Linked Data Platform

Victor Saquicela, Boris Villazón-Terrazas
bvillazon@isoco.com
@boricles
Slides available at: http://www.slideshare.net/boricles/
Introduction

- LDP Goals and concepts
- LDP Implementations
- Apache Marmotta
Linked Data was defined by Tim Berners-Lee with the four principles

It is necessary a simple read-write Linked Data Architecture

Based on HTTP access to web resources that describe their state using RDF.
The goal of LDP is to define a specification required to allow the definition of a writeable Linked Data API.

There are two types of Linked Data Platform Resources (LDPRs)
- LDP RDF Sources (LDP-RSs)
- LDP Non-RDF Sources (LDP-NRs), e.g., HTML files, images, etc.

Resources respond to retrieval operations using HTTP GET
LDP goal & concepts

- Writeable aspects include
  - Creation of new resources (POST or PUT)
  - Updates (PUT or PATCH)
  - Deletion of resources

- LDP defines creation for a special kind of Resource called a Container (LDPC), which is able to respond to requests to create new resources.

- LDPC is a specialization of a LDP-RS representing a collections of links to LDPRSSs
LDP goal & concepts

- The simplest container is the Basic Container (LDP-BC), which defines the basic containment described using a generic vocabulary.

- A Direct Container is a specialisation of a Basic Container. Additional assertions called membership triples which use a domain-specific vocabulary are made by a Direct Container as part of the creation process.
Direct container membership triples can be about subjects other than Container resource. An example is a Photo management application where a Photo Container is used for the management of Photos, and where membership triples then express the relationship between a User and a Photo.
LDP goal & concepts
LDP current status

- W3C Candidate Recommendation

**Linked Data Platform 1.0**

W3C Candidate Recommendation 19 June 2014

This version:

Latest published version:
- [http://www.w3.org/TR/ldp/](http://www.w3.org/TR/ldp/)

Latest editor's draft:
- [http://www.w3.org/2012/ldp/hg/ldp.html](http://www.w3.org/2012/ldp/hg/ldp.html)

Test suite:
- [https://dvcs.w3.org/hg/ldpwg/raw-file/tip/tests/ldp-testsuite.html](https://dvcs.w3.org/hg/ldpwg/raw-file/tip/tests/ldp-testsuite.html)

Implementation report:
- [https://dvcs.w3.org/hg/ldpwg/raw-file/tip/tests/reports/ldp.html](https://dvcs.w3.org/hg/ldpwg/raw-file/tip/tests/reports/ldp.html)

Previous version:
- [http://www.w3.org/TR/2014/WD-ldp-20140311/](http://www.w3.org/TR/2014/WD-ldp-20140311/)

Editors:
- Steve Speicher, IBM Corporation
- John Arwe, IBM Corporation
- Ashok Malhotra, Oracle Corporation

Copyright © 2014 W3C® (MIT, ERCIM, Keio, Beihang), All Rights Reserved. W3C liability, trademark and document use rules apply.
LDP implementations

1. LDP4j (Client and Server)
2. Apache Marmotta (Client and Server)
3. Callimachus (Client and Server)
4. Carbon LDP (Server)
5. Eclipse Lyo (Client and Server, Software Lifecycle)
6. RWW.IO
7. node_ldp (Server)
8. OpenLink Data Spaces (Hosted Service; LDP Client & Server)
9. OpenLink Virtuoso (Data Server; LDP Client & Server)
10. rdflib.js (Client Library)
11. rdf.sh (Client)
12. RWW Play Server
13. Tabulator (Client)
14. TopBraid Live (Client and Server)
Apache Marmotta
Apache Marmotta is an Open Platform for Linked Data

The goal of Apache Marmotta is to provide an open implementation of a Linked Data Platform that can be used, extended and deployed easily by organizations who want to publish Linked Data or build custom applications on Linked Data.
Thank You!!
Linked Data Platform

Victor Saquicela, Boris Villazón-Terrazas
bvroleton@isoco.com
@boricles
Slides available at: http://www.slideshare.net/boricles/