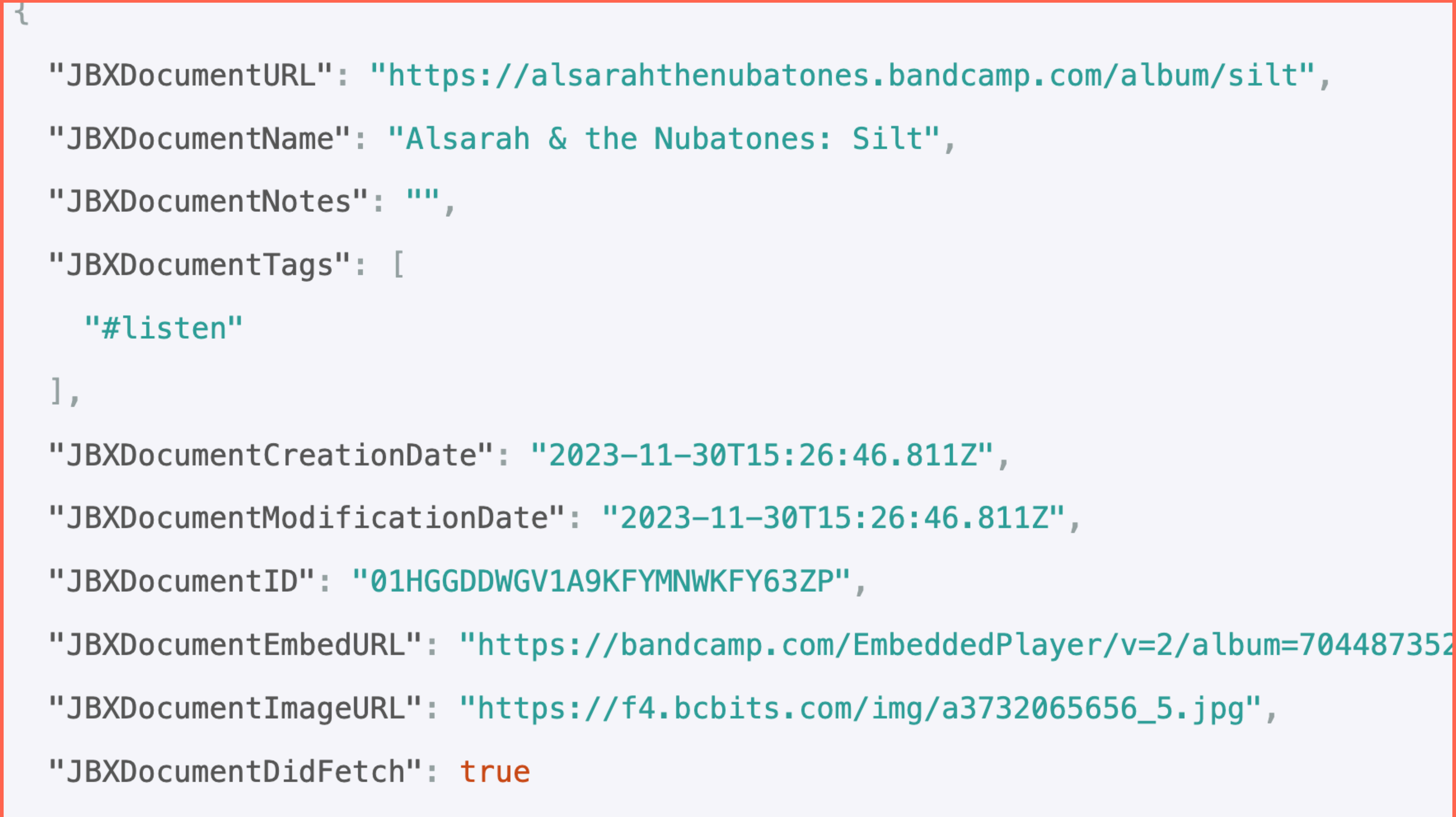
```
@prefix boo: <http://www.w3.org/2002/01/bookmark#>.
@prefix crdt: <https://vocab.noeldemartin.com/crdt/>.


bookm:b93d9944-d54d-42f6-a39b-6ea3f9217763
    a boo:Bookmark;
    rdfs:label "sdf";
    boo:hasTopic "sdfg";o
    boo:id "b93d9944-d54d-42f6-a39b-6ea3f9217763";
    boo:recalls <http://example.com>.
bookm:b93d9944-d54d-42f6-a39b-6ea3f9217763-metadata
```

