View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Purdue E-Pubs

Purdue University
Purdue e-Pubs

Libraries Faculty and Staff Scholarship and Research

Purdue Libraries

6-12-2012

Refactoring HUBzero for Linked Data

Michael Witt
Purdue University, mwitt@purdue.edu

Yongyang Yu
Purdue University, yu163@purdue.edu

Follow this and additional works at: http://docs.lib.purdue.edu/lib_fsdocs

Part of the Library and Information Science Commons

Recommended Citation

Witt, M. & Yu, Y. (2012). Refactoring HUBzero for Linked Data. Proceedings of the 12th ACM/IEEE Joint Conference on Digital Libraries, Washington, DC. http://dx.doi.org/10.1145/2232817.2232845

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

Refactoring HUBzero for Linked Data

Michael Witt
Purdue University Libraries
504 W. State Street
West Lafayette, IN 47907 USA
765-494-8703
mwitt@purdue.edu

Yongyang Yu
Dept. of Computer Science, Purdue University
305 N. University Street
West Lafayette, IN 47907 USA
765-494-6010
yu163@cs.purdue.edu

ABSTRACT

The HUBzero cyberinfrastructure provides a virtual research environment that includes a set of tools for web-based, scientific collaboration and a platform for publishing and using resources such as executable software, source code, images, learning modules, videos, documents, and datasets. Released as open source software in 2010, HUBzero has been implemented on a typical LAMP stack (Linux, Apache, MySQL, and PHP) and utilizes the Joomla! content management system. This paper describes the subsequent refactoring of HUBzero to produce and expose Linked Data from its backend, relational database, altering the external expression of the data without changing its internal structure. The Open Archives Initiative Object Reuse and Exchange (OAI-ORE) specification is applied to model the basic structural semantics of HUBzero resources as Nested Aggregations, and data and metadata are mapped to vocabularies such as Dublin Core and published within the web representations of the resources using RDFa. Resource Maps can be harvested using an RDF crawler or an Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) data provider that were bundled for demonstration purposes. A visualization was produced to browse and navigate the relations among data and metadata from an example hub.

Categories and Subject Descriptors

H.3.7 [**Information Storage and Retrieval**]: Digital Libraries

General Terms

Experimentation, Standardization

Keywords

Linked Data, Object Reuse and Exchange (OAI-ORE), Open Archives Initiative, Resource Description Framework (RDF), Virtual Research Environments, HUBzero

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

JCDL'12, June 10–14, 2012, Washington, DC, USA.
Copyright 2012 ACM 978-1-4503-1154-0/12/06 ...$10.00.

1. BACKGROUND

The establishment of the Network for Computational Nanotechnology in 2002 by the National Science Foundation led to the development of nanoHUB.org, a unique cyberinfrastructure platform for developing and sharing nanoscience resources online, and in particular, an integrated development environment for creating and executing simulations within a web browser and that utilize backend grid resources. By 2007, nanoHUB.org was serving approximately 1,000 resources to over 56,000 users from 172 different countries [1]. Based on this success, the NSF supported the generalization of the platform so that it could be adapted and adopted by other scientific communities [2]. The result, HUBzero[1], was released as open source software in 2010 and is maintained by a non-profit consortium led by Purdue, Clemson, and Indiana Universities with the University of Wisconsin. Over twenty-five "hubs" have been deployed in a wide variety of communities such as pharmaceutical product development, energy, social change, high performance computing, earthquake engineering, ethics, and science, technology, engineering, and mathematics (STEM) education.

HUBzero functions much like an institutional repository system in which users can create and upload resources, which are organized by resource type. By default, HUBzero supports courses, seminars, tools, downloads, series, workshops, publications, and teaching materials. Descriptive metadata inputted by the submitter of the resource enables it to be searched and browsed in the system. HUBzero allows users to tag, rank, discuss, and annotate resources as well as integrate them into various social networking sites. Each type of resource offers its own manner of presentation (e.g., tools prompt for input and can be executed online, seminars can be streamed as video or audio, etc.) Usage statistics and citations are reported back to those who submit resources.

HUBzero utilizes the Joomla content management system, which provides a framework for managing users, content, and web-based functionality. Hub resources are instantiated using the Resource component, which is a Joomla extension. Joomla utilize a model-view-controller (MVC) architecture that separates the underlying data from the presentation and interaction of the component and its user. In practical terms, this allows multiple views and interactions with the same data model by modifying different PHP scripts. Data relating to hub resources are populated by SQL queries into variables that are parsed and presented differently depending on the context of the interaction.

