

isban

Center for Open Middleware / Ontology Engineering Group

Universidad Politécnica de Madrid,

Spain.

# Building interoperable read-write Linked Data applications with the W3C Linked Data Platform and the LDP4j framework

*ESWC 2015 - May 31<sup>st</sup> , 2015 - Portoroz (Slovenia)*

Miguel Esteban Gutiérrez,  
**Nandana Mihindukulasooriya,**

Raúl García Castro,

Building interoperable read-write Linked Data applications with the W3C Linked Data Platform and the LDP4j framework

# INTRODUCTION TO LDP

# Refreshing the basics

## “Linked Data principles

- Use URIs as names for things
- Use HTTP URIs so that people can look up those names
- When someone looks up a URI, provide useful information, using the standards (RDF\*, SPARQL)
- Include links to other URIs, so that they can discover more things”



Tim Berners-Lee. **Linked data-design issues** (2006) <http://www.w3.org/DesignIssues/LinkedData.html>  
Leo Sauermann and Richard Cyganiak. **Cool URIs for the Semantic Web** (2008) <http://www.w3.org/TR/cooluris/>  
Tom Heath and Christian Bizer. **Linked Data: Evolving the Web into a Global Data Space** (2011) <http://linkeddatabook.com/>

# Let's see an example

<http://id.nandana.org/card#me> ← (1) (2)

```
GET /card#me HTTP/1.1
Host: id.nandana.org
```

← (3)



```
HTTP/1.1 200 OK
Content-Type: text/turtle; charset=UTF-8
Content-Length: 250
```

```
@prefix dc: <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
```

```
< http://id.nandana.org/card> a foaf:PersonalProfileDocument;
  foaf:primaryTopic <http://example.org/nandana#me> ;
  dc:title 'Nandana's FOAF personal profile' .
```

```
< http://id.nandana.org/card#me> a foaf:Person;
  foaf:name 'Nandana Mihindukulasooriya';
  foaf:workplacehomepage <http://www.oeg-upm.net/> ;
  foaf:workplacehomepage <http://www.centeropenmiddleware.com/>;
  foaf:schoolhomepage <http://www.fi.upm.es/> ;
  foaf:topic_interest <http://dbpedia.org/resource/linked_data> ;
  foaf:based_near <http://dbpedia.org/resource/madrid> .
```

← (4)

HELLO  
my name is

*Nandana*  
*Mihindukulasooriya*

# Linked Data in practice

- Linked Data resources (documents)
  - easier to traverse, live data, less computational cost to the provider
  - slow querying
- Public SPARQL endpoints
  - live data
  - high cost to the provider, low availability
- Data dumps + private SPARQL endpoints
  - set up and maintenance cost, stale data, silos
  - high availability and performance
- Linked Data Fragments
  - less workload on server, high availability
  - more bandwidth, slow querying

**LDP makes these  
resources LD+RW**

# Linked Data Platform 1.0

<http://www.w3.org/2012/ldp/>

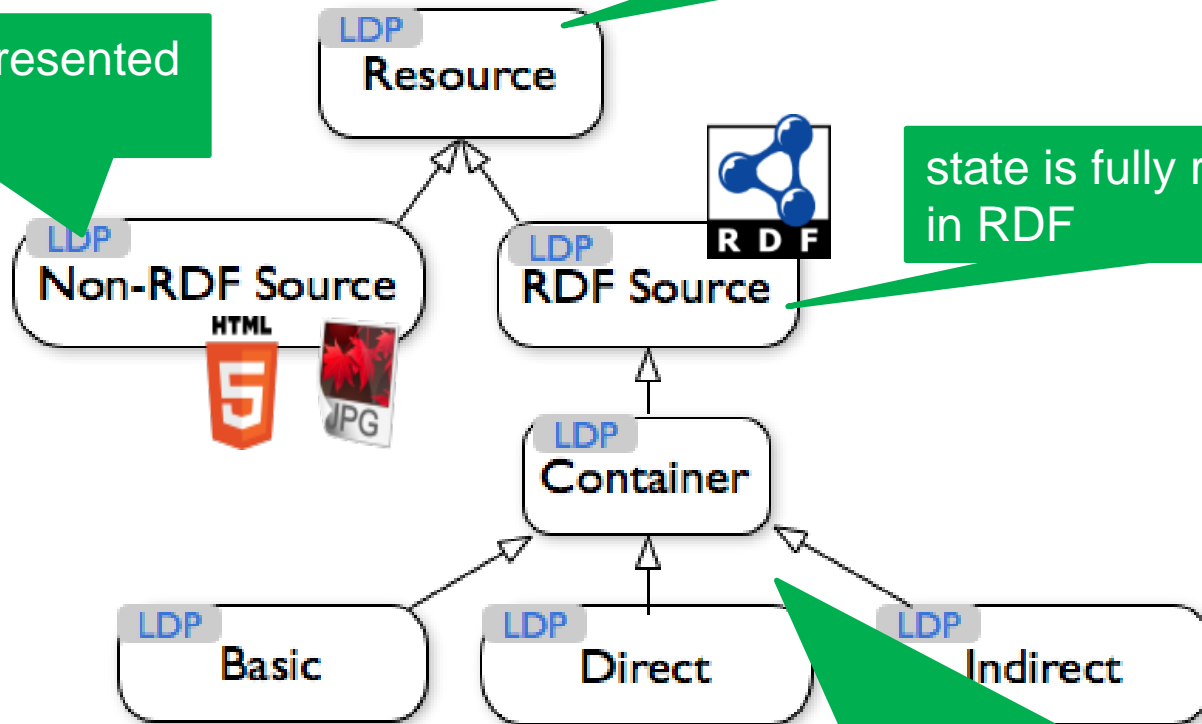


- RESTful protocol for **Read / Write** Linked Data
- Extensions to HTTP protocol
  - Pagination, ordering, partial updates, introspection, creation factories, collections
- Standardization focused of **interoperability**
- Developed by *W3C Linked Data Platform Working Group*
  - 54 participants from 32 organizations
    - Industry: IBM, Oracle, EMC, Fujitsu, OpenLink Software, Base22, ...
    - Academia: **Universidad Politécnica de Madrid**, University of Oxford, DERI, Université de Lyon, INRIA, ...
- *Progress:*
  - W3C Candidate Recommendation 19 June 2014,
  - W3C Last Call Working Draft (3) 7 September 2014,
  - W3C Proposed Recommendation October-November 2014
  - W3C Recommendation, 26 February 2015

# LDP Terminology

state is **NOT** represented  
in RDF

conforms to the lifecycle patterns  
and conventions defined in LDP



state is fully represented  
in RDF

a special type of RDF Source that acts as

- an enumeration of a collection of linked documents
- a creation factory

Building interoperable read-write Linked Data applications with the W3C Linked Data Platform and the LDP4j framework

# EXAMPLES



# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
  - Paging
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
  - Creating Linked Data Platform containers
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)

# Looking up a Linked Data resource

- a simple HTTP GET with
  - additional guarantees
    - LDP mandates some features that are optional in HTTP
      - e.g., e-tags, HEAD, OPTIONS
  - Some extensions
    - e.g., paging



# Looking up a Linked Data resource

## GET on LDPR

```
GET /nandana HTTP/1.1  
Host: example.org  
Accept: text/turtle
```

 GET

```
HTTP/1.1 200 OK  
Content-Type: text/turtle; charset=UTF-8  
Link: <http://www.w3.org/ns/ldp#Resource>; rel="type"  
ETag: "123456789"  
Allow: OPTIONS,HEAD,GET,PUT,PATCH  
Accept-Patch: text/ldpatch  
Content-Length: 250
```

 200 OK

```
@prefix dc: <http://purl.org/dc/terms/> .  
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
```

```
<http://example.org/nandana> a foaf:PersonalProfileDocument;  
  foaf:primaryTopic <http://example.org/nandana#me> ;  
  dc:title 'Nandana's FOAF file' .
```

```
<http://example.org/nandana#me> a foaf:Person;  
  foaf:name 'Nandana Mihindukulasooriya';  
  foaf:currentProject <http://www.seals-project.eu/> .
```

# Looking up a Linked Data document

## GET on LDPR

GET /nandana HTTP/1.1

Host:

HTTP/1.1 conformant

should support Turtle and  
JSON-LD representations

advertises support for LDP

Accept:

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

ETag: "123456789"

Allow: OPTIONS,HEAD,GET,PUT,PATCH

Accept-Patch: text/ldpatch

advertises which operations are allowed

provides  
an entity tag

Content-Length: 250

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

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

advertises media types accepted by the PATCH method

<http://example.org/nandana> a foaf:PersonalProfileDocument;

foaf:primaryTopic <http://example.org/nandana>;

dc:title 'Nandana's FOAF file' .

reuses existing vocabularies

use standard vocabularies [ e.g., Dublin Core, RDF(S) ]

<http://example.org/nandana#me> a foaf:Person;

foaf:name 'Nandana Mihindukulasooriya';

foaf:currentProject <http://www.seals-project.eu/> .

has at least one rdf:type set explicitly

# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
  - Paging
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
  - Creating Linked Data Platform containers
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)

# Paging

- Some resources are too large
- Clients may not be able to handle the complete resource
- Alternatives
  - Server-driven
  - Client-driven
- Client preferences
  - max-triple-count, max-kbyte-count, max-member-count
- Page links
  - **next**, canonical, first, last,

# Looking up a Linked Data document

## GET on LDPR with paging (I)

```
GET /nandana HTTP/1.1  
Host: example.org  
Accept: text/turtle  
Prefer: return=representation; max-triple-count="500"
```

GET

```
HTTP/1.1 303 See Other  
Location: <http://example.org/nandana?page1>
```

303 See Other

# Looking up a Linked Data document

## GET on LDPR with paging (I)

GET /nandana HTTP/1.1

Host: example.org

Accept: text/turtle

Prefer: return=representation; max-triple-count="500"

communicates support for LDP Paging and  
maximum desired size

GET

HTTP/1.1 303 See Other

Location: <http://example.org/nandana?page1>

redirects to the first page

303 See Other



# Looking up a Linked Data document

## GET on LDPR with paging (II)

```
GET /nandana?page1 HTTP/1.1
Host: example.org
Accept: text/turtle
Prefer: return=representation; max-triple-count="500"
```

GET

```
HTTP/1.1 200 OK
Content-Type: text/turtle
ETag: "_87e52ce291112"
Link: <http://www.w3.org/ns/ldp#Resource>; rel="type", <http://www.w3.org/ns/ldp#Page>; rel="type"
Link: <http://example.org/customer-relations?p=2>; rel='next'
Link: <http://example.org/nandana>; rel='canonical'; etag="_87e52ce291110"
Allow: GET,OPTIONS,HEAD
```