A SOLID BOOKMARK



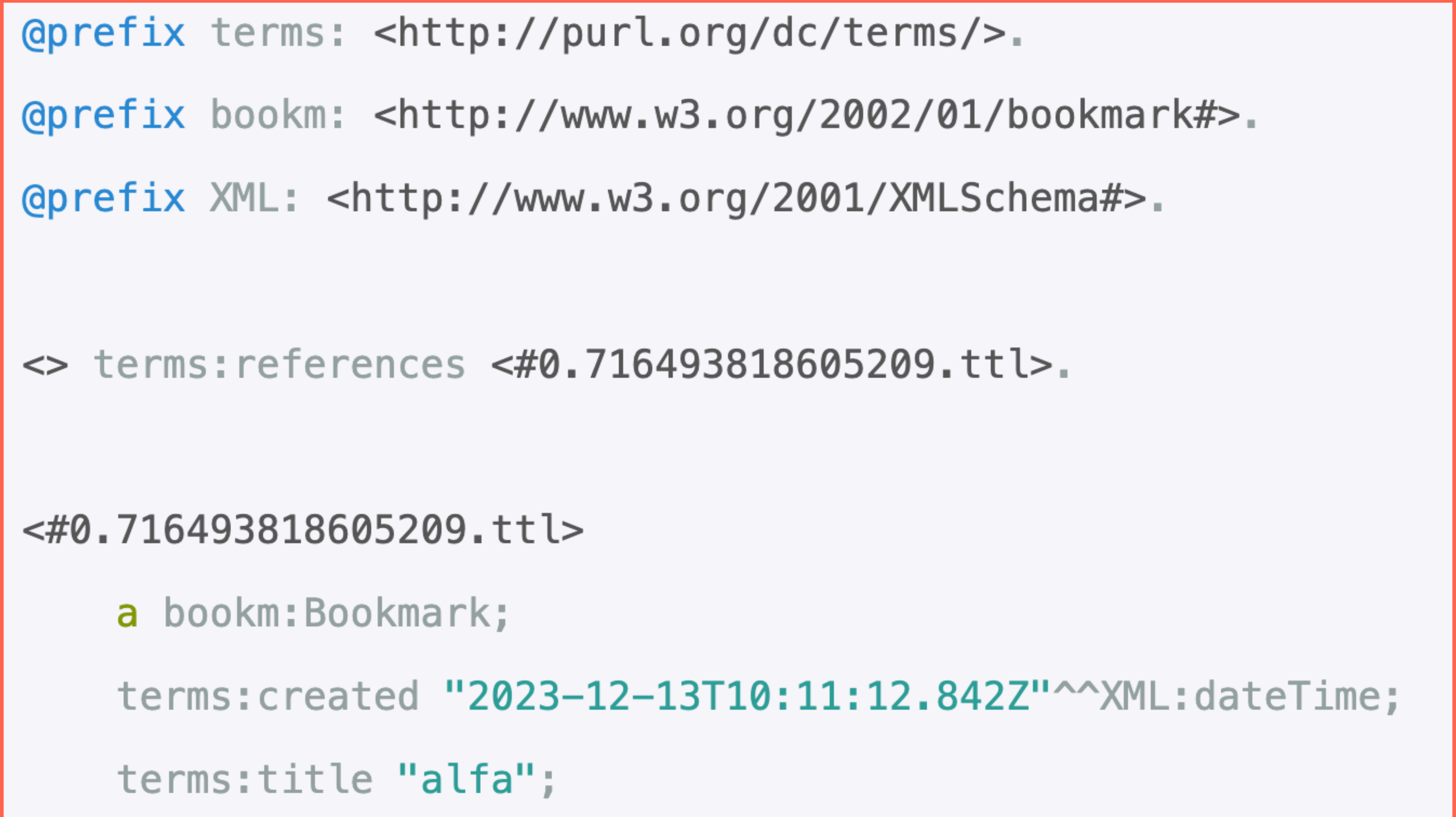
```json
{
  "url": "https://www.google.com",
  "title": "Google",
  "description": "alfa",
  "tags": [
    "bravo"
  ],
  "createdAt": "2023-11-30T15:02:09.711Z",
  "updatedAt": "2023-11-30T15:08:29.561Z",
  "id": "a9308a39c5294b0b9269d0c650d8c5f1",
  "@context": "http://remotestorage.io/spec/modules/bookmarks/archive-bookmark"
}
```