[1] http://hubzero.org

2. DATA MODELING AND MAPPING

2.1 Linked Data

While the hub presents a robust interface and rich functionality for users, its resources are not presented in a way that user-agents (i.e., a user that is a machine) can understand and use. With this motivation, Linked Data was selected as the approach because of its simplicity and high rate of adoption in the digital library domain relative to other Semantic Web approaches. The four basic "rules" of Linked Data are to use Uniform Resource Identifiers (URI) to identify things; use HyperText Transfer Protocol (HTTP) URIs that can be linked and followed; provide useful metadata when URIs are dereferenced, such as Resource Description Framework (RDF); and link to other URIs to create relations to other information [3]. The goal of this work is to enable hubs to become a part of this "web of data" and to serve as a practical introduction of HUBzero and its communities to the Open Archives Initiative.

2.2 Object Reuse and Exchange (OAI-ORE)

The OAI-ORE specification defines a set of standards for describing and exchanging aggregations of web resources. In simple terms, OAI-ORE defines an Aggregation as a collection of web resources (Aggregated Resources), and each Aggregation includes a Resource Map that lists the contents of the Aggregation and additional metadata such as the relationship between an Aggregation and something else on the web. An Aggregation can include other Aggregations, resulting in a Nested Aggregation [4]. Applying this to HUBzero, its basic structural semantics can be captured by describing the entire hub as an Aggregation with hub resources grouped by type (e.g., seminars, tools, publications) as Nested Aggregations. Lastly, each individual hub resource is defined as its own Aggregation, nested within its respective Aggregation by resource type containing its own metadata and Aggregated Resources.

2.3 Dublin Core

HUBzero requires users who are submitting a resource to identify the type of resource and to describe it using a local, custom schema. Dublin Core[2] provides a convenient and generic descriptive vocabulary that was straight-forward to map. This information is stored natively in HUBzero's MySQL database in relational tables.

RDF triples are constituted using the URI of the hub resource Aggregation (using hash notation) as the subject, the URI to the Dublin Core term as the predicate, and a string literal as the object, which is parsed from variables returning data from Joomla from the database and environment. The example hub allows submitters to apply a Creative Commons license to their work; in these cases, the URI to the license is used for the object to `dc:rights`. The `dc:publisher` is assigned statically.

3. TOOLS

3.1 OAI-PMH Data Provider

The Open Archives Initiative Protocol for Metadata Harvesting defines a web service for data providers to expose metadata records serialized in eXtensible Markup Language (XML) for retrieval by service providers (i.e., harvesters) using a simple set of six verbs [Identify, ListSets, ListMetadataFormats, List Identifiers, ListRecords, and GetRecord], which are called as HTTP requests [5]. The OAI-PMH data provider created for HUBzero was written in PHP and implemented as a stand-alone component in Joomla. Each resource type in the hub is exposed as an OAI-PMH Set, and each resource in the hub minimally furnishes a descriptive metadata record in Dublin Core. The data provider also advertises OAI-ORE Resource Maps using the ListMetadataFormats verb and will furnish the Resource Maps of Aggregations that represent individual hub resources when the service provider specifies a metadataPrefix of "oai_ore". The Resource Map for a hub resource references the Set (`ore:isAggregatedBy`) to which it belongs as well as the data and metadata that it aggregates (`ore:aggregates`). Resource Maps are serialized as RDF/XML. The OAI-PMH data provider implements one of the batch discovery methods suggested by the OAI-ORE specification [6]. Because it was designed as a stand-alone component in the Joomla framework, the data provider could be easily modified to expose metadata from other content managed by Joomla outside or independent of HUBzero.