200 OK

```
@prefix dc: <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
```

```
<http://example.org/nandana> a foaf:PersonalProfileDocument;
  foaf:primaryTopic <http://example.org/nandana#me> ;
  dc:title 'Nandana's FOAF file' .
```

```
<http://example.org/nandana#me> a foaf:Person;
  foaf:name 'Nandana Mihindukulasooriya';
```

```
...
```

# Looking up a Linked Data document

## GET on LDPR with paging (II)

```
GET /nandana?page1 HTTP/1.1
Host: example.org
Accept: text/turtle
Prefer: return=representation; max-triple-count="500"
```

GET

```
HTTP/1.1 200 OK
Content-Type: text/turtle
ETag: "_87e52ce291112"
Link: <http://www.w3.org/ns/ldp#Resource>; rel="type", <http://www.w3.org/ns/ldp#Page>; rel="type"
Link: <http://example.org/customer-relations?p=2>; rel='next'
Link: <http://example.org/nandana>; rel='canonical'; etag="_87e52ce291110"
Allow: GET,OPTIONS,HEAD
```

advertises the response representation is a page

200 OK

provides a link to the next page

```
@prefix dc: <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
```

```
<http://example.org/nandana> a foaf:PersonalProfileDocument;
```

```
foaf:primaryTopic <http://example.org/nandana#me>;
dc:title "Nandana Mihindukulasooriya";
```

provides a link to resource being paged and the entity tag of complete resource

```
<http://example.org/nandana#me> a foaf:Person;
foaf:name 'Nandana Mihindukulasooriya';
```

...

# Looking up a Linked Data document

## GET on LDPR with paging (optimized)

```
GET /nandana HTTP/1.1
Host: example.org
Accept: text/turtle
Prefer: return=representation; max-triple-count="500"
Prefer: contents-of-related
```

```
HTTP/1.1 2XX Contents of Related
Content-Type: text/turtle
ETag: "_87e52ce291112"
Link: <http://www.w3.org/ns/ldp#Resource>; rel="type", <http://www.w3.org/ns/ldp#Page>; rel="type"
Link: <http://example.org/customer-relations?p=2>; rel='next'
Link: <http://example.org/nandana>; rel='canonical'; etag="_87e52ce291110"
Allow: GET,OPTIONS,HEAD
```

```
@prefix dc: <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
```

```
<http://example.org/nandana> a foaf:PersonalProfileDocument;
  foaf:primaryTopic <http://example.org/nandana#me> ;
  dc:title 'Nandana's FOAF file' .
```

```
<http://example.org/nandana#me> a foaf:Person;
  foaf:name 'Nandana Mihindukulasooriya';
```

```
...
```

# Looking up a Linked Data document

## GET on LDPR with paging (optimized)

```
GET /nandana HTTP/1.1
```

```
Host: example.org
```

```
Accept: text/turtle
```

```
Prefer: return=representation; max-triple-count="500"
```

```
Prefer: contents-of-related
```

communicates that the client understand  
"contents-of-related" status code

```
HTTP/1.1 2XX Contents of Related
```

```
Content-Type: text/turtle
```

```
ETag: "_87e52ce291112"
```

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

```
Link: <http://example.org/customer-relations?p=2>; rel='next'
```

```
Link: <http://example.org/nandana>; rel='canonical'; etag="_87e52ce291110"
```

```
Allow: GET,OPTIONS,HEAD
```

paged response status code

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

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

```
<http://example.org/nandana> a foaf:PersonalProfileDocument;  
  foaf:primaryTopic <http://example.org/nandana#me> ;  
  dc:title 'Nandana's FOAF file' .
```

```
<http://example.org/nandana#me> a foaf:Person;  
  foaf:name 'Nandana Mihindukulasooriya';
```

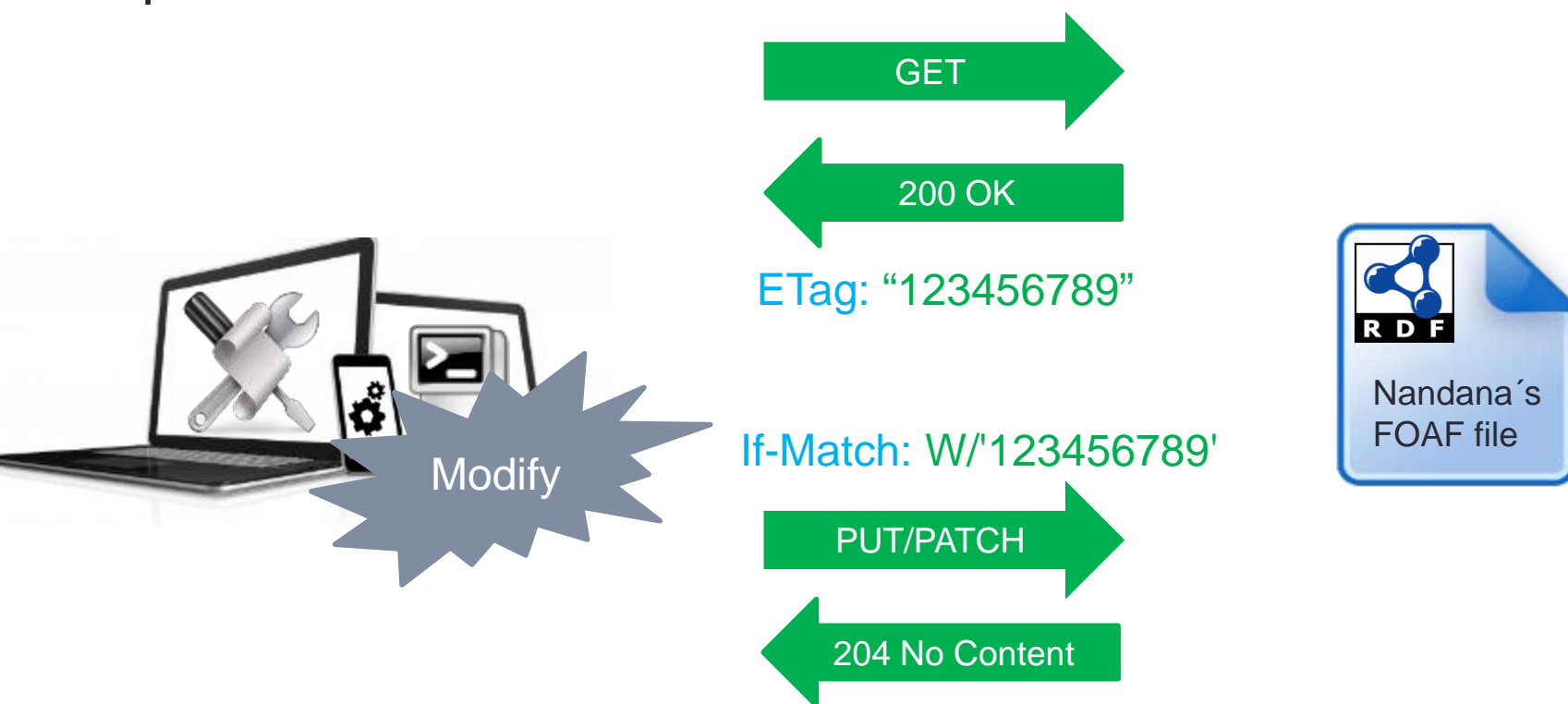
```
...
```

# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
  - Paging
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
  - Creating Linked Data Platform containers
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)

# Modifying a Linked Data resource

- uses HTTP PUT or PATCH operations
- encourages conditional requests to avoid “lost update” problem



# Modifying a Linked Data resource (I)

## GET on LDPR

```
GET /nandana HTTP/1.1  
Host: example.org  
Accept: text/turtle
```

 GET

```
HTTP/1.1 200 OK  
Content-Type: text/turtle; charset=UTF-8  
Link: <http://www.w3.org/ns/ldp#Resource>; rel="type"  
ETag: "123456789"  
Allow: OPTIONS,HEAD,GET,PUT,PATCH  
Accept-Patch: text/ldpatch  
Content-Length: 250
```

 200 OK

```
@prefix dc: <http://purl.org/dc/terms/> .  
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
```

```
<http://example.org/nandana> a foaf:PersonalProfileDocument;  
  foaf:primaryTopic <http://example.org/nandana#me> ;  
  dc:title 'Nandana's FOAF file' .
```

```
<http://example.org/nandana#me> a foaf:Person;  
  foaf:name 'Nandana Mihindukulasooriya';  
  foaf:currentProject <http://www.seals-project.eu/> .
```

# Modifying a Linked Data resource (II)

## PUT on LDPR

PUT /nandana HTTP/1.1

Host: example.org

If-Match: W/'123456789'

Content-Type: text/turtle

PUT

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

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

<http://example.org/nandana> a foaf:PersonalProfileDocument;  
 foaf:primaryTopic <http://example.org/nandana#me> ;  
 dc:title 'Nandana's FOAF file' .

<http://example.org/nandana#me> a foaf:Person;  
 foaf:name 'Nandana Mihindukulasooriya';  
 foaf:currentProject <http://www.ldap4j.org/> .

HTTP/1.1 204 No Content

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

204 No Content



# Modifying a Linked Data resource

## PUT on LDPR

PUT /nandana HTTP/1.1

Host: example.org

If-Match: W/'123456789'

Content-Type: text/turtle

uses conditional requests

PUT

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

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

<http://example.org/nandana> a foaf:PersonalProfileDocument;  
foaf:primaryTopic <http://example.org/nandana#me> ;  
dc:title 'Nandana's FOAF file' .

<http://example.org/nandana#me> a foaf:Person;  
foaf:name 'Nandana Mihindukulasooriya';  
**foaf:currentProject** <http://www.ldap4j.org/> .

Modify

HTTP/1.1 204 No Content

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

204 No Content

# Modifying a Linked Data resource

## PATCH on LDPR

PATCH /nandana HTTP/1.1

Host: example.org

If-Match: W/'123456789'

Content-Type: text/ldpatch

PATCH

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

Delete <http://example.org/nandana#me> foaf:currentProject <http://www.seals-project.eu/>.

Add <http://example.org/nandana#me> foaf:currentProject <http://www.ldap4j.org/>.

