



# Introduction to the Linked Data Platform (LDP)

Hector Correa  
[hjc14@psu.edu](mailto:hjc14@psu.edu)

The Pennsylvania State University

# Agenda

- What is the Linked Data Platform (LDP)
- Basic definitions of concepts and rules
- Examples
- Why this is important
- Q&A

# Related Sessions

- **Linked data, URIs, and labels** by Trey Terrel
- **Introducing Hydra-works: PCDM in Hydra** by Lynette Rayle

# Linked Data

First introduced by Tim Berners-Lee in 2006

**"The Semantic Web isn't just about putting data on the web. It is about making links, so that a person or machine can explore the web of data"**

- Identify things with valid URLs
- Use standards for representation
- Include other URLs in your data

See <http://www.w3.org/DesignIssues/LinkedData.html>

https://en.wikipedia.org/wiki/Abraham\_Lincoln

The Wikipedia logo is visible in the top left corner.

The page title is "Abraham Lincoln".

The main text states: "This article is about the American president. For other uses, see [Abraham Lincoln \(disambiguation\)](#)".

The article begins with a brief summary of Lincoln's life, mentioning his birth in Hodgenville, Kentucky, and his rise to become the 16th President of the United States.

A large portrait of Abraham Lincoln is displayed, with the caption: "Lincoln in 1863 at the age of 54".

The sidebar contains sections for "Interaction" (Help, About Wikipedia, Community portal, Recent changes, Contact page), "Tools" (What links here, Related changes, Upload file, Special pages, Permanent link, Page information, Wikidata item, Cite this page), and "Languages" (Afrikaans, Alemannisch, ହୃଦୟ, English, العربية, Aragonés, Asturianu, Aymar aru, Azerbaijani, বাংলা, Bân-lâm-gú, Башҡортса, Беларуская, Беларуская (тарашкевіца), Bikol Central).

The sidebar also lists his political career:

- 16th President of the United States**  
In office  
March 4, 1861 – April 15, 1865  
Vice President: Hannibal Hamlin (1861–1865); Andrew Johnson (1865)
- Preceded by: James Buchanan  
Succeeded by: Andrew Johnson
- Member of the U.S. House of Representatives from Illinois's 7th district**  
In office  
March 4, 1847 – March 4, 1849  
Preceded by: John Henry  
Succeeded by: Thomas Harris
- Member of the Illinois House of Representatives**  
In office  
1834–1842
- Personal details**  
Born: February 12, 1809  
Hodgenville, Kentucky, U.S.  
Died: April 15, 1865 (aged 56)  
Petersen House, Washington, D.C., U.S.

Birth place

[https://en.wikipedia.org/wiki/Abraham\\_Lincoln](https://en.wikipedia.org/wiki/Abraham_Lincoln)

# Linked Data

Abraham\_Lincoln label "Abraham Lincoln" .

Abraham\_Lincoln birthDate "1809-02-12" .

Abraham\_Lincoln profession "Lawyer" .

[ . . . ]

Abraham\_Lincoln birthPlace Hodgenville,\_Kentucky .

[http://dbpedia.org/resource/Abraham\\_Lincoln](http://dbpedia.org/resource/Abraham_Lincoln)

[http://dbpedia.org/resource/Hodgenville,\\_Kentucky](http://dbpedia.org/resource/Hodgenville,_Kentucky)

[http://dbpedia.org/data/Abraham\\_Lincoln.n3](http://dbpedia.org/data/Abraham_Lincoln.n3)

# RDF 101

- Resource Description Framework

<http://www.w3.org/TR/rdf-primer/>

- Data expressed in triples

`<subject> <predicate> <object>`

- For example

`<hydraconnect> <location> "Minneapolis"`

`<hydraconnect> <date> "2015-09-22"`

# From read-only to read-write

- How are we supposed to **update Linked Data**
- Intuitively we assume that we must use HTTP  
GET/POST and Resource Description  
Framework (RDF)...
- ...but what are the mechanics, the rules, how  
do we deal with containership

