

# The Solid Ecosystem

///// Tutorial CTWE WS2019/2020 //////////////////////////////////////

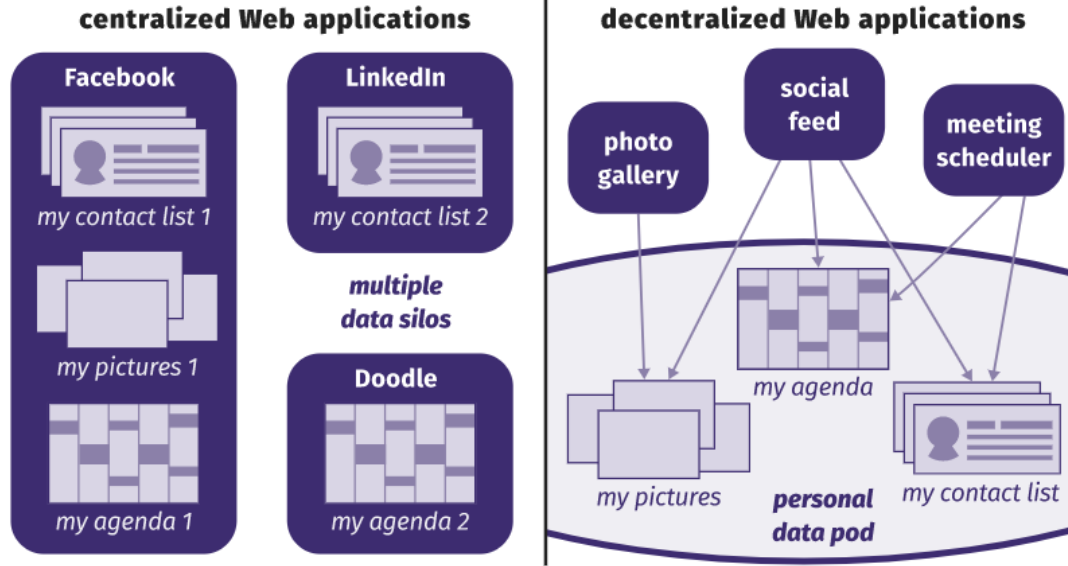
**Dang Vu**

Professorship for Distributed and Self-Organizing Systems

The background is a dark green chalkboard filled with various mathematical symbols and diagrams drawn in white chalk. In the top left, there is a pie chart with one slice labeled '20%' and another section labeled '8001'. To the right of the pie chart are symbols for a circle with a horizontal line through it, a plus sign, a minus sign, a multiplication sign, and a percent sign. Below these are the Greek letters alpha and a square with a cross inside. In the bottom right, the value of pi is written as '= 3.1415926'. Other symbols include the Greek letters epsilon, sigma, and phi, as well as a large 'X' and a checkmark. On the left side, there are small stick figures and a large white shape that partially obscures the text.

# Linked Data Platform

# You can grant apps and people access to very specific parts of your data.



- <https://rubenverborgh.github.io/Solid-Lecture-2018/>



**“Linked Data Platform (LDP)** is a Linked Data specification defining a set of integration patterns for building RESTful HTTP services that are capable of read-write of RDF data.”

- [https://en.wikipedia.org/wiki/Linked\\_Data\\_Platform](https://en.wikipedia.org/wiki/Linked_Data_Platform)
- <https://www.w3.org/TR/ldp-primer/>
- <https://www.w3.org/TR/ldp/>

