TripFS: Exposing File Systems as Linked Data

Bernhard Schandl, bernhard.schandl@univie.ac.at
University of Vienna, Department of Distributed and Multimedia Systems

Abstract: File systems are highly interesting sources of information since large amounts of digital information are stored using plain file hierarchies. However the question of how file system data can be integrated into the Web of Data has not yet bet been sufficiently addressed. In this paper we give a short overview on *TripFS*, a Java-based server software that extracts RDF descriptions from a file system, links file resources to other relevant data sources, and exposes these data sets according to Linked Open Data principles.

In many application contexts, hierarchical file systems are an important means of data storage for unstructured or heavily heterogeneous content. Files are used in a wide variety of applications, ranging from personal data on typical desktops, over shared folders in an enterprise, to data stored on a web server and published world wide. As these examples show, files are highly generic and universally usable.

Files are typically accessed either by locating them via their absolute file path (usually by traversing through a directory hierarchy), or using a full-text search engine that keeps an index over file contents. Both approaches are well supported by common operating systems. However there exist no platform-independent mechanisms that allow file system contents to be integrated into a larger information network: files cannot be linked to other information objects, and their metadata descriptions cannot be processed in a platform- and operating system-independent manner.

Because of these deficiencies, it is obvious to apply Linked Open Data principles to file systems. *TripFS*, presented in this paper, is a tool that represents directories and files as RDF resources, extracts metadata and creates links to other data sets, serves these metadata as RDF or HTML using content negotiation, and provides a SPARQL endpoint that allows clients to execute queries over the entire file system. TripFS is entirely implemented in Java and consists of six main components:

Store. TripFS is implemented using the Jena Semantic Web Framework and abstracts over a concrete RDF storage. It has been successfully tested with an in-memory storage as well as on top of a PostgreSQL relational database.

Crawler. On startup, TripFS crawls the file system starting from a given root directory. All files and directories are scheduled for subsequent extraction and interlinking, Each file is assigned a globally unique, dereferenceable HTTP URI, which is independent from the file path and remains intact when the file is modified or moved.

Watcher. TripFS tracks changes in the file system (creation, deletion, and modification of files); after a change, the affected files are re-scheduled for metadata extraction and interlinking. Hence the LOD representation is always in sync with the actual file system.

Extractors. TripFS provides a modular framework for metadata extraction. It re-uses components from the Aperture framework[1], which provides extractors for a number of popular file formats (including Office documents and multimedia data like images and audio files). The framework extracts data depending on the file format (e.g., title and artist from music files, width and height from images, a.s.f.) and stores them in the RDF store. Extracted data mostly adheres to the NEPOMUK Semantic Desktop ontologies[2].

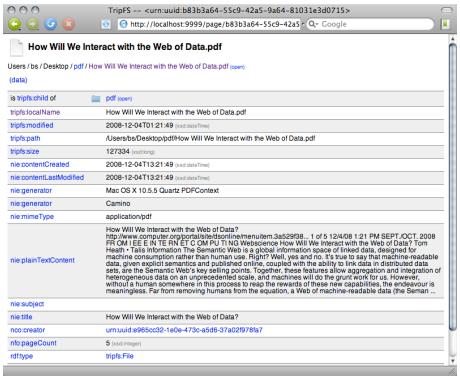

Figure 1: TripFS HTML Rendition

Linkers. Similar to extraction, a pluggable set of linkers can be instantiated within the TripFS server. After crawling and extraction, files are scheduled for linking. Currently we have implemented two experimental linker components, one that links music files (e.g., in the MP3 format) to Musicbrainz[3] using artist name, track name, and duration, and one that links paginated documents (e.g., PDF or MS Word files) to ACM publication records[4] based on the document title. These linkers generate `owl:sameAs` and `rdfs:seeAlso` links; a detailed analysis on the linking quality is subject to further research. Additional linkers and extractors can be easily added to the TripFS system by implementing corresponding interfaces.

Web Server. RDF descriptions about files are served according to Linked Open Data principles as RDF (RDF/XML, Turtle, N3) and HTML (see Fig. 1), which can be accessed by the client using content negotiation. The HTML rendition is enriched with embedded RDFa descriptions and file system-specific graphic elements (e.