HTTP/1.1 204 No Content

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

204 No Content

# Modifying a Linked Data resource

## PATCH on LDPR

PATCH /nandana HTTP/1.1

Host: example.org

If-Match: W/'123456789'

Content-Type: text/ldpatch

uses conditional requests

uses LDP Patch format

PATCH

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

Delete <http://example.org/nandana#me> foaf:currentProject <http://www.seals-project.eu/> .

Add <http://example.org/nandana#me> foaf:currentProject <http://www.ldap4j.org/> .

HTTP/1.1 204 No Content

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

204 No Content

# Modifying a Linked Data resource

## Constraint validation errors

HTTP/1.1 422 Unprocessable Entity

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

Link: <http://example.org/constraints/101>; rel='http://www.w3.org/ns/ldp#constrainedBy'

← 422 Unprocessable Entity

# Modifying a Linked Data resource

## Constraint validation errors

HTTP/1.1 422 Unprocessable Entity

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

publishes any constraints on create/update

Link: <http://example.org/constraints/101>; rel='http://www.w3.org/ns/ldp#constrainedBy'

- constrainedBy link provides any constraints on the update

GET /constraints/101 HTTP/1.1

Host: example.org

Accept: \*/\*

GET

HTTP/1.1 200 OK

Content-Type: text/plain; charset=UTF-8

Content-Length: 150

constraints can be machine readable  
(e.g., Shape Expressions language)  
or in natural language

200 OK

When modifying the current project of a person, the project must be preserved using the foaf:pastProject property. Suggested Update:

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

Delete <http://example.org/nandana#me> foaf:currentProject <http://www.seals-project.eu/>.

Add <http://example.org/nandana#me> foaf:currentProject <http://www.ldap4j.org/>.

Add <http://example.org/nandana#me> foaf:pastProject <http://www.seals-project.eu/>.

# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
  - Paging
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)

# LDP Basic Container

- Allows creation of new resources through POST
- Maintains containment triples

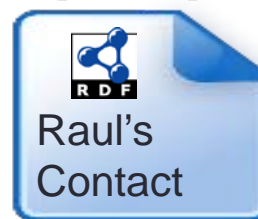
	Subject	Predicate	Object
Pattern	LDPC-URI	ldp:contains	document-URI
Example	<a href="http://example.org/nandana/contacts/">http://example.org/nandana/contacts/</a>	ldp:contains	<a href="http://example.org/nandana/contacts/raul">http://example.org/nandana/contacts/raul</a>

<http://example.org/nandana/contacts/>



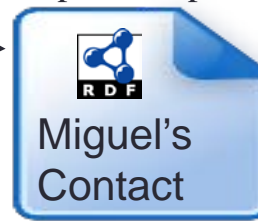
ldp:contains

<http://example.org/nandana/contacts/raul>



ldp:contains

<http://example.org/nandana/contacts/miguel>



# LDP Basic Container

```
GET /nandana/contacts/ HTTP/1.1  
Host: example.org  
Accept: text/turtle
```

 GET

```
HTTP/1.1 200 OK  
Content-Type: text/turtle; charset=UTF-8  
Link: <http://www.w3.org/ns/ldp#BasicContainer>; rel='type', <http://www.w3.org/ns/ldp#Resource>;  
rel='type'  
ETag: W/'123456711'  
Allow: OPTIONS,HEAD,GET,POST  
Accept-Post: text/turtle  
Content-Length: 250
```

 200 OK

```
@prefix dc: <http://purl.org/dc/terms/> .  
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
@prefix ldp: <http://www.w3.org/ns/ldp#>.
```

```
<http://example.org/nandana/contacts/> a ldp:Container, ldp:BasicContainer;  
    dc:title 'Nandana's contact list';  
    ldp:contains <http://example.org/nandana/friends/raul>,  
                <http://example.org/nandana/friends/miguel> .
```



# LDP Basic Container

GET /nandana/contacts/ HTTP/1.1

Host: example.org

Accept: text/turtle

advertises LDP Basic Container semantics

GET

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

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

ETag: W/'123456711'

Allow: OPTIONS,HEAD,GET,POST

Accept-Post: text/turtle

Content-Length: 250

200 OK

supports the POST method

advertises media types accepted by the POST method

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

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

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

has containment triples

<http://example.org/nandana/contacts/> a ldp:Container, ldp:BasicContainer;

dc:title 'Nandana's contact list';

ldp:contains <http://example.org/nandana/friends/raul> ,

<http://example.org/nandana/friends/miguel> .

# LDP Basic Container

## Creating an LDPR



# LDP Basic Container

## Creating an LDPR

```
POST /nandana/contacts/ HTTP/1.1
Host: example.org
Slug: fernando
Link: <http://www.w3.org/ns/ldp#Resource>; rel='type'
Content-Type: text/turtle
```

POST

```
@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .
```

```
<> a vcard:Individual;
  vcard:hasURL <http://example.org/fernando#me> ;
  vcard:hasEmail <mailto:fernado@example.com>;
  vcard:fn "Fernando Serena";
  vcard:hasTelephone [ a vcard:Home, vcard:Voice;
    vcard:hasValue <tel:+3465555555> ] .
```

# LDP Basic Container

## Creating an LDPR

POST /nandana/contacts/ HTTP/1.1

Host: example.org

Slug: fernando

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

Content-Type: text/turtle

@prefix vcard: <http://www.w3.org/2006/vcard/ns#>

```
<> a vcard:Individual;
vcard:hasURL <http://example.org/fernando#me>;
vcard:hasEmail <mailto:fernando@example.com>;
vcard:fn "Fernando Serena";
vcard:hasTelephone [ a vcard:Home, vcard:Voice;
vcard:hasValue <tel:+34655555555> ] .
```

POST

provides a hint for the new URI to be minted

requests the LDPR interaction model

a null URI is used to refer to the resource to be created

HTTP/1.1 201 Created

Location: http://example.org/nandana/contacts/fernando

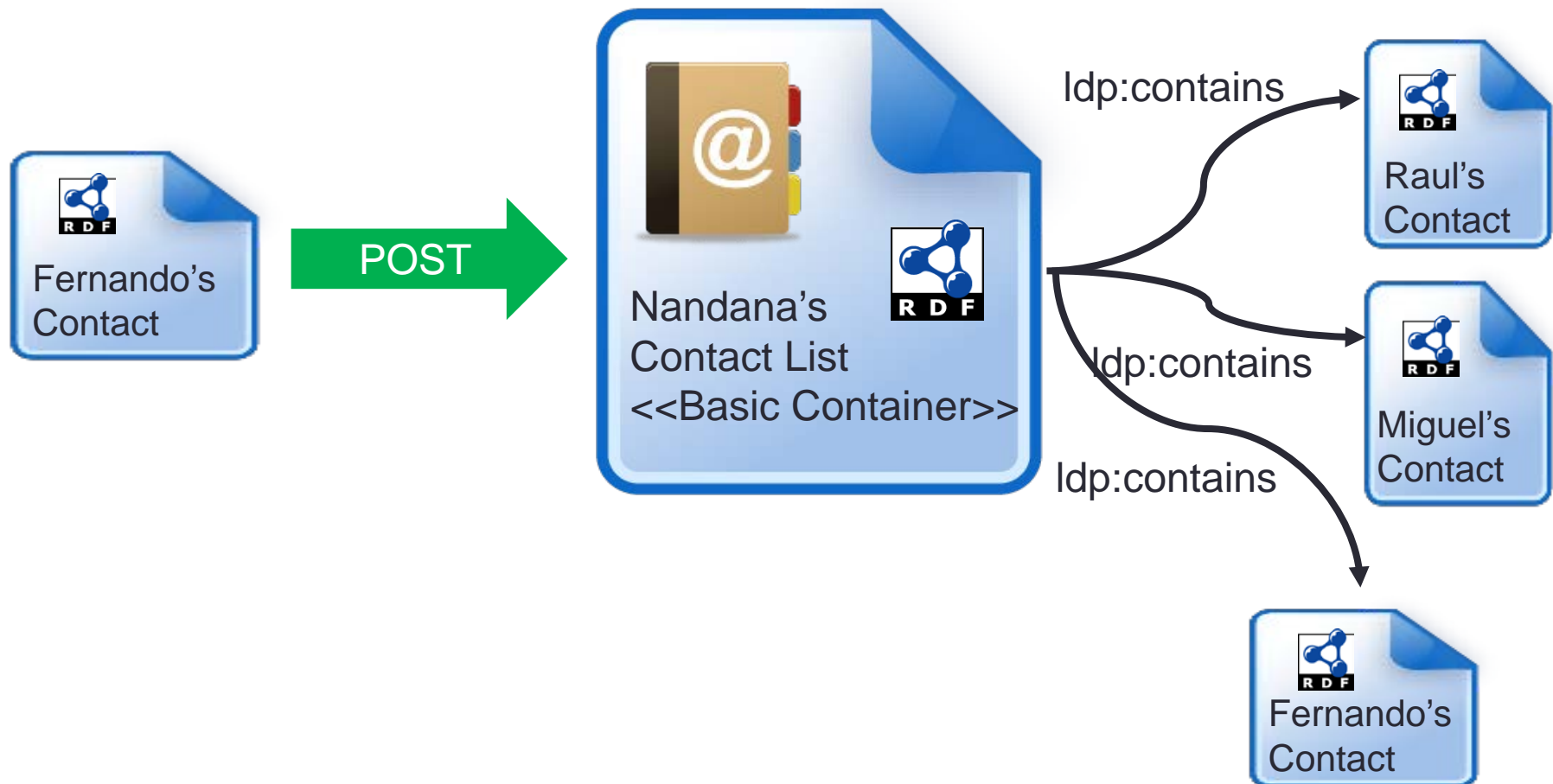
Link: <http://www.w3.org/ns/ldp#Resource>; rel='type'

Content-Length: 0

provides the URI of the newly created resource

# LDP Basic Container

## Creating an LDPR



# LDP Basic Container

## After the resource creation

```
GET /nandana/contacts/ HTTP/1.1
Host: example.org
Accept: text/turtle
```

GET

```
HTTP/1.1 200 OK
Content-Type: text/turtle; charset=UTF-8
Link: <http://www.w3.org/ns/ldp#BasicContainer>; rel='type', <http://www.w3.org/ns/ldp#Resource>;
rel='type'
ETag: W/'123456712'
Allow: OPTIONS,HEAD,GET,POST
Accept-Post: text/turtle
Content-Length: 250

@prefix dc: <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix ldp: <http://www.w3.org/ns/ldp#>.

<http://example.org/nandana/contacts/> a ldp:Container, ldp:BasicContainer;
    dc:title 'Nandana's contact list';
    ldp:contains <http://example.org/nandana/friends/raul>,
<http://example.org/nandana/friends/miguel> ,
    <http://example.org/nandana/friends/fernando> .
```