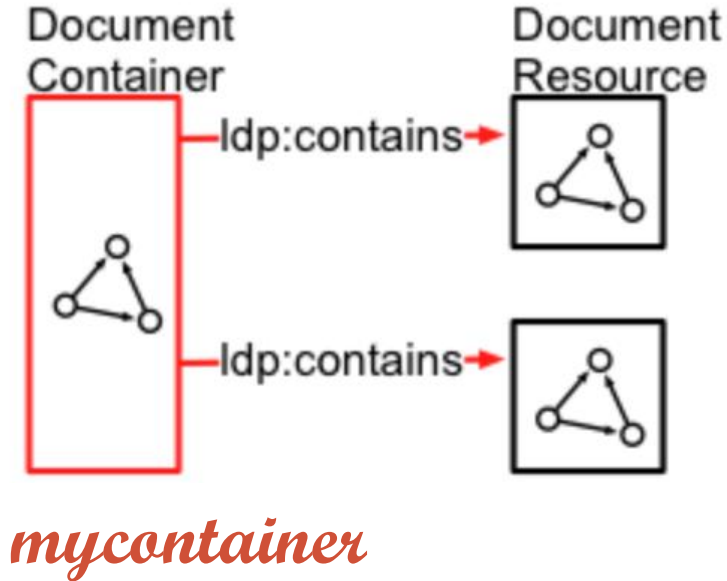


- Based on RDF
- Use HTTP Verbs for CRUD Operations
  - POST
  - GET
  - PUT/PATCH
  - DELETE
- Cover “RDF sources” as well as binary resources (LDPR)
- Organize these resources in LDP containers (LDPC)

# Implementations

- Apache Marmotta
- Trellis LDP
- Fedora Commons Repository
- SoLiD

GET /mycontainer/ HTTP/1.0



HTTP/1.0 200 OK

Link: <<http://www.w3.org/ns/ldp#BasicContainer>>; rel="type",  
<<http://www.w3.org/ns/ldp#Resource>>; rel="type"

@prefix dcterms: <<http://purl.org/dc/terms/>>.

@prefix ldp: <<http://www.w3.org/ns/ldp#>>.

<<http://example.org/mycontainer/>>

a ldp:Container, ldp:BasicContainer;

dcterms:title 'My data storage on the Web' .

Document  
Container



ldp:contains →

Document  
Resource



ldp:contains →



*mycontainer*

POST /mycontainer/ HTTP/1.0

Link: <<http://www.w3.org/ns/ldp#Resource>>; rel="type"

Slug: myfile.ttl

Content-Type: text/turtle

@prefix dc: <<http://purl.org/dc/terms/>> .

@prefix foaf: <<http://xmlns.com/foaf/0.1/>> .

<> a foaf:PersonalProfileDocument;

foaf:primaryTopic <#me> ;

dc:title 'My Profile file' .

<#me> a foaf:Person;

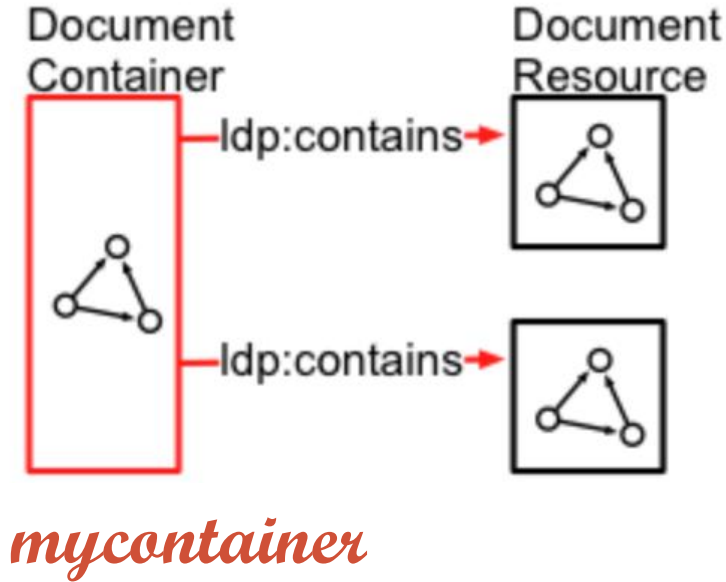
foaf:name 'Oliver Meier' .