# Linked Data Platform (LDP)

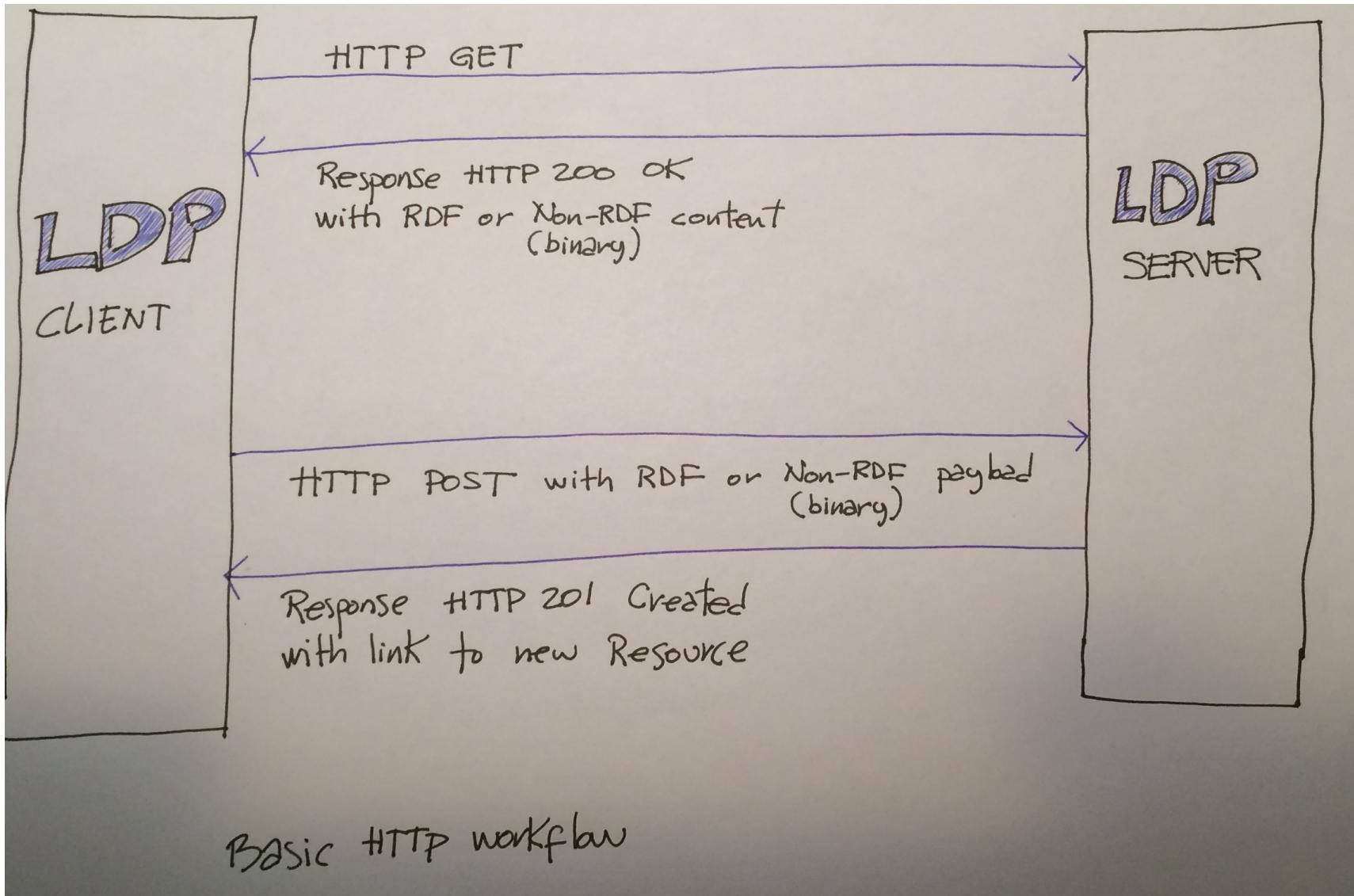
“defines a set of rules for HTTP operations on web resources, some based on RDF, to provide an architecture for **read-write** Linked Data on the web” <http://www.w3.org/TR/ldp/>