A SOLID BOOKMARK



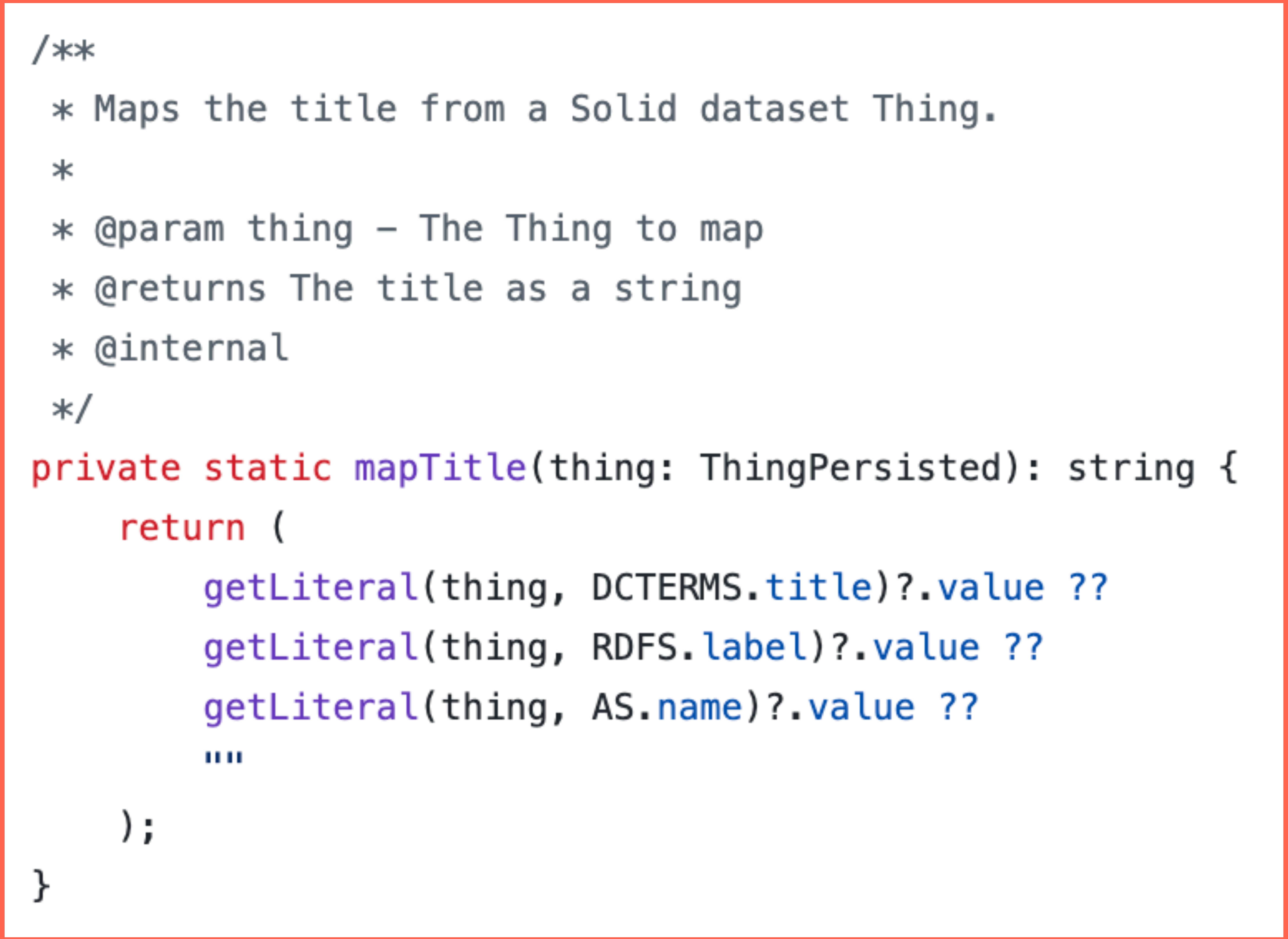
```
{
  "JBXDocumentURL": "https://alsarahthenubatones.bandcamp.com/album/silt",
  "JBXDocumentName": "Alsarah & the Nubatones: Silt",
  "JBXDocumentNotes": "",
  "JBXDocumentTags": [
    "#listen"
  ],
  "JBXDocumentCreationDate": "2023-11-30T15:26:46.811Z",
  "JBXDocumentModificationDate": "2023-11-30T15:26:46.811Z",
  "JBXDocumentID": "01HGGDDWGV1A9KFYMNWKFY63ZP",
  "JBXDocumentEmbedURL": "https://bandcamp.com/EmbeddedPlayer/v=2/album=704487352
  "JBXDocumentImageURL": "https://f4.bcbits.com/img/a3732065656_5.jpg",
  "JBXDocumentDidFetch": true
```