HTTP/1.0 201 Created

Location: <http://example.org/mycontainer/myfile.rdf>

Link: <<http://www.w3.org/ns/ldp#Resource>>; rel="type"





POST /mycontainer/ HTTP/1.0

Link: <<http://www.w3.org/ns/ldp#Resource>>; rel="type"

Slug: myfile.png

Content-Type: image/png

# binary data

HTTP/1.0 201 Created

Location: <http://example.org/mycontainer/myfile.png>

Link: <<http://www.w3.org/ns/ldp#Resource>>; rel="type"

Link: <<http://example.org/mycontainer/myfile.png/meta>>;  
rel="describedby"



## What is SoLiD?

- Technology Stack
  - LDP REST API
  - Authentication
  - Authorization
- Ecosystem for Social Apps
- Practical Implementation

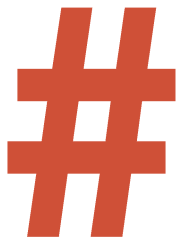
- <https://github.com/solid/solid>
- <https://github.com/solid/solid-spec>

## Extension to LDP

- “POD” - concept
- WebID-based Authentication
- Access Control Lists (.acl) based on WAC
- SPARQL-endpoint for each resource
- Basic prototypical Web User Interface
- Multiple application types

The background is a dark blue gradient with a grid pattern. It features several concentric circles and arcs, some of which are composed of binary code (0s and 1s). A large, white, stylized number '2' is positioned on the left side of the image, partially overlapping the text.

# The SoLiD specification



## Time for our own POD

- <https://github.com/solid/node-solid-server>
  - <https://hub.docker.com/r/aveltens/solid-server>
  - <https://solid.inrupt.com/>
- <https://solid.inrupt.com/docs>
- <https://github.com/solid/userguide>

- Identity
- Profiles
- Authentication
- Authorization and Access Control
- Content Representation
- Reading and Writing Resources
- Social Web App Protocols



# Identity: how is this implemented?

## Using WebIDs

Example:

<https://vunguyenhaidang.solid.community/profile/card#me>

Or

<https://dangvu.com/profile/card#me>

# Profiles: What are WebID Profile Document?

```
@prefix solid: <http://www.w3.org/ns/solid/terms#>.
@prefix foaf: <http://xmlns.com/foaf/0.1/>.
@prefix pim: <http://www.w3.org/ns/pim/space#>.
@prefix schema: <http://schema.org/>.
@prefix ldp: <http://www.w3.org/ns/ldp#>.
```

```
<>
  a foaf:PersonalProfileDocument ;
  foaf:maker <https://vunguyenhaidang.solid.community/profile/card#me> ;
  foaf:primaryTopic <https://vunguyenhaidang.solid.community/profile/card#me> .

<https://vunguyenhaidang.solid.community/profile/card#me>
  a foaf:Person ;
  a schema:Person ;

  foaf:name "Dang Vu Nguyen Hai" ;

  solid:account </> ; # link to the account uri
  pim:storage </> ; # root storage

  ldp:inbox </inbox/> ;

  pim:preferencesFile </settings/prefs.ttl> ; # private settings/preferences
  solid:publicTypeIndex </settings/publicTypeIndex.ttl> ;
  solid:privateTypeIndex </settings/privateTypeIndex.ttl> .
```

User identity

User preferences discover

Security credentials

Linked Data Format: turtle (.ttl), JSON-LD or HTML+RDFa

# Authentication:

## Primary Authentication :

- Traditional approach: username-and-password
- WebID-TLS: public-private key pair
- WebID-OIDC: based on the OAuth2/OpenID (currently implementing)

## Secondary Authentication:

- Passwords can be forgotten, browser certificates can be lost...
- > Account Recovery

# Authorization and Access Control:

## Web Access Control (WAC)

- The resources are identified by URIs
- Users and the groups are all identified by URIs
- Authorization statements are placed into Access Control List Resource (.acl)

```
<#authorization1>
  a      acl:Authorization;
  acl:agent    <https://alice.databox.me/profile/card#me>; # Alice's WebID
  acl:accessTo <https://alice.databox.me/docs/file1>;
  acl:mode     acl:Read,
                acl:Write,
                acl:Control.
```

# Content Representation:

## Two kinds of resources

- Linked Data resources (RDF in form of JSON-LD, Turtle, HTML+RDFa, etc)
- Everything else (binary data and non-linked-data structured text)

## Resources are grouped in directory- like **Containers**

- Basic Container
- Direct Container
- Indirect Container

# Reading and Writing Resources:

- HTTPS REST API
- WebSocket API

## What is globbing?

TO aggregate all RDF resources from a container and retrieve them with a single GET operation


Alternative: using SPARQL



# Social Web App Protocols:

What are they for?