W3C Recommendation as of Feb/2015

Linked Data Platform (LDP)  
is an HTTP API for  
read-write Linked Data

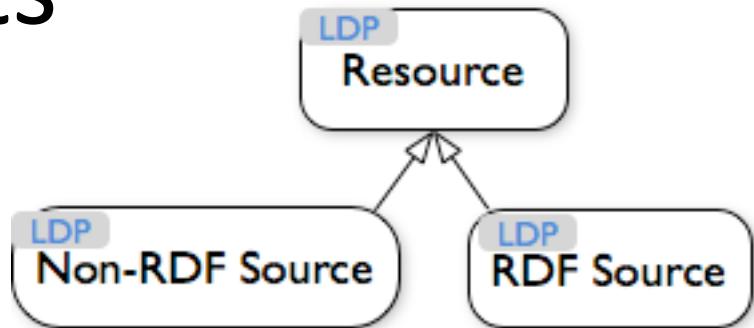
API stands for Application Programming Interface.  
An API defines the rules to communicate with a program.

# Basic HTTP workflow



# Concepts

- **Everything is a Resource**
- **RDF Sources:** expressed in Resource Description Framework (triples)
- **Non-RDF Sources:** everything else (web pages, PDFs, image/audio/video files, binaries, et cetera)



# Create a New RDF Source

## HTTP request

POST localhost/

Slug: **hydraconnect2015**

HTTP POST to create a  
new RDF Source  
(**hydraconnect2015**)

<> dcterms:title "Hydra Connect 2015".

<> dcterms:subject "Hydra conference in Minneapolis".

Triples for the  
new resource

## HTTP response

HTTP/1.1 201 Created

Location: <http://localhost/hydraconnect2015/>

Yup, created...

...and here is the URL of  
the new resource

# Fetch an Existing RDF Source

**Client makes HTTP request**

GET localhost/**hydraconnect2015**

HTTP GET request

**Server returns HTTP response**

HTTP/1.1 200 OK

Content-Type: text/turtle

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

RDF source that  
is a container

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

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

<localhost/**hydraconnect2015**> a ldp:BasicContainer ;  
dc:title "Hydra Connect 2015" ;  
dc:subject ""Hydra conference in Minneapolis" .

HTTP Headers

HTTP Body

# Create a New Non-RDF Source

## HTTP request

POST localhost/

Slug: logo.jpg

**Content-Type: image/jpeg**

Content- Length: 1020

[binary of the image goes here]