200 OK


# LDP Basic Container

## After the resource creation

```
GET /nandana/contacts/ HTTP/1.1
Host: example.org
Accept: text/turtle
```

GET

```
HTTP/1.1 200 OK
Content-Type: text/turtle; charset=UTF-8
Link: <http://www.w3.org/ns/ldp#BasicContainer>; rel='type', <http://www.w3.org/ns/ldp#Resource>;
rel='type'
ETag: W/'123456712'
Allow: OPTIONS,HEAD,GET,POST
Accept-Post: text/turtle
Content-Length: 250
```

200 OK

```
@prefix dc: <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix ldp: <http://www.w3.org/ns/ldp#> .
```

a containment triple is added for the newly created resource

```
<http://example.org/nandana/contacts/> a ldp:Container, ldp:BasicContainer;
  dc:title 'Nandana's contact list';
  ldp:contains <http://example.org/nandana/friends/raul>,
<http://example.org/nandana/friends/miguel> ,
  <http://example.org/nandana/friends/fernando> .
```

# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - **PUT to create**
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)



# Creating a Linked Data resource

## PUT on a non-existing URI

- Clients can decide the URI to be used

```
PUT /nandana/contacts/fernando HTTP/1.1
```

```
Host: example.org
```

```
Content-Type: text/turtle
```

PUT

```
@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .
```

```
< http://example.org/nandana/contacts/fernando> a vcard:Individual;  
vcard:hasURL <http://example.org/fernando#me> ;  
vcard:hasEmail <mailto:fernando@example.com>;  
vcard:fn "Fernando Serena";  
vcard:hasTelephone [ a vcard:Home, vcard:Voice;  
vcard:hasValue <tel:+34655555555> ] .
```

```
HTTP/1.1 204 No Content
```

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

204 No Content

# Creating a Linked Data resource PUT on a non-existing URI

- Clients can decide the URI to be used

PUT /nandana/contacts/fernando HTTP/1.1

Host: example.org

Content-Type: text/turtle

the URI for the resource to be created

PUT

@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .

```
< http://example.org/nandana/contacts/fernando> a vcard:Individual;
vcard:hasURL <http://example.org/fernando#me> ;
vcard:hasEmail <mailto:fernando@example.com>;
vcard:fn "Fernando Serena";
vcard:hasTelephone [ a vcard:Home, vcard:Voice;
vcard:hasValue <tel:+34655555555> ] .
```

HTTP/1.1 204 No Content

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

204 No Content

# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)

# Deleting a Linked Data document

```
DELETE /nandana/contacts/fernando HTTP/1.1  
Host: example.org
```

DELETE

```
HTTP/1.1 204 No Content  
Link: <http://www.w3.org/ns/ldp#Resource>; rel='type'
```

204 No Content

- Corresponding containment triples will be removed from the containers upon deletion

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
@prefix ldp: <http://www.w3.org/ns/ldp#>.
```

```
<http://example.org/nandana/contacts/> ldp:contains  
    <http://example.org/nandana/contacts/fernando> .
```

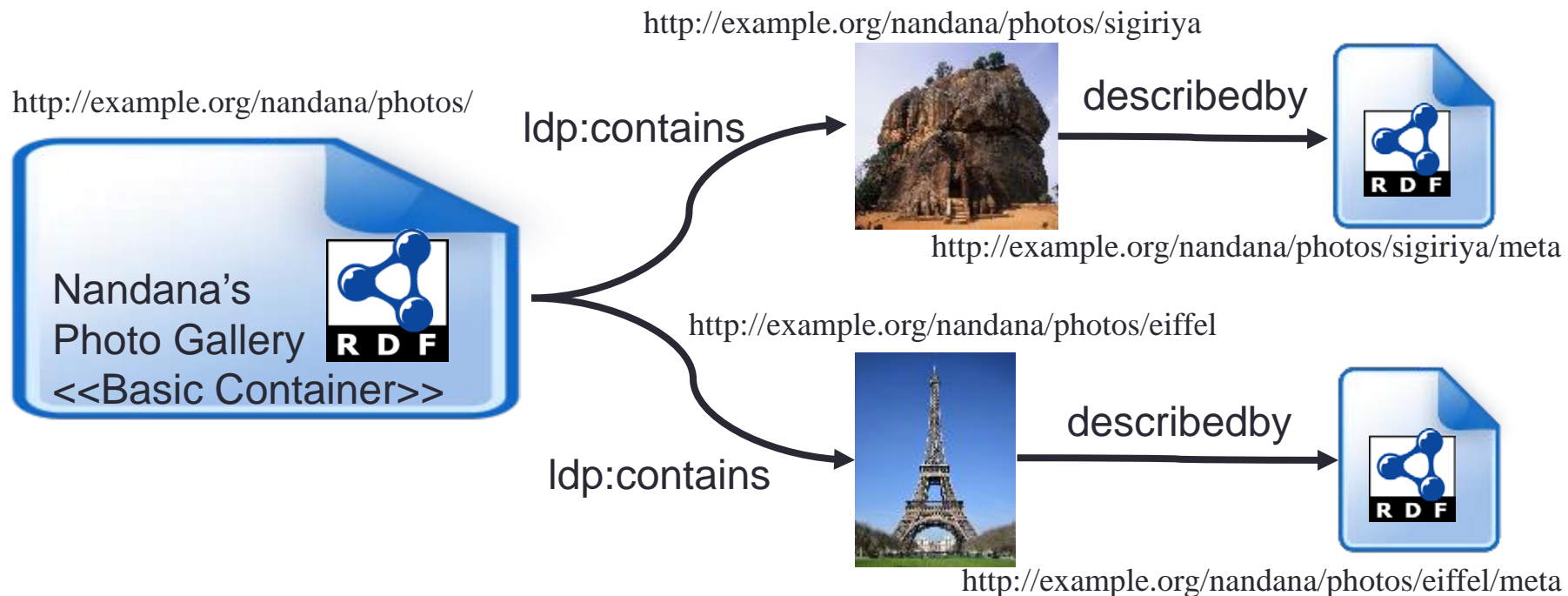
# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)

# Managing non-RDF resources

## A container for managing photos

- LDPRs can be non-RDF sources (LDP-NR)
- An LDP-NR source typically has an associated LDP-RS to contain data about the LDP-NR



# Managing non-RDF resources

## A container for managing photos

OPTIONS /nandana/photos/ HTTP/1.1  
Host: example.org

OPTIONS

HTTP/1.1 204 No Content  
Link: <http://www.w3.org/ns/ldp#BasicContainer>; rel='type', <http://www.w3.org/ns/ldp#Resource>; rel='type'  
Allow: OPTIONS,HEAD,GET,POST  
Accept-Post: image/png, image/jpeg

204 No Content

# Managing non-RDF resources

## A container for managing photos

OPTIONS /nandana/photos/ HTTP/1.1  
Host: example.org

OPTIONS

HTTP/1.1 204 No Content  
Link: <http://www.w3.org/ns/ldp#BasicContainer>; rel='type', <http://www.w3.org/ns/ldp#BinaryResource>; rel='type'  
Allow: OPTIONS,HEAD,GET,POST  
Accept-Post: image/png, image/jpeg

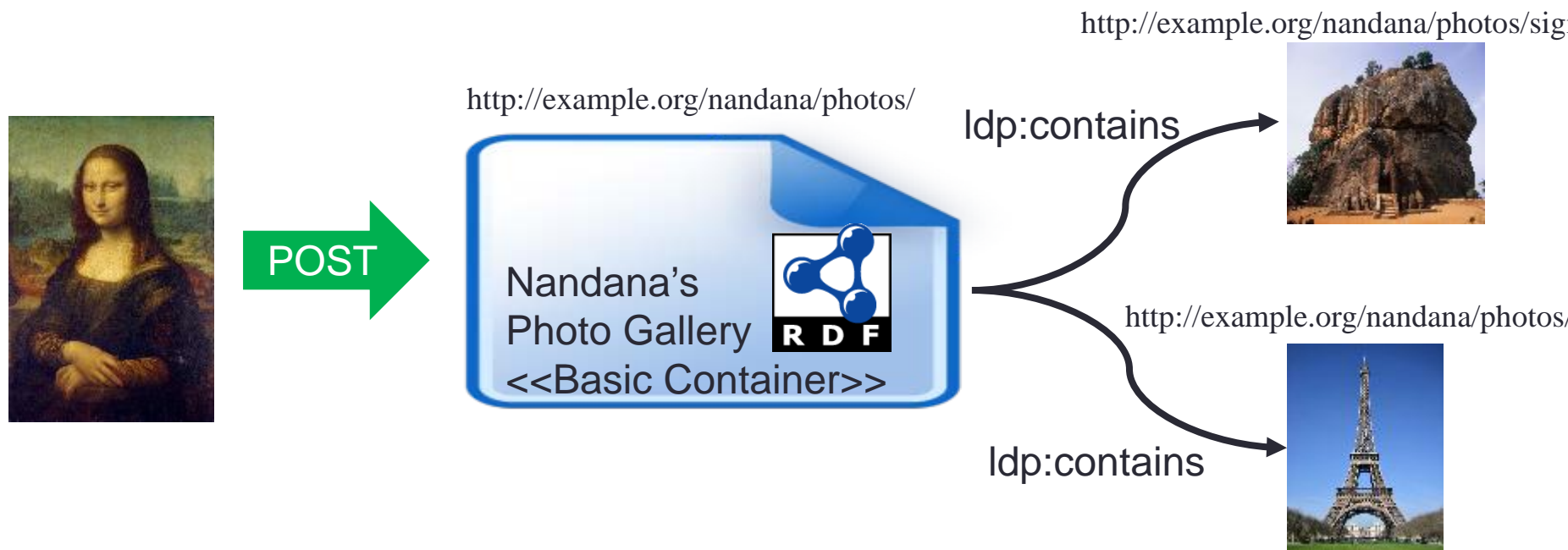
204 No Content

advertises support for non-RDF (binary)  
resource creation



# Managing non-RDF resources

## Creating a new binary resource



# Managing non-RDF resources

## Creating a new binary resource

```
POST /nandana/photos/ HTTP/1.1
```

```
Host: example.org
```

```
Slug: monalisa
```

```
Content-Type: image/png
```

```
Content-Length: 1020
```

 POST

```
### binary content of the image ###
```

```
HTTP/1.1 201 Created
```

```
Location: http://example.org/nandana/photos/monalisa
```

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