- Notifications
- Friends Lists, Followers and Following

A large, bold white number '3' is positioned on the left side of the slide. The background is a dark blue gradient with a pattern of embossed question marks. Three question marks are highlighted in a reddish-orange color: one in the upper right, one in the middle left, and one in the lower center.

# 3

# Hands-on



# Time for making a first Solid App

- <https://solid.inrupt.com/docs/app-on-your-lunch-break>

# Learning objectives:

- Building a basic app with Solid
- Logging in and out
- Reading data from a Solid pod

# Prerequisites:

- HTML, CSS, JavaScript knowledge

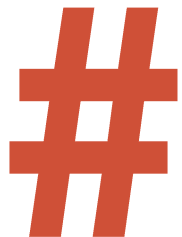
# Required tools:

- Text editor (i.e. Visual Studio Code)
- a Web Server to run your app locally (i.e local-web-server)

Download sample code: [mytuc.org/nhyr](https://mytuc.org/nhyr)

# Use Cases



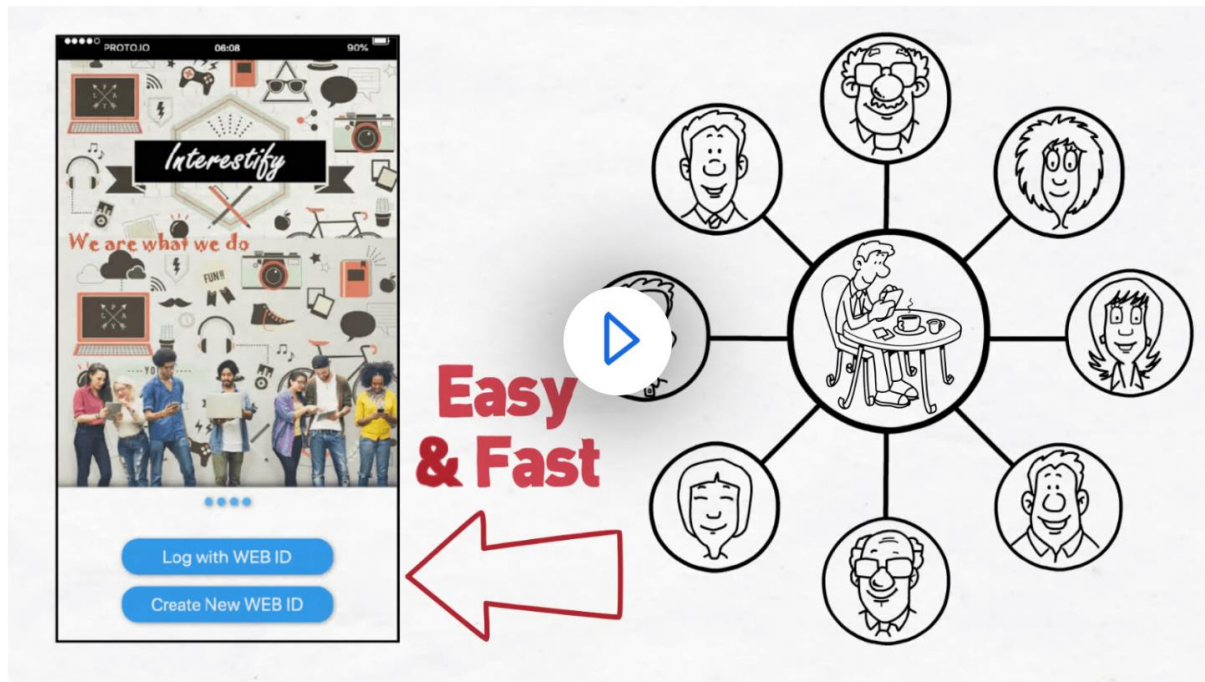


<https://github.com/solid/solid-apps>

# Management of medical health records



# Find new friends



The background is a landscape photograph featuring a prominent rock arch in the center-left, with a bright light source (likely the sun) shining through it, creating a lens flare effect. The sky is filled with soft, pinkish-purple clouds. In the foreground, there is a field of tall, dry grass. To the right, there are rolling hills. A large, white, stylized number '5' is overlaid on the left side of the image, partially obscuring the landscape.

# Conclusion