

# Linked Data y Linked Data Platform

## Lenguajes y Tecnologías de la Web Semántica

Concepción de Sistemas de Información

2021



# Contenidos

- Ideas Básicas
- Los Principios
- Cómo Fabricar Linked Data

# Contenidos

- Ideas Básicas
- Los Principios
- Cómo Fabricar Linked Data

# Qué es Linked Data?

- Es una política de publicación de datos basada en 4 ppios.
  - Use URIs como nombres para las cosas.
  - Use URIs HTTP , de forma que se pueda acceder a las cosas.
  - Cuando alguien accede URI, devuelve información útil, usando standars (RDF, SPARQL).
  - Incluya links a otras URIs, de manera que se puedan descubrir más cosas.

# Contenidos

- Ideas Básicas
- Los Principios
- Cómo Fabricar Linked Data

# Principio 1: Use URIs como nombres para las cosas

- Sirven como identificadores globales, independientes de los contextos.
- Todas las “cosas” tiene una URI
  - Las cosas (cada persona, lugar, objeto real) del mundo real de las que hablo.
  - Las cosas que digo (documentos) de las cosas.
- Es saludable diferenciar los objetos de las cosas que digo de los objetos.

# Principio 2: Use URIs HTTP

- Al derreferenciar una URI, se debe acceder a lo que se necesita.
- Es diferente según quien lo pide:
  - Persona: HTML
  - Máquina: RDF
- Soluciones:
  - 303 URIs: Se basan en negociación de contenido.
  - Hash URIs: Se hace referencia a fragmentos.

# 303 URIs

- Si se derreferencia una URI de un “objeto real”, hay 4 pasos:
  - El cliente hace un GET con los cabezales adecuados:
    - `Accept: application/rdf+xml`  
si es una aplicación.
    - `Accept: text/html`  
si es un browser.
  - El servidor reconoce que no puede proveer la URI directamente, y devuelve una nueva URI con status `303 See Other`
  - El cliente hace el GET a la nueva dirección.
  - El servidor envía los datos con un status `200 OK`.

# Hash URIs

- La idea es que las URIs son fragmentos de un documento usando # .
- La derefferencia se resuelve en tres pasos:
  - El cliente hace un GET.
  - El servidor envía todo el documento.
  - El cliente obtiene (o se posiciona) en el fragmento referenciado.

# Comparación de las URIs

- 303 URIs
  - Se necesitan dos llamadas HTTP para resolver un único pedido. Esto puede llevarse a una sola llamada con programación.
  - Sólo trae lo pedido.
  - Se usa cuando hay un conjunto grande recursos de diferentes tamaños. Ej: dbpedia, linkedimdb.
- Hash URIs
  - Sólo se hace una llamada.
  - Trae todo el documento base.
  - Se usa cuando hay que referenciar a recursos que es viable representar como partes de un documento. Ej: Vocabularios (foaf,owl,rdf,rdfs,etc)

# Principio 3: Provea Información Util

- En parte resuelto por los mecanismos de negociación de contenido.
- Es importante usar estándares para representar la información:
  - HTML
  - RDF/XML
  - Turtle
  - RDFa

# Principio 4: Ponga Links a otras URIs

- Links de Identificación:
  - Se usan para identificar “cosas”.
  - Se basa en usar `owl:sameAs`.
  - Se fundamenta en permitir integración. Ej: la misma persona en diferentes redes sociales.
- Links de Relación:
  - Relaciona el recurso que lo contiene, con otro recurso.
  - Son los links “normales” pero con la propiedad que los relaciona.
  - Se fundamenta en mantener el conocimiento de las propiedades del recurso.

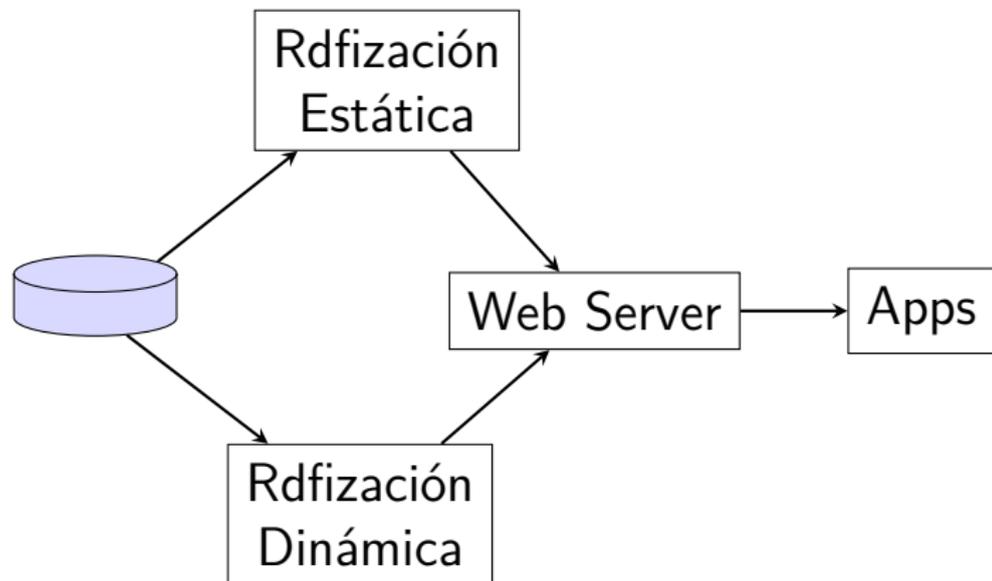
# Principio 4: Ponga Links a otras URIs

- Links entre Vocabularios:
  - Se usan en las definiciones de los vocabularios para relacionarlo con otros.
  - Hay varios tipos de links entre vocabularios.
  - se fundamente en que los vocabularios deben ser conocidos y derreferenciables para que otros los puedan usar.

# Contenidos

- Ideas Básicas
- Los Principios
- Cómo Fabricar Linked Data

# Proceso de Publicación



# Bibliografía

- Heath, T. y Bizer, C. 2011. Linked Data: Evolving the Web into a Global Data Space [en línea]. 1st. S.l.: Morgan & Claypool. ISBN 978-1-60845-431-0. Disponible en: <http://linkeddatabook.com/>.