A SOLID BOOKMARK

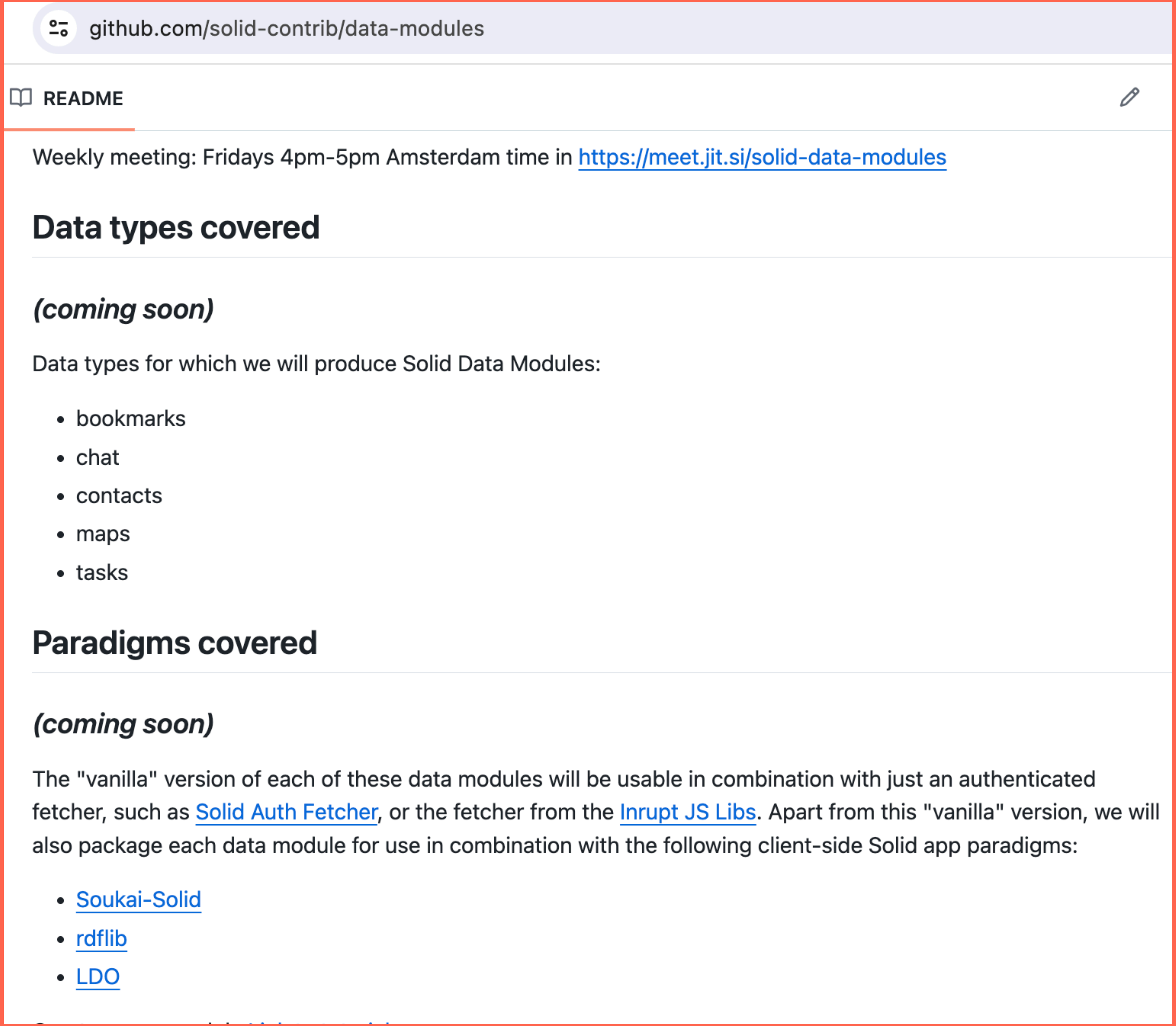


```
@prefix terms: <http://purl.org/dc/terms/>.
@prefix bookm: <http://www.w3.org/2002/01/bookmark#>.
@prefix XML: <http://www.w3.org/2001/XMLSchema#>.

<> terms:references <#0.716493818605209.ttl>.

<#0.716493818605209.ttl>
    a bookm:Bookmark;
    terms:created "2023-12-13T10:11:12.842Z"^^XML:dateTime;
    terms:title "alfa";
```