```
Link: <http://example.org/nandana/photos/monalisa/meta>; rel="describedby"
```

```
Content-Length: 0
```

 201 Created

# Managing non-RDF resources

## Creating a new binary resource

```
POST /nandana/photos/ HTTP/1.1
```

```
Host: example.org
```

```
Slug: monalisa
```

```
Content-Type: image/png
```

```
Content-Length: 1020
```

POST

```
### binary content of the image ###
```

provides the URI of the newly created resource

```
HTTP/1.1 201 Created
```

```
Location: http://example.org/nandana/photos/monalisa/
```

```
Link: <http://www.w3.org/2007/11/02/rdf-datatype#xhtml>
```

```
rel="type"
```

```
Link: <http://example.org/nandana/photos/monalisa/meta>; rel="describedby"
```

```
Content-Length: 0
```

provides the URI of the newly created resource  
advertises it is a binary resource  
contains the data about the binary resource

# Managing non-RDF resources

## Looking up the new binary resource

```
GET /nandana/photos/monalisa HTTP/1.1  
Host: example.org  
Accept: image/png
```

 GET

```
HTTP/1.1 200 OK  
Content-Type: image/png  
Link: <http://www.w3.org/ns/ldp#NonRDFSource>; rel="type", <http://www.w3.org/ns/ldp#Resource>;  
rel="type"  
Link: <http://example.org/nandana/photos/monalisa/meta>; rel="describedby"  
ETag: "123456790"  
Allow: OPTIONS,HEAD,GET, DELETE  
Content-Length: 1020
```

 200 OK

```
### binary content of the image ###
```

# Managing non-RDF resources

## Looking up the new binary resource

```
GET /nandana/photos/monalisa HTTP/1.1
Host: example.org
Accept: image/png
```

GET

```
HTTP/1.1 200 OK
Content-Type: image/png
Link: <http://www.w3.org/ns/ldp#NonRDFSource>; rel="type",
      <http://www.w3.org/ns/ldp#Resource>; rel="type"
Link: <http://example.org/nandana/photos/monalisa/meta>; rel="describedby"
ETag: "123456790"
Allow: OPTIONS,HEAD,GET, DELETE
Content-Length: 1020
```

200 OK

provides the URI of an associated LDP-RS that contains the data about the binary resource

```
### binary content of the image ###
```

# Managing non-RDF resources

## Looking up the associated LDP-RS

```
GET /nandana/photos/monalisa/meta HTTP/1.1  
Host: example.org  
Accept: text/turtle
```

 GET

```
HTTP/1.1 200 OK  
Content-Type: text/turtle  
Link: <http://www.w3.org/ns/ldp#Resource>; rel="type"  
ETag: "123456793"  
Allow: OPTIONS,HEAD,GET, PUT, DELETE
```

 200 OK

```
prefix wdrr: <http://www.w3.org/2007/05/powder-s#> .  
@prefix dcterms: <http://purl.org/dc/terms/> .
```

```
<http://example.org/nandana/photos/monalisa> wdrr:describedby  
    <http://example.org/nandana/photos/monalisa/meta> ;  
dcterms:creator <http://example.org/nandana#me>;  
dcterms:created "2013-05-05T10:00"^^xsd:dateTime .
```

# Managing non-RDF resources

## Looking up the associated LDP-RS

```
GET /nandana/photos/monalisa/meta HTTP/1.1
Host: example.org
Accept: text/turtle
```

 GET

```
HTTP/1.1 200 OK
Content-Type: text/turtle
Link: <http://www.w3.org/ns/ldp#Resource>; rel="type"
ETag: "123456793"
Allow: OPTIONS,HEAD,GET, PUT, DELETE
```

 200 OK

```
prefix wdrrs: <http://www.w3.org/2007/06/wdrrs#> .
@prefix dcterms: <http://purl.org/dc/terms/> .
```

provides metadata about the binary resource

```
<http://example.org/nandana/photos/monalisa> wdrrs:describedby
    <http://example.org/nandana/photos/monalisa/meta> ;
dcterms:creator <http://example.org/nandana#me>;
dcterms:created "2013-05-05T10:00"^^xsd:dateTime .
```

# Few simple scenarios

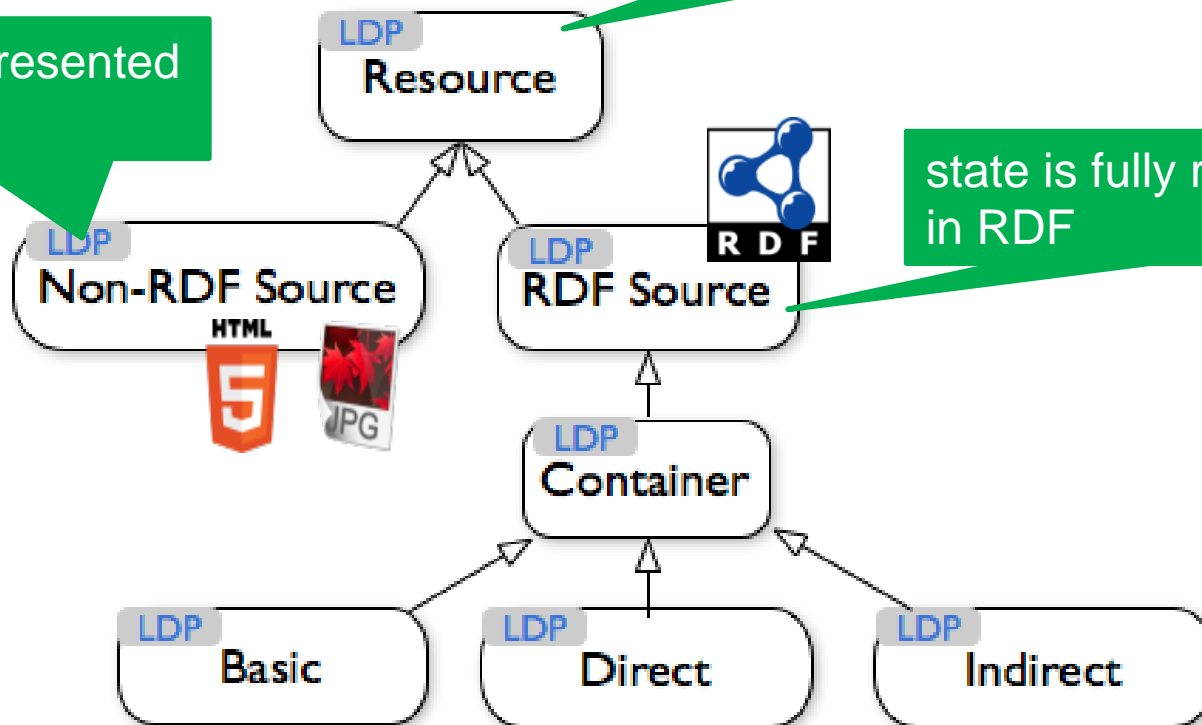
- Looking up a Linked Data resource (GET on LDPR)
  - Paging
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- **Types of LDP Containers**
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)



# LDP Terminology

state is **NOT** represented  
in RDF

conforms to the lifecycle patterns  
and conventions defined in LDP



state is fully represented  
in RDF

a special type of RDF Source that acts as

- an enumeration of a collection of linked documents
- a creation factory

# Isn't LDP Basic Containers enough?

- For most use cases, may be.
- There are some limitations
  - can't use domain-specific relationships
    - containment triples have a fixed pattern for <s, p, o> of the triple
  - containment triples can only say relationships among two web resources (the LDPC and resources created using it)
    - some use cases require relationships between resources other than the LDP Container itself
    - some use cases require relationship among real world resource (in contrast to information resources / documents)
- LDP Direct/Indirect containers introduce membership triples to address aforementioned limitations

# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
  - Paging
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)

# LDP Direct Container

- Maintains the containment triple similar to the basic containers
- Maintains membership triples
- Can use domain vocabularies

	Subject	Predicate	Object
Pattern	membership-constant-URI	membership-predicate	member-derived-URI
Example	http://example.org/nandana#me	foaf:knows	http://example.org/nandana/contacts/raul

defined in the container

# LDP Direct Container

<http://example.org/nandana/contacts/>

<http://example.org/nandana/contacts/raul>



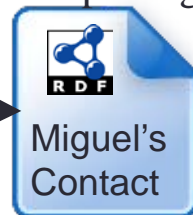
ldp:contains



foaf:knows

<http://example.org/nandana/contacts/miguel>

ldp:contains

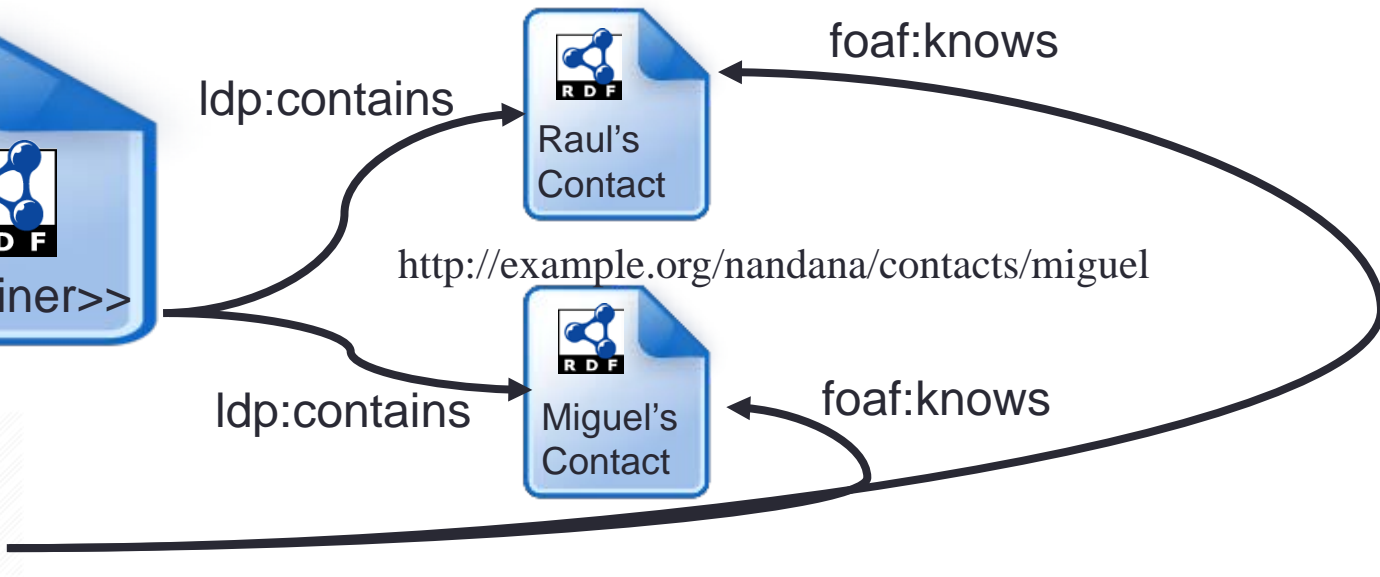


foaf:knows



Nandana

<http://example.org/nandana#me>



# Direct Containers

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

ETag: W/'123456711'

Allow: OPTIONS,HEAD,GET,POST

Accept-Post: text/turtle

Content-Length: 250

200 OK

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

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

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

<http://example.org/nandana/friends/> a ldp:Container, ldp:DirectContainer;

ldp:membershipResource <http://example.org/nandana#me>;

ldp:hasMemberRelation foaf:knows;

dc:title 'Nandana's contact list';

ldp:contains <http://example.org/nandana/friends/raul> ,

<http://example.org/nandana/friends/miguel> .

<http://example.org/nandana#me> foaf:knows <http://example.org/nandana/friends/raul> ,

<http://example.org/nandana/friends/miguel> .

# Direct Containers

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

ETag: W/'123456711'

Allow: OPTIONS,HEAD,GET,POST

Accept-Post: text/turtle

Content-Length: 250

200 OK

advertises LDP Basic Container interaction model

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

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

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

defines the membership-constant-URI

<http://example.org/nandana/friends/> a ldp:Container, ldp:DirectContainer;

ldp:membershipResource <http://example.org/nandana#me>;

ldp:hasMemberRelation foaf:knows;

defines the membership predicate

dc:title 'Nandana's contact list';

ldp:contains <http://example.org/nandana/friends/raul>;

<http://example.org/nandana/friends/miguel> .

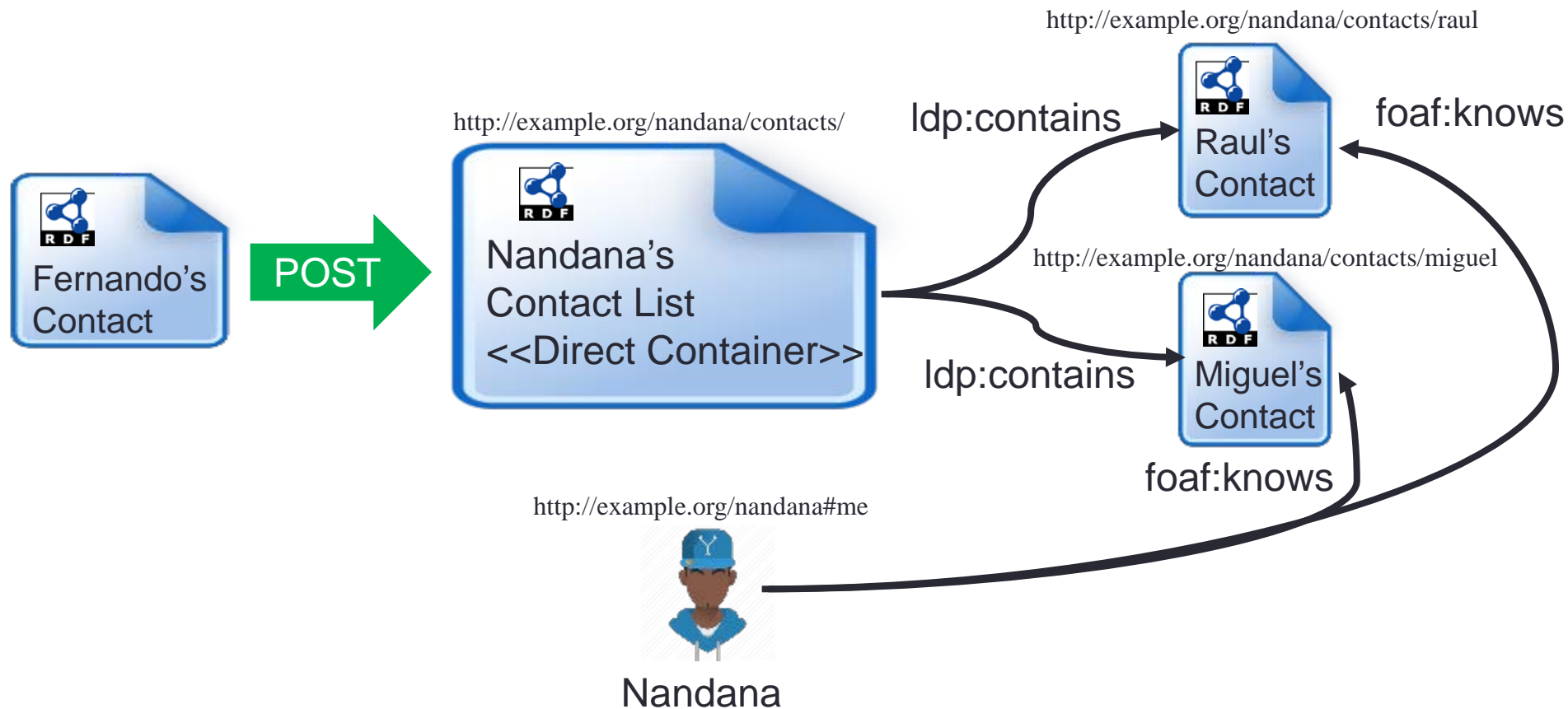
contains membership triples

<http://example.org/nandana#me> foaf:knows <http://example.org/nandana/friends/raul>;

<http://example.org/nandana/friends/miguel> .

# LDP Basic Container

## Creating an LDPR





# LDP Direct Container

## Creating an LDPR