HTTP POST to create  
a new Non-RDF Source

Binary data

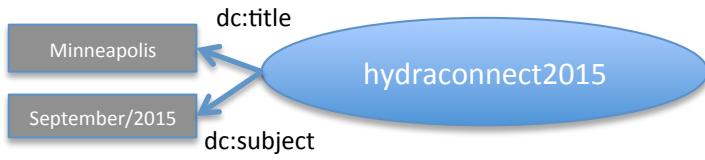
## HTTP response

HTTP/1.1 201 Created

Location: http://localhost/logo.jpg

...and here is the URL of  
the new resource

# Show me the Graph

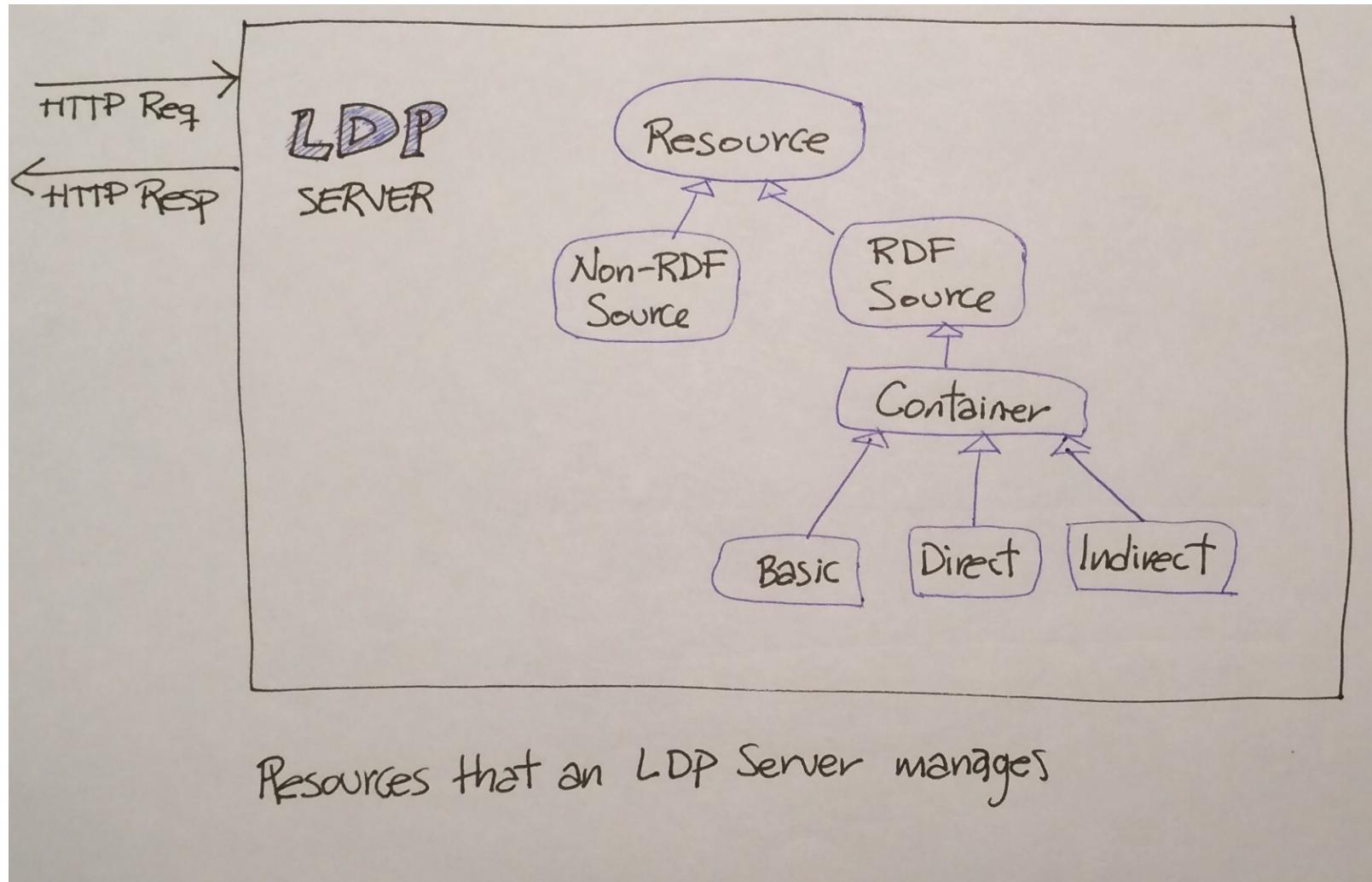


## Legend



→ Added by LDP Client

# Concepts (expanded)



# Adding an RDF Source to a Basic Container (1/2)

## **HTTP request**

POST localhost/hydraconnect2015/

Slug: **session1**

POST to Basic Container



<> dcterms:title "Welcome to Hydra Connect 2015".

<> dcterms:subject "blah blah".