SOLID BOOKMARK MODULE



```
/**
 * Maps the title from a Solid dataset Thing.
 *
 * @param thing - The Thing to map
 * @returns The title as a string
 * @internal
 */
private static mapTitle(thing: ThingPersisted): string {
    return (
        getLiteral(thing, DCTERMS.title)?.value ??
        getLiteral(thing, RDFS.label)?.value ??
        getLiteral(thing, AS.name)?.value ??
        ""
    );
}
```



Weekly meeting: Fridays 4pm-5pm Amsterdam time in https://meet.jit.si/solid-data-modules

Data types covered

(coming soon)

Data types for which we will produce Solid Data Modules:

- bookmarks
- chat
- contacts
- maps
- tasks

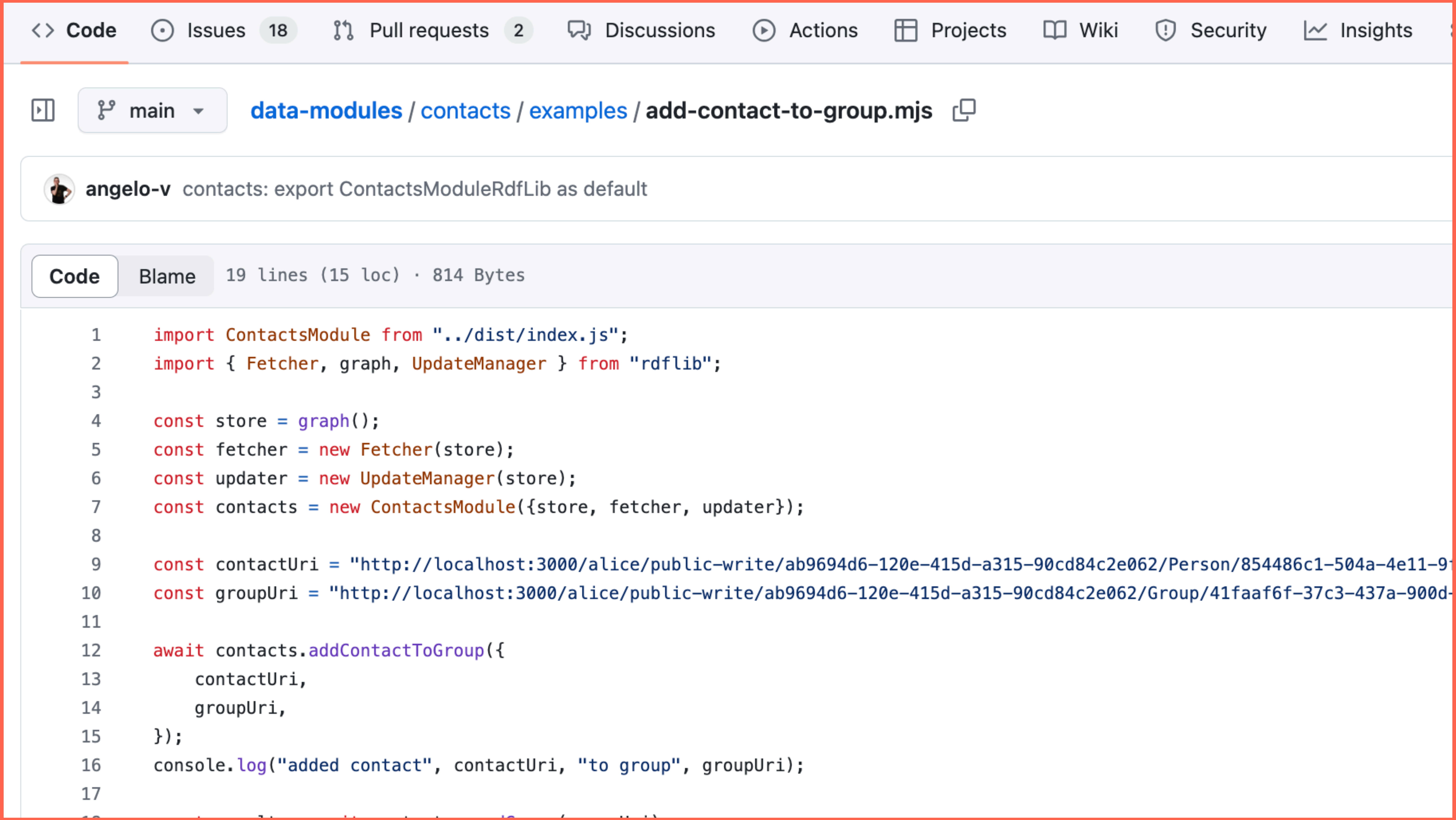
Paradigms covered

(coming soon)

The "vanilla" version of each of these data modules will be usable in combination with just an authenticated fetcher, such as Solid Auth Fetcher, or the fetcher from the Inrupt JS Libs. Apart from this "vanilla" version, we will also package each data module for use in combination with the following client-side Solid app paradigms:

- Soukai-Solid
- rdflib
- LDO

DATA MODULES DX



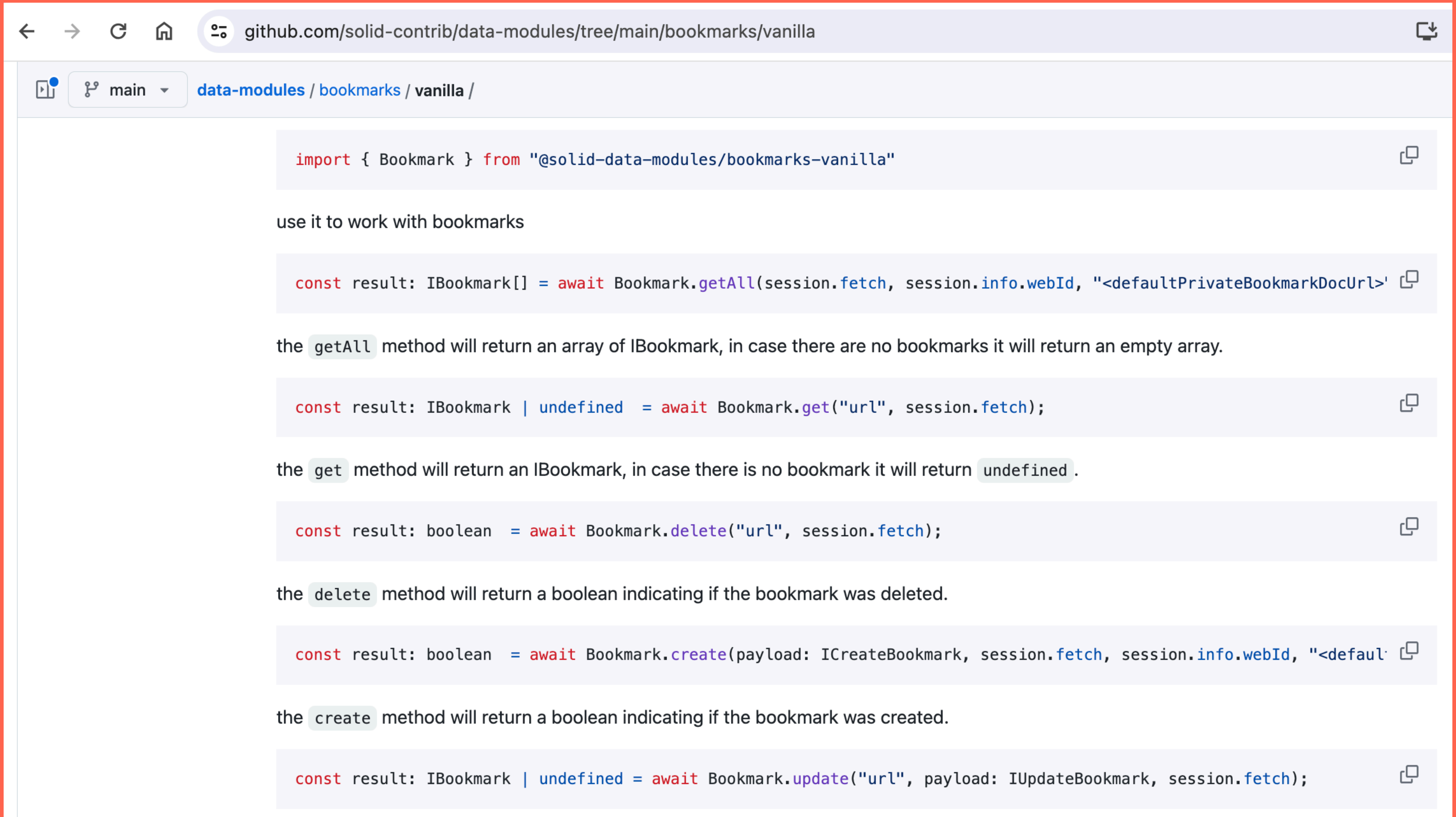
data-modules / contacts / examples / add-contact-to-group.mjs