```
POST /nandana/contacts/ HTTP/1.1
Host: example.org
Slug: fernando
Link: <http://www.w3.org/ns/ldp#Resource>; rel='type'
Content-Type: text/turtle
```

 POST

```
@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .
```

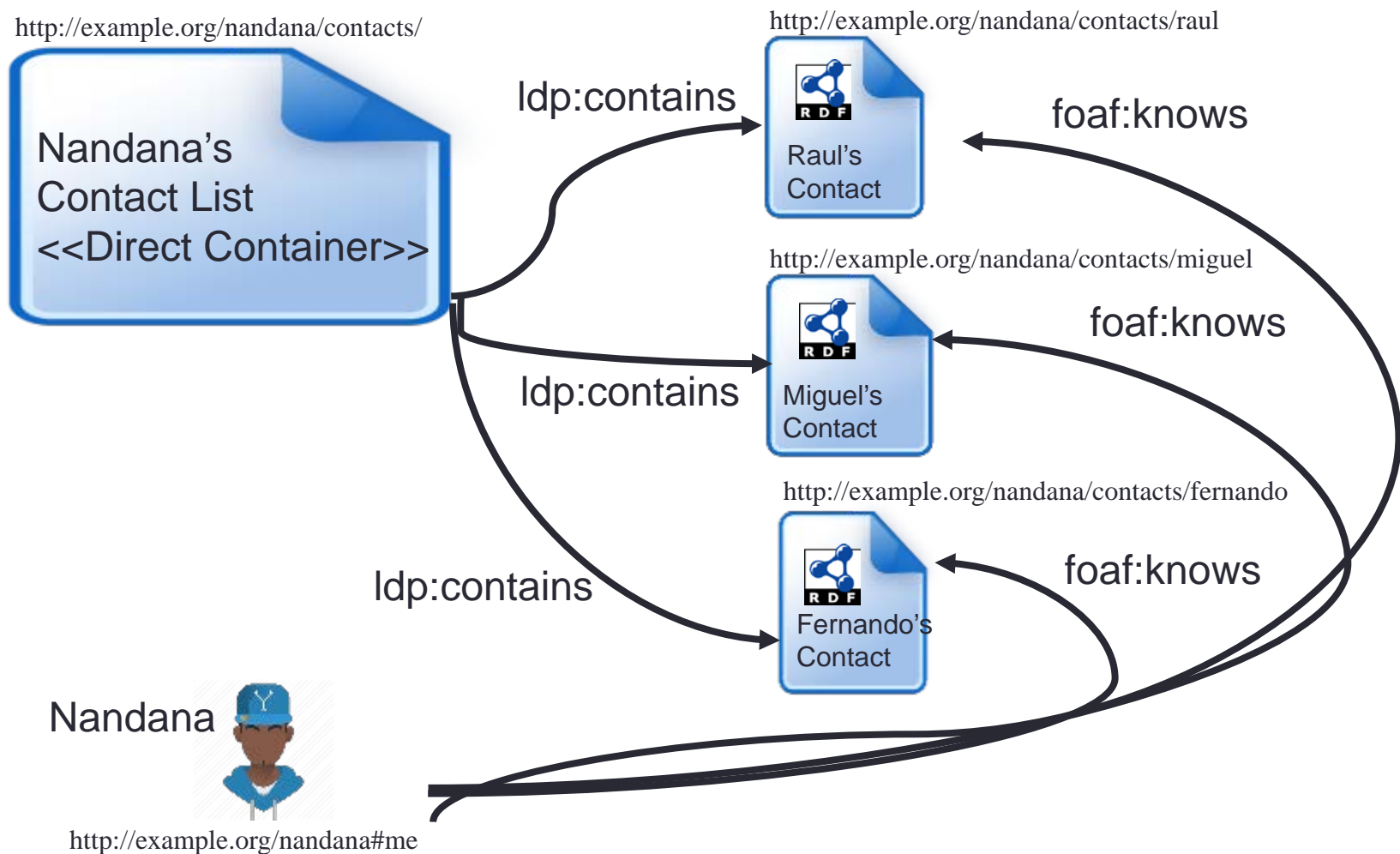
```
<> a vcard:Individual;
  vcard:hasURL <http://example.org/fernando#me> ;
  vcard:hasEmail <mailto:fernando@example.com>;
  vcard:fn "Fernando Serena";
  vcard:hasTelephone [ a vcard:Home, vcard:Voice;
    vcard:hasValue <tel:+34655555555> ] .
```