Table 1: Dublin Core Mapping to HUBzero

Dublin Core	HUBzero table
dc:title	jos_resources.title
dc:creator	jos_resources.created_by
dc:subject	jos_tags.raw_tag
dc:date	jos_resources.created
dc:identifier	jos_resources.id
dc:description	jos_resources.introtext
dc:type	jos_resources.type
dc:publisher	(*statically assigned*)
dc:rights	jos_resources.params

3.2 RDF Crawler

To demonstrate discovery and use of Linked Data from the web, a crawler was written in Python to traverse all the resource web pages in an example hub. The crawler begins from the top-level of the hub (the beginning splash page, which is the Aggregation that represents the entire hub) and dereferences URIs to the Aggregations and Resource Maps for resource types and individual resources. The task of the crawler is to parse the HTML source and extract div fragments that have a class property of "`ResourceMap`" or "`Aggregation`". The BeautifulSoup[3] parser was used to make the crawler resilient to ill-formed HTML fragments such as the incorrect embedding of `h2` tags within `span` tags.

Resources Maps are serialized in the RDFa+XHTML format such that metadata are embedded within div fragments in the HTML source. The `class` property of the div fragment indicates whether the fragment describes the Resource Map or the Aggregation based on its respective string value. The Dublin Core metadata are wrapped within `span` tags with `property="dc:title"`, etc. String literals are embedded in span tags, whereas URIs are embedded in `a` tags. The job of the crawler is to parse all the contents embedded

[2] `http://dublincore.org`

[3] `http://www.crummy.com/software/BeautifulSoup/`

in these tags and generate the triples with proper subjects, predicates, and objects.

The crawler traverses URIs to all the RDF triples and stores them in an N-Triples file. N-Triples is a line-based, plain text serialization format for RDF graphs. Each line of the file represents a single statement of information or a comment. Each statement consists of three parts, separated by whitespace – the subject, the predicate and the object – and is terminated with a full stop. Subjects take the form of URIs, which are delimited by angle brackets. Objects may be URIs or string literals, which are represented by a C-style string. The crawler builds a resource URI list by crawling and parsing URIs within each resource type of the hub. Whenever the crawler parses an a tag of the HTML with `rel` property of value of `ore:aggregates`, it will insert the URI contained in the tag in the list. By iterating all the resource types in the prototype, all the URIs of resources are inserted into the list. For each single URI in the list, the crawler parses the `div` fragments in the HTML source file and stores the RDF triples in the result file.

After all the triples have been harvested from the example hub, the N-Triples file is flushed to a TDB triplestore for permanent storage and retrieval within a Jena framework[4]. Jena implements APIs for dealing with Semantic Web building blocks such as RDF. TDB is a persistent graph storage layer for Jena. TDB works with the Jena SPARQL query engine to provide a SPARQL endpoint along with a number of extensions (e.g., property functions, aggregates, arbitrary length property paths).

3.3 Graph Browser

With the triples populated from the hub and stored in the native triplestore, a graph browser was developed to help us analyze the data quality and validate the metadata exposed as Linked Data (see Figure 1). JUNG[5], a Java based graph framework, was used to achieve the goal of visualization. JUNG is a software library that provides a common and extensible language for modeling, analysis, and visualization of data that can be represented as a graph or network. Our implementation of the graph browser has an interactive interface that enables one to click on individual nodes and navigates through the hub data and metadata.

The graph browser will first send a SPARQL query `"SELECT ?s ?p ?o WHERE ?s ?p ?o;"` to retrieve all the triples in the TDB triplestore. The graph browser will display the hub resources in different layers as the user clicks on various nodes in the browser. A circle node in the graph represents a URI or a plain string. A directed arrow with a URI above it denotes the predicates between the subject and the object. The circle node which the arrow points out is the subject of the triple and the circle node which the arrow points to is the object of the triple. There are three different colors of the circle nodes in all, i.e., blue, green, and grey. The blue circle nodes represent URIs of Aggregated Resources and Resource Maps. The green circle nodes represent plain strings appearing in the object field of the triple. The grey circle nodes are pointed to by dotted arrows from green circle nodes. These grey circle nodes indicate shared boundaries that connect to a larger graph outside of the immediate current scope.

The first page of the graph browser shows the aggregation of all the triples in the hub in a graphical form. One can click on any blue circle node to have a clear look at the details of this particular Aggregated Resource or Resource Map. If the blue node that one clicks on is a Resource Map, the browser will display the descriptive information of the Resource Map and the URI of the Aggregated Resource it describes. If the blue node that one clicks on is an Aggregated Resource, the browser will graph its Resource Map, all the Dublin Core metadata records about this Aggregated Resource, any resource it aggregates, and any Aggregation it belongs to. If a green circle node is clicked on, the browser will display all the triples with this string value in the object field.