angelo-v contacts: export ContactsModuleRdfLib as default

19 lines (15 loc) · 814 Bytes

```
import ContactsModule from "../dist/index.js";
import { Fetcher, graph, UpdateManager } from "rdflib";

const store = graph();
const fetcher = new Fetcher(store);
const updater = new UpdateManager(store);
const contacts = new ContactsModule({store, fetcher, updater});

const contactUri = "http://localhost:3000/alice/public-write/ab9694d6-120e-415d-a315-90cd84c2e062/Person/854486c1-504a-4e11-9
const groupUri = "http://localhost:3000/alice/public-write/ab9694d6-120e-415d-a315-90cd84c2e062/Group/41faaf6f-37c3-437a-900d-

await contacts.addContactToGroup({
    contactUri,
    groupUri,
});
console.log("added contact", contactUri, "to group", groupUri);

```

DATA MODULES DX


github.com/solid-contrib/data-modules/tree/main/bookmarks/vanilla
main
data-modules / bookmarks / vanilla /

```
import { Bookmark } from "@solid-data-modules/bookmarks-vanilla"
```

use it to work with bookmarks

```
const result: IBookmark[] = await Bookmark.getAll(session.fetch, session.info.webId, "<defaultPrivateBookmarkDocUrl>'
```

the `getAll` method will return an array of IBookmark, in case there are no bookmarks it will return an empty array.

```
const result: IBookmark | undefined  = await Bookmark.get("url", session.fetch);
```

the `get` method will return an IBookmark, in case there is no bookmark it will return `undefined`.

```
const result: boolean  = await Bookmark.delete("url", session.fetch);
```

the `delete` method will return a boolean indicating if the bookmark was deleted.

```
const result: boolean  = await Bookmark.create(payload: ICreateBookmark, session.fetch, session.info.webId, "<defaul
```

the `create` method will return a boolean indicating if the bookmark was created.

```
const result: IBookmark | undefined = await Bookmark.update("url", payload: IUpdateBookmark, session.fetch);
```

Solid Data Modules

As The Engine Of

Solid App Interop

PONDER SOURCE

Shape Trees As The Engine Of Data Discovery

PONDER SOURCE

Pod Indexer

As The Engine Of

Data Discovery

Shape Trees As The Engine Of Data Grants

PONDER SOURCE

Solid Data Modules

+ Indexer +

SAI

PONDER SOURCE

GITHUB.COM/
SOLID-CONTRIB/
DATA-MODULES

PONDER
SOURCE