```
HTTP/1.1 201 Created
Location: http://example.org/nandana/contacts/fernando
Link: <http://www.w3.org/ns/ldp#Resource>; rel='type'
Content-Length: 0
```

 201 Created

# Direct Containers

## After the resource creation



# Direct Containers

## After the resource creation

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

ETag: W/'123456711'

Allow: OPTIONS,HEAD,GET,POST

Accept-Post: text/turtle

Content-Length: 250

200 OK

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

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

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

<http://example.org/nandana/friends/> a ldp:Container, ldp:DirectContainer;

ldp:membershipResource <http://example.org/nandana#me>;

ldp:hasMemberRelation foaf:knows;

dc:title 'Nandana's contact list';

ldp:contains <http://example.org/nandana/friends/raul> ,

<http://example.org/nandana/friends/miguel> , <http://example.org/nandana/friends/fernando> .

<http://example.org/nandana#me> foaf:knows <http://example.org/nandana/friends/raul> ,

<http://example.org/nandana/friends/miguel> , <http://example.org/nandana/friends/fernando> .

# Direct Containers

## After the resource creation

```
HTTP/1.1 200 OK
Content-Type: text/turtle; charset=UTF-8
Link: <http://www.w3.org/ns/ldp#DirectContainer>; rel='type', <http://www.w3.org/ns/ldp#Resource>;
rel='type'
ETag: W/'123456711'
Allow: OPTIONS,HEAD,GET,POST
Accept-Post: text/turtle
Content-Length: 250
```

 200 OK

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

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

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

a containment triple is added for the newly created resource

<http://example.org/nandana/friends/> a ldp:Container, ldp:DirectContainer;

ldp:membershipResource <http://example.org/nandana#me>;

ldp:hasMember <http://example.org/nandana/friends/raul>, <http://example.org/nandana/friends/miguel>, <http://example.org/nandana/friends/fernando> .

dc:title "Nandana's contact list" ;

ldp:contains <http://example.org/nandana/friends/raul>, <http://example.org/nandana/friends/miguel>, <http://example.org/nandana/friends/fernando> .

<http://example.org/nandana#me> foaf:knows <http://example.org/nandana/friends/raul>, <http://example.org/nandana/friends/miguel>, <http://example.org/nandana/friends/fernando> .

# Few simple scenarios

- Looking up a Linked Data resource (GET on LDPR)
  - Paging
- Modifying a Linked Data resource (PUT/PATCH on LDPR)
- Creating Linked Data resources
  - Basic Container (POST on LDP-BC)
  - PUT to create
- Deleting a Linked Data resource (DELETE on LDPR)
- Managing non-RDF resources
- Types of LDP Containers
  - Direct Container (POST on LDP-DC)
  - Indirect Container (POST on LDP-IC)

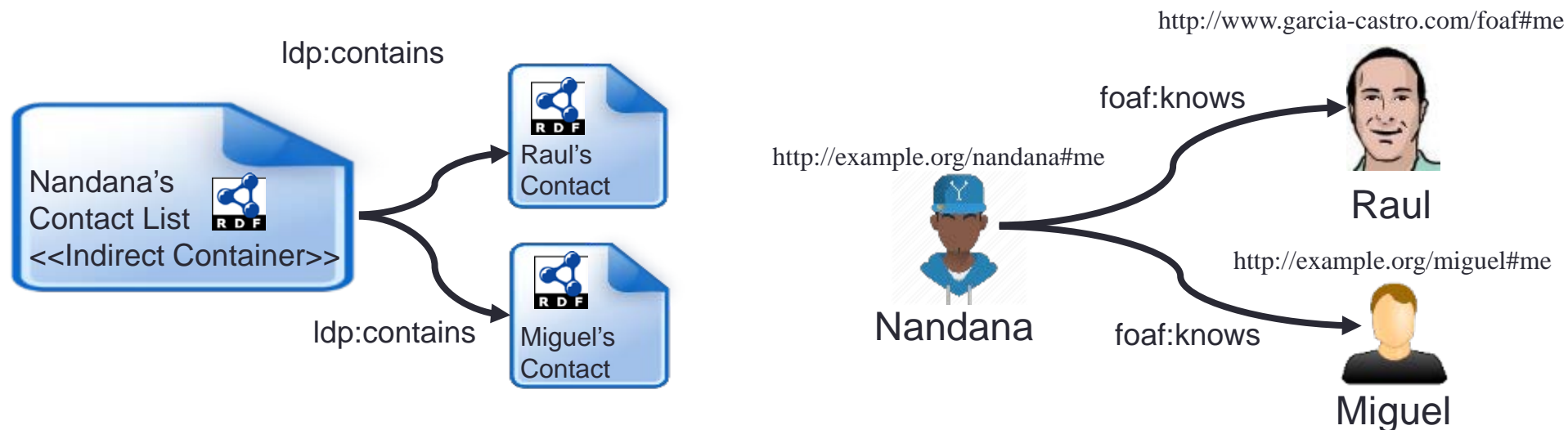
# LDP Indirect Container

- Maintains the containment triple similar to the basic containers
- Maintains membership triples

	Subject	Predicate	Object
Pattern	membership-constant-URI	membership-predicate	member-derived-URI
Example	<a href="http://example.org/nandana#me">http://example.org/nandana#me</a>	foaf:knows	<a href="http://example.org/nandana/contacts/raul">http://example.org/nandana/contacts/raul</a>

defined in the container

- Full flexibility with the membership triples



# Indirect Containers

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

ETag: W/"123456711"

Allow: OPTIONS,HEAD,GET,POST

Accept-Post: text/turtle

Content-Length: 250

← 200 OK

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

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

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

@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .

<http://example.org/nandana/friends/> a ldp:Container, ldp:IndirectContainer;

ldp:membershipResource <http://example.org/nandana#me>;

ldp:hasMemberRelation foaf:knows;

ldp:insertedContentRelation vcard:hasURL;

dc:title 'Nandana's contact list';

ldp:contains <http://example.org/nandana/friends/raul>, <http://example.org/nandana/friends/miguel> .

<http://example.org/nandana#me> foaf:knows <http://www.garcia-castro.com/foaf#me> ,

<http://example.org/miguel#me> .

# Indirect Containers

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

ETag: W/'123456711'

Allow: OPTIONS,HEAD,GET,POST

Accept-Post: text/turtle

Content-Length: 250

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

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

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

@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .

<http://example.org/nandana/friends/> a ldp:Container, ldp:IndirectContainer;

ldp:membershipResource <http://example.org/nandana#me>;

ldp:hasMemberRelation foaf:knows;

**ldp:insertedContentRelation vcard:hasURL;**

ldp:contains <http://example.org/nandana/friends/raul> ,

<http://example.org/nandana/friends/miguel> .

<http://example.org/nandana#me> foaf:knows <http://www.garcia-castro.com/foaf#me> ,

<http://example.org/miguel#me> .

200 OK

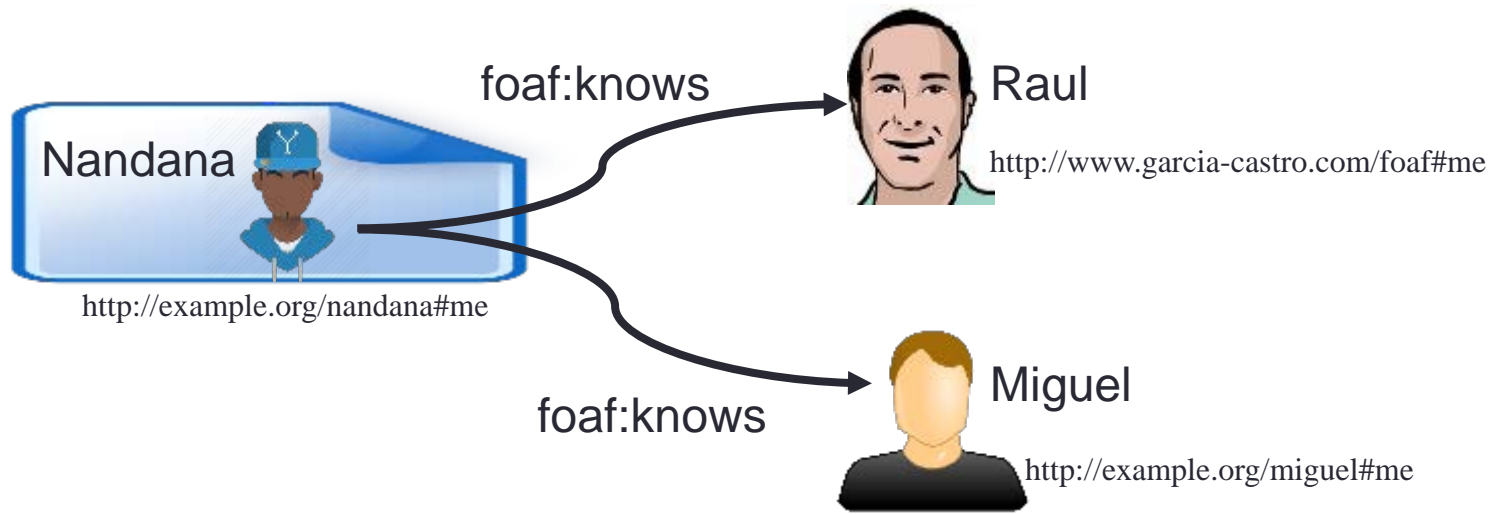
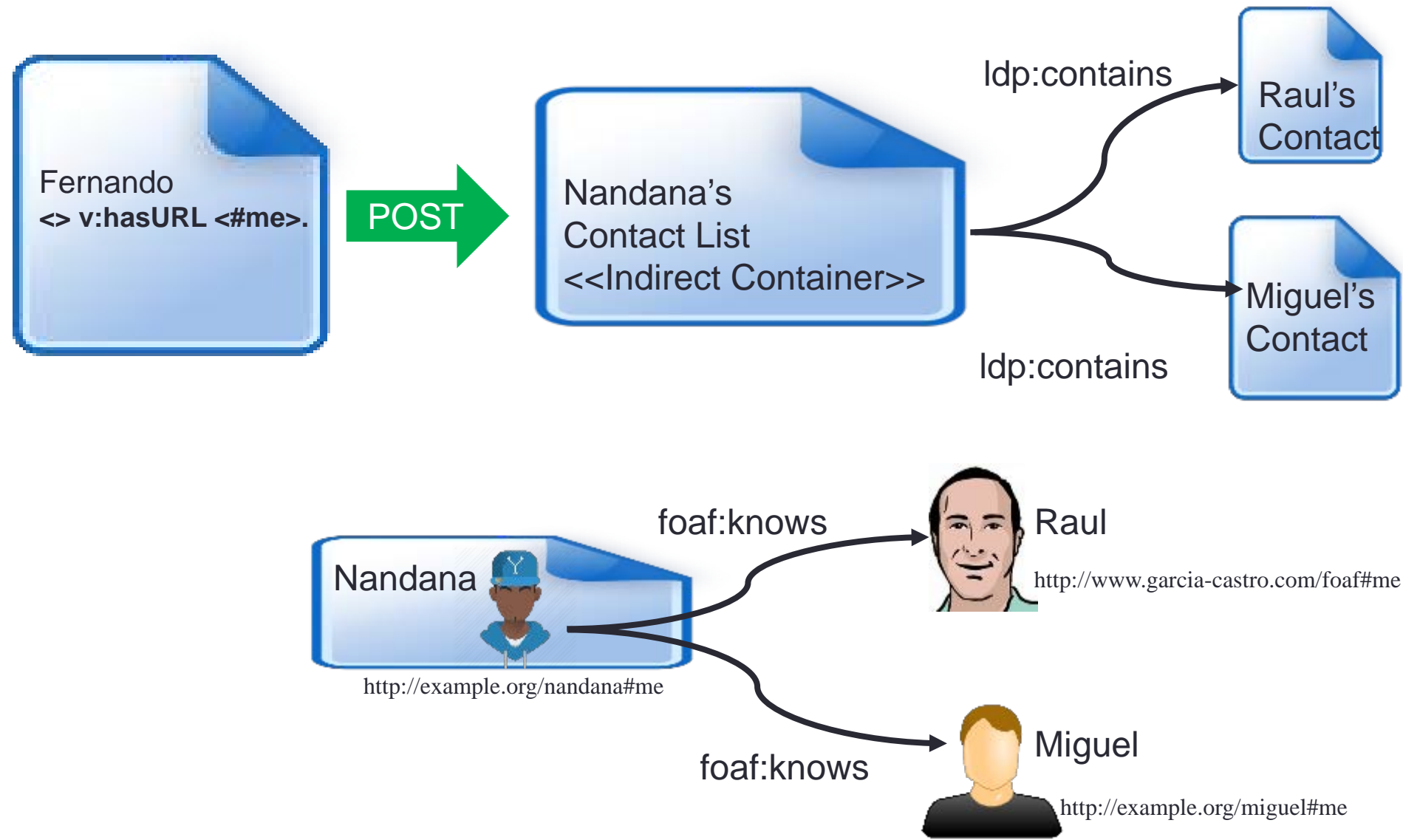
ldp:insertedContentRelation defines how to select the object of the membership triple

links to any resource



# LDP Indirect Container

## Creating an LDPR



# LDP Indirect Container

## Creating an LDPR

POST /nandana/contacts/ HTTP/1.1

Host: example.org

Slug: fernando

Content-Type: text/turtle

POST

@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .

<> a vcard:Individual;

vcard:hasURL <#me> ;

vcard:hasEmail <mailto:fernando@example.com>;

vcard:fn "Fernando Serena";

vcard:hasTelephone [ a vcard:Home, vcard:Voice;

vcard:hasValue <tel:+34655555555> ] .

HTTP/1.1 201 Created

Location: http://example.org/nandana/friends/fernando

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

Content-Length: 0

201 Created

# LDP Indirect Container

## Creating an LDPR