With the help of the graph browser, we can debug the correctness of exposing the Dublin Core metadata records from the hub and validate the Resource Maps. This graph browser serves as a supplementary tool to navigate through RDF triples by introducing an interactive way between users and the data. It has also been a helpful tool for demonstrating to people the network power of Linked Data and to visualize the linkages that are being created between their data and metadata. Numerous classes and demonstrations of the visualization have been given in the campus Fakespace FLEX system Virtual Reality Theater.

4. FUTURE WORK

The combination of OAI-ORE and Dublin Core enables a base representation of HUBzero and its content.; however, richer and more specific vocabularies should be identified and incorporated. It would be useful to create more links to other, related information on the web, for example, to link to scientific workflows (e.g., MyExperiment) that are supported by hub tools and datasets. Another important, next step is to incorporate vocabularies on the front-end data ingest to produce more URIs and fewer string literals as objects in the RDF. For example, tags could be constructed as Library Congress Subject Headings or another linkable subject classification scheme.

Inspired by Tarrant et al.[7], a synchronization tool to replicate content and semantics from HUBzero to other repositories using OAI-ORE has been developed and described in a different paper that is currently under review.

Two new tools are under development now: one to enable users to create and publish their own collections (Aggregations) on the hub, and another to leverage Linked Data for semantic search.

The Linked Data implementation for HUBzero along with the OAI-PMH Joomla component, RDF Crawler, and Graph Browser are in the process of being incorporated into the HUBzero open source software distribution and should be available in the next major version release of HUBzero. An assessment is planned for 2013.

5. ACKNOWLEDGMENTS

This work was sponsored by a National Leadership Grant (LG- 05-10-0117-10) from the Institute of Museum and Library Services in collaboration with Clemson University and the National Parks Service. Contributions to software development were also made by Venkata Surya Srik Gandikota, Aswathy Siviram, Siddharth Kumar Singh, and Naveen Somasundaram.

6. REFERENCES

[4] http://incubator.apache.org/jena/documentation/tdb/index.html

[5] http://jung.sourceforge.net/

[1] G. Klimeck, M. McLennan, S.P. Brophy, G.B. Adams, and M.S. Lundstrom. Nanohub.org: advancing education and research in nanotechnology, *Computing in Science & Engineering* 10(5): 17–23, October 2008.

[2] M. McLennan, and R. Kennell. HUBzero: a platform for dissemination and collaboration in computational science and engineering, *Computing in Science & Engineering* 12(2):48–53, March 2010.

[3] C. Bizer, T. Heath, and T. Berners-Lee. Linked-Data - the story so far, *International Journal on Semantic Web and Information Systems* 5(3):1–22, 2009.

[4] C. Lagoze, H. Van de Sompel, M. Nelson, S. Warner, R. Sanderson, and P. Johnston. A web-based resource model for scholarship 2.0: Object Reuse & Exchange, *Concurrency and Computation: Practice and Experience*, June 2010.

[5] H. Van de Sompel, M.L. Nelson, C. Lagoze, and S. Warner. Resource harvesting within the OAI-PMH framework, *D-lib magazine* 10(12), December 2004.

[6] Open Archives Initiative Executive and Technical Committees. ORE User Guide - Resource Map Discovery, `http://www.openarchives.org/ore/1.0/discovery.html`, 2008.

[7] D. Tarrant, B. O'Steen, T. Brody, S. Hitchcock, N. Jefferies, and L. Carr. Using OAI-ORE to transform digital repositories into interoperable storage and services applications, `http://journal.code4lib.org/articles/1062`, March 2009.