## **HTTP response**

HTTP/1.1 201 Created

Location: http://localhost/hydraconnect2015/**session1**

# Adding an RDF Source to a Basic Container (2/2)

## HTTP request

```
GET localhost/hydraconnect2015/ HTTP/1.1
```

Fetch the Basic Container  

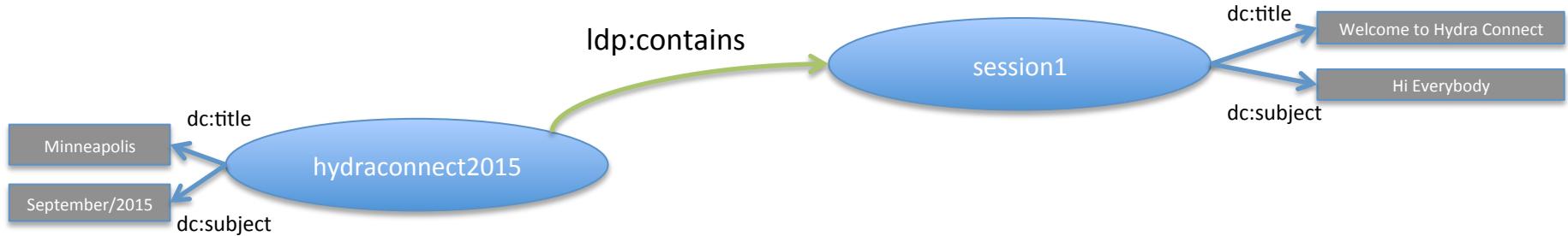

## HTTP response

```
[the usual HTTP headers go here]
```

```
<localhost/hydraconnect2015/> a ldp:BasicContainer ;  
    dc:title "Hydra Connect 2015" ;  
    dc:subject ""Hydra conference in Minneapolis" ;  
    ldp:contains <localhost/hydraconnect2015/session1> .
```

New triple automatically  
added by the LDP Server  


# Show me the Graph



## Legend



→ Added by LDP Client

→ Added by LDP Server

# Direct Containers (1/2)

## HTTP request

POST localhost/ HTTP/1.1

Create Direct Container

Slug: **speakers**

```
<> dc:title "People that speaks at conferences".  
<> rdf:DirectContainer .  
<> ldp:membershipResource <localhost/hydraconnect2015> .  
<> ldp:hasMemberRelation hasSpeaker .
```

Containership

## HTTP request

POST localhost/**speakers** HTTP/1.1

Add new resource to it

Slug: **janedev**

```
<> dc:title "Jane Developer".
```

# Direct Container (2/2)

**HTTP request**

GET localhost/**hydraconnect2015**

The membershipResource...



**HTTP response**

[bunch of HTTP headers go here]

```
<localhost/hydraconnect2015/> a ldp:BasicContainer ;  
    dc:title "Hydra Connect 2015" ;  
    dc:subject {"Hydra conference in Minneapolis"} ;  
    ldp:contains <localhost/hydraconnect2015/session1> ;  
    hasSpeaker <localhost/speakers/janedev> .
```

