Weaving repository contents into the Semantic Web

Pascal-Nicolas Becker | Technische Universität Berlin | SWIB14 | Bonn, December 1-3, 2014
Digital Repositories

Repositories are systems to safely store and publish digital objects and their descriptive metadata.

Not in the meaning of software repositories.

Examples:
• Digital archives
• Institutional repositories (preprints, postprints, open access publications, …)
• Digital image libraries
• Research data repositories
• …

More than 2500 Open Access repositories worldwide.

The data stored in repositories are particularly suited to be used in the Semantic Web:

- Metadata already exist in a structured form.
- They do not have to be generated or entered manually for publication as Linked Data.
- “Just” convert the data in RDF, add links and publish them respecting the Linked Data Principles.
“Although the WorldWideWeb still represents only a small fraction of the overall usage, this access mode is expected to become dominant in the near future.”

Paul Ginsparg 1994
Current data exchange with Repositories

- OAI-PMH (Open Archive Initiative – Protocol for Metadata Harvesting): de facto standard in the context of repositories
- But: limited to that context
- Google retired support for OAI-PMH in 2008 (used before as alternative to the sitemap protocol)
- “Just” an interface, not a format

- Linked Data is a generic, native way of data exchange, not only in the field of repositories
- Data published following the Linked Data Principles is self-descriptive
- Linked Data simplifies data exchange with repositories
Characteristics of repositories

• Different repositories may use different metadata schemas.
  ➢ Conversion must be highly configurable and extendable.
• Metadata may use already existing vocabularies (e.g. Dublin Core, LCSH, …).
  ➢ Convert metadata values to URIs / links.
• Repository contents change rarely (to be citable and reliable).
  Conversion may be time intensive.
  ➢ Convert data and store converted data in a cache.
• Repositories generate URIs that shall be used to address their content.
  ➢ Reuse those URIs, add content negotiation to them.
• Persistent Identifiers (handle, DOI, …) violate the Linked Data Principles.
  ➢ Use Persistent Identifiers in form of HTTP(S) URIs (http://dx.doi.org/…).
Extending Repositories

- Add a Triple Store.
  - Use it as cache for converted data.
  - Use it to provide a SPARQL endpoint.
- Add methods to convert data into RDF and to add links.
- Add a module to serve data as RDF serializations.
- Add content negotiation.
What do Repositories store?

“What Repositories are systems to safely store and publish digital objects and their descriptive metadata.”

- Digital objects
  - One or several files: Documents (PDF, Text, …), Tables (CSV, …), Images (PNG, Tiff …), Audio (Wave, …), Video, File Archives, …

- Descriptive metadata
  - Structured metadata as key – value: dc.title, dc.contributor.author, dc.description, dc.date.available, dc.subject.lcsh, dc.subject.ddc, …

- We can’t convert the files (technical problems, far too much work).
- But we can convert the metadata and link to the files!
Convert existing metadata to RDF

- Repository software can be extended to support more or other metadata fields.
- Dublin Core is used often, but there are other metadata schemas as well.

- Make the conversion highly configurable!
- Use RDF for the configuration (so all features of RDF can be used in the configuration easily).
- Use Reification to describe the results.
- Use Placeholders where necessary, e.g. URIs used by the repository.
- Use Regular Expressions to generate Literals and/or URIs from a metadata value.
- Create a vocabulary to write such configurations.
Example: DSpace Metadata RDF Mapping Vocabulary

http://digital-repositories.org/ontologies/dspace-metadata-mapping/

- One Mapping describes how to convert one metadata field in RDF.
- Can detect the metadata field by its name (key) and a regular expression used on its value.
- Creates one or several triples.
- Can use a placeholder for the URI of the object being converted currently.
- Can create Literals or Resources as needed.
- Can specify value types and language tags.
- Can use the language tag DSpace stores for some metadata fields.
- Can reuse the metadata value, of course.
- May use regular expressions to modify metadata values used as Literals or Resource URIs.
Example: DSpace Metadata RDF Mapping Vocabulary

@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix dm: <http://digital-repositories.org/ontologies/dspace-metadata-mapping/0.2.0#> .
@prefix : <#> .

:title
   dm:metadataName "dc.title" ;
   dm:creates [
      dm:subject dm:DSpaceObjectIRI ;
      dm:predicate dcterms:title ;
      dm:object dm:DSpaceValue ;
   ];
.
Example: DSpace Metadata RDF Mapping Vocabulary

:doi
  dm:metadataName „dc.identifier.doi“ ;
  dm:condition „^doi:“ ;
  dm:creates [
    dm:subject dm:DSpaceObjectIRI ;
    dm:predicate dc:identifier ;
    dm:object [
      a dm:ResourceGenerator ;
      dm:modifier [
        dm:matcher „^doi:(.*)$“ ;
        dm:replacement „http://dx.doi.org/$1“ ;
      ] ;
      dm:pattern „$DSpaceValue“ ;
    ] ;
  ] ;
Describing Repositories

• Beside converting metadata it is worth describing the repository itself.
• Who is running the repository? Does it have an OAI-PMH interface? Where can I find a SPARQL endpoint? How is the content structured? …
• A vocabulary to link to the digital objects (files) is needed as well.

• For DSpace, I created the DSpace Repository Ontology:
  http://digital-repositories.org/ontologies/dspace/
• A Digital Repositories Ontology would be great, describing repositories independent from the software used to create them.
  • A mapping between such an ontology and the DSpace Repository Ontology, the EPrints Ontology or any other would be great!
  • If you are interested in creating such an Ontology as well: please contact me.
Things to mention, even if they should be clear

• Reuse existing URIs wherever possible, don’t create your own URI if there already exists one.
  • E.g.: For classifications like the Library of Congress Subject Headings URIs already exists.
• Create URIs only for you own entities or if you have enough information.
  • Do not create URIs for authors unless you can distinguish different authors with the same name!
  • Think about whether the author should create his or her own URI or if it is really up to you to create one.
  • But create URIs for the objects in your repository.
• Create links wherever possible.
DSpace 5

- DSpace is the most often used software for Open Access Repositories worldwide
- Release of DSpace version 5.0 planned for December 2014 (release candidates are out, testathon is running)
- Will contain support for Linked Data (RDF/XML, Turtle, N-Triples, SPARQL)
- Will support content negotiation
- Highly configurable, good default configuration included

- Test it yourself:
  http://demo.dspace.org/data/handle/10673/5/ttl
  http://demo.dspace.org/data/handle/10673/5/ttl?text
  wget -O - --header='Accept: text/turtle' http://demo.dspace.org/jspui/handle/10673/5
  or download and install a release candidate

If you’re about to use DSpace 5.0 or above please consider switching Linked Data Support on.
Weaving repository contents into the Semantic Web | Pascal-Nicolas Becker | SWIB14 | Bonn, December 3, 2014

Slide 16

Technische Universität Berlin
Universitätsbibliothek
Pascal-Nicolas Becker
p.becker@tu-berlin.de

Servicezentrum Forschungsdaten und –publikationen
http://www.szf.tu-berlin.de

Repository DepositOnce
http://depositonce.tu-berlin.de

Thesis „Repositorien und das Semantic Web“ (in German)
http://www.pnjb.de/uni/diplomarbeit/