```
POST /nandana/contacts/ HTTP/1.1
```

```
Host: example.org
```

```
Slug: fernando
```

```
Content-Type: text/turtle
```

 POST

```
@prefix vcard: <http://www.w3.org/2006/vcard/ns#> .
```

```
<> a vcard:Individual;
```

```
vcard:hasURL <#me> ;
```

```
vcard:hasEmail <mailto:fernando@example.com>;
```

```
vcard:fn "Fernando Serena";
```

```
vcard:hasTelephone [ a vcard:Home, vcard:Voice;
```

```
vcard:hasValue <tel:+34655555555> ] .
```

POSTed resource contains a triple with the predicate defined in the `ldp:insertedContentRelation`

```
HTTP/1.1 201 Created
```

```
Location: http://example.org/nandana/friends/fernando
```

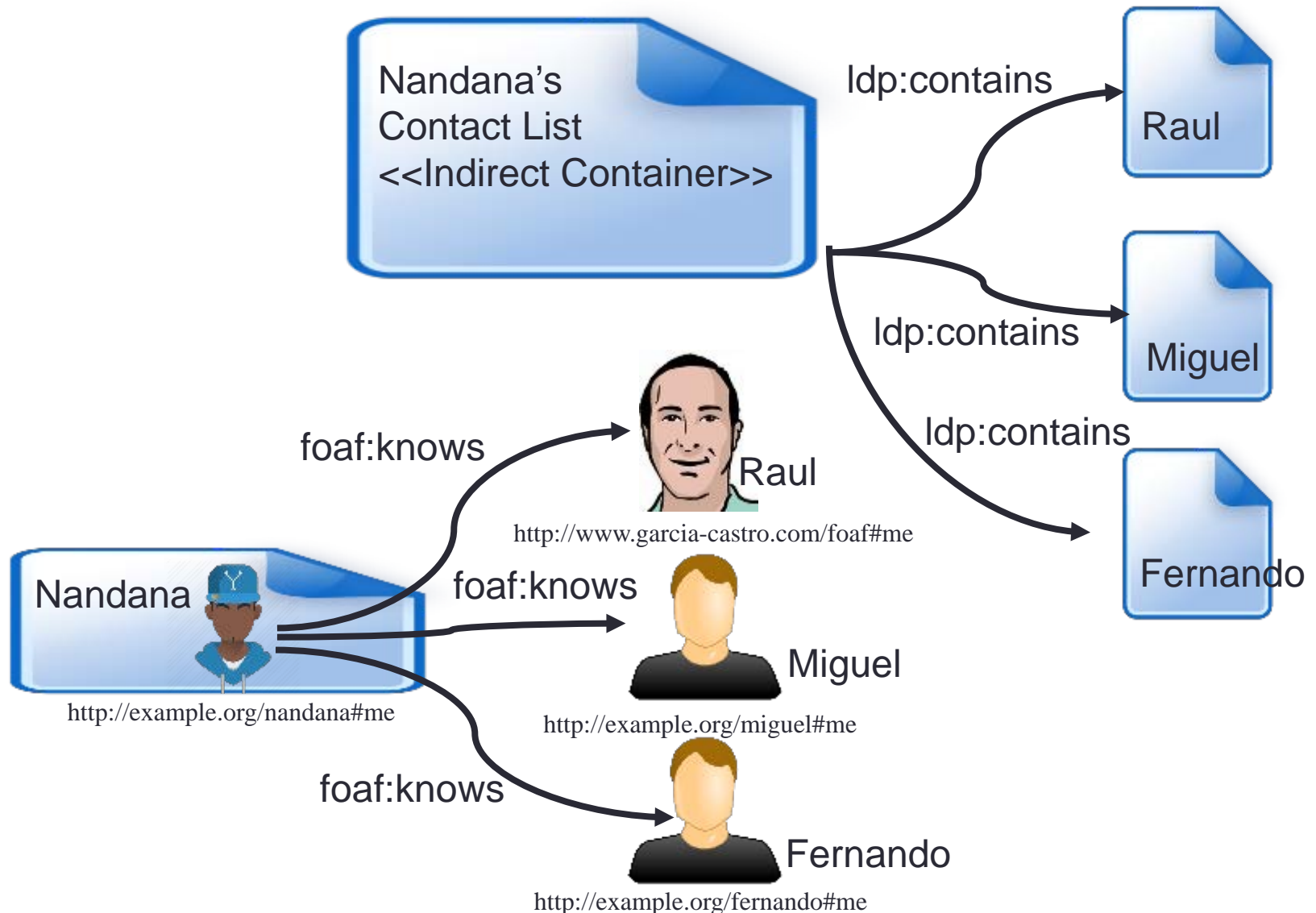
```
Link: <http://www.w3.org/ns/ldp#Resource>; rel='type'
```

```
Content-Length: 0
```

 201 Created

# Indirect Containers

## After the resource creation



# Indirect Containers

## After the resource creation

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

ETag: W/'123456711'

Allow: OPTIONS,HEAD,GET,POST

Accept-Post: text/turtle

Content-Length: 250

200 OK

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

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

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

<http://example.org/nandana/friends/> a ldp:Container, ldp:DirectContainer;

ldp:membershipResource <http://example.org/nandana#me>;

ldp:hasMemberRelation foaf:knows;

dc:title 'Nandana's friends';

ldp:contains <http://example.org/nandana/friends/raul> ,

<http://example.org/nandana/friends/miguel> , <http://example.org/nandana/friends/fernando> .

<http://example.org/nandana#me> foaf:knows <http://www.garcia-castro.com/foaf#me> ,

<http://example.org/miguel#me> <http://example.org/fernando#me> .

# Indirect Containers

## After the resource creation

HTTP/1.1 200 OK

Content-Type: text/turtle; charset=UTF-8

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

ETag: W/'123456711'

Allow: OPTIONS,HEAD,GET,POST

Accept-Post: text/turtle

Content-Length: 250

200 OK

@prefix dc: <http://purl.org/dc/terms/> .  
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
@prefix ldp: <http://www.w3.org/ns/ldp#> .

a containment triple is added for the newly created resource

<http://example.org/nandana/friends/> a ldp:Container, ldp:DirectContainer;  
ldp:membershipResource <http://example.org/nandana#me> .

a membership triple is added for the newly created resource

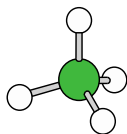
the object of the membership triple is selected based on ldp:insertedContentRelation

<http://example.org/nandana/friends/miguel>, <http://example.org/nandana/friends/fernando> .

<http://example.org/nandana#me> foaf:knows <http://www.garcia-castro.com/foaf#me>,  
<http://example.org/miguel#me> <http://example.org/fernando#me> .

# References

- Linked Data Platform 1.0
  - <http://www.w3.org/TR/ldp/>
- Linked Data Platform 1.0 Primer
  - <http://www.w3.org/TR/ldp-primer/>
- Linked Data Platform Best Practices and Guidelines
  - <http://www.w3.org/TR/ldp-bp/>
- Linked Data Platform Paging 1.0
  - <http://www.w3.org/TR/ldp-paging/>
- Linked Data Patch Format
  - <http://www.w3.org/TR/ldpatch/>



# Questions?

---

Miguel Esteban Gutiérrez,  
**Nandana Mihindukulasooriya**,  
Raúl García Castro,  
Asunción Gómez Pérez  
Center for Open Middleware / Ontology Engineering Group  
Universidad Politécnica de Madrid,  
Spain.