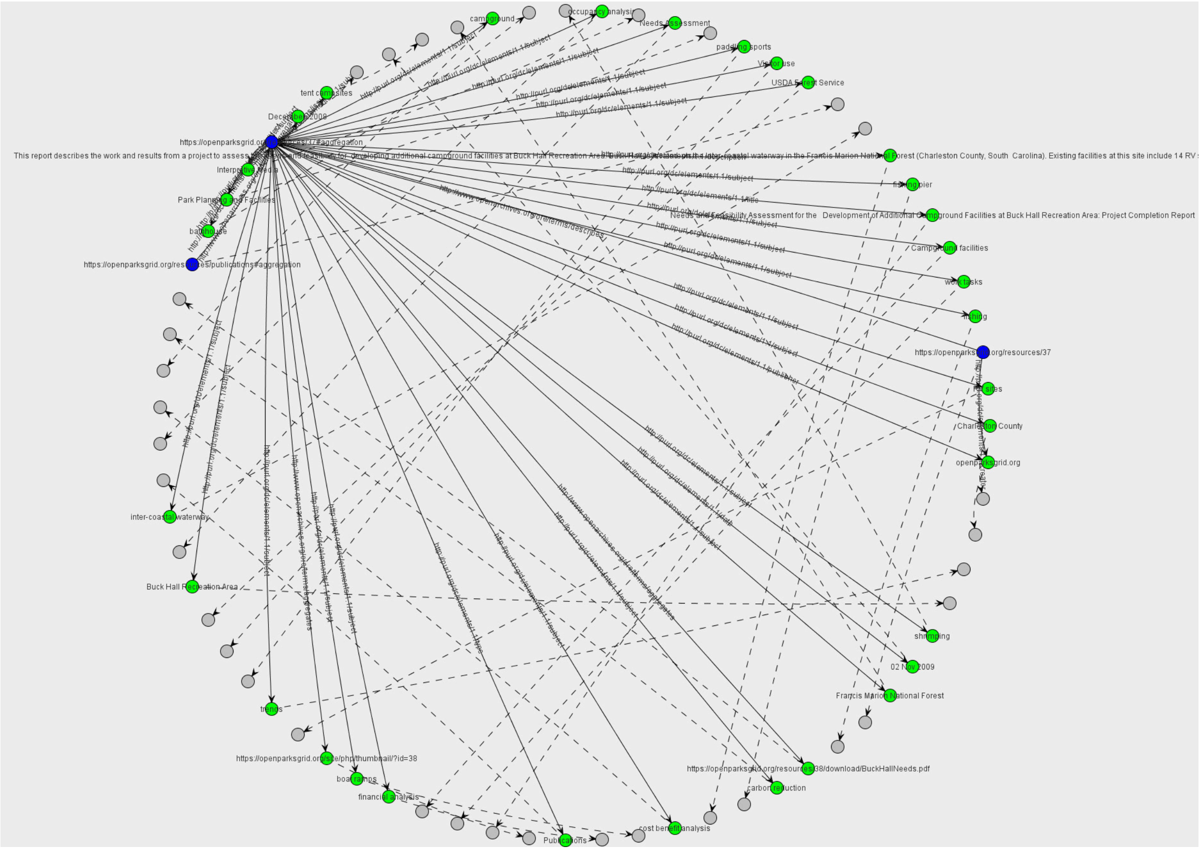
campground
occupancy analysis
Needs Assessment
paddling sports
Visitor use
USDA Forest Service
tent campsites
December 2008
https://openparksgrid.org/resources/37#aggregation
This report describes the work and results from a project to assess the need and feasibility for developing additional campground facilities at Buck Hall Recreation Area. Buck Hall is located on the inter-coastal waterway in the Francis Marion National Forest (Charleston County, South Carolina). Existing facilities at this site include 14 RV s
Interpretive Media
Park Planning and Facilities
bathhouse
https://openparksgrid.org/resources/publications#aggregation
fishing pier
Needs and Feasibility Assessment for the Development of Additional Campground Facilities at Buck Hall Recreation Area: Project Completion Report
Campground facilities
work tasks
fishing
https://openparksgrid.org/resources/37
RV sites
Charleston County
openparksgrid.org
inter-coastal waterway
Buck Hall Recreation Area
shrimping
02 Nov 2009
Francis Marion National Forest
trends
https://openparksgrid.org/site/php/thumbnail/?id=38
boat ramps
financial analysis
https://openparksgrid.org/resources/38/download/BuckHallNeeds.pdf
carbon reduction
Publications
cost benefit analysis
http://purl.org/dc/elements/1.1/subject
http://purl.org/dc/elements/1.1/title
http://purl.org/dc/elements/1.1/description
http://purl.org/dc/elements/1.1/publisher
http://purl.org/dc/elements/1.1/date
http://purl.org/dc/elements/1.1/type
http://purl.org/dc/elements/1.1/creator
http://www.openarchives.org/ore/terms/describes
http://www.openarchives.org/ore/terms/aggregates

Figure 1: Connected Graph of Hub Resource Aggregation