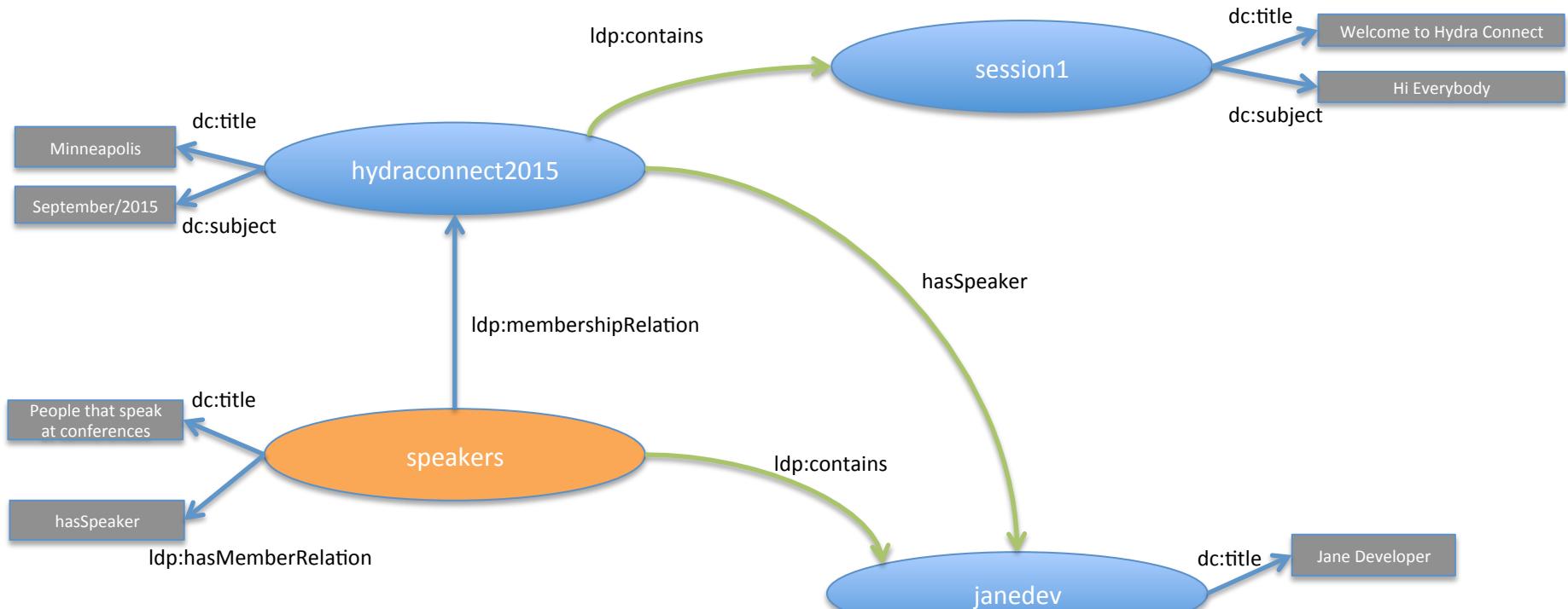


...has new triple with  
hasMemberRelation  
predicate



...pointing to the new  
resource

# Show me the Graph



## Legend



→ Added by LDP Client

→ Added by LDP Server

# Containers

- “Containers are at the heart of LDP” – [Robert Sanderson](#)
- In Direct and Indirect Containers you can revert the relationship (hasMemberRelation vs isMemberOfRelation)
- In an Indirect Container you can link to a totally different resource than the one added
- Shameless plug: LDP Containers for the Perplexed  
<http://tinyurl.com/ldp-containers>
- Read the W3C LDP Recommendation <http://www.w3.org/TR/ldp/>

# Updates

- Updates have not been standardized :(
- The Linked Data Platform Working Group ***is currently favoring LD Patch but still is deciding.***
- Other candidates include SPARQL 1.1, SparqlPatch, TurtlePatch, and RDF Patch
- As of July/2015, Linked Data Patch Format (LD Patch) is a note, *not* a recommendation
- LD Patch <http://www.w3.org/TR/2015/NOTE-lpatch-20150728/>

# Why is LDP important to Hydra?

- Fedora 4 is an LDP Server\*
- The Hydra Stack (i.e. your app) is an LDP Client
  - gems: LDP, ActiveTriples, ActiveFedora



\* Some restrictions apply

# Thanks!

- Official W3C LDP Recommendation

<http://www.w3.org/TR/ldp/>

- LDP Primer

<http://www.w3.org/TR/ldp-primer/>

- Slides/notes/examples

<http://hectorcorrea.com/introduction-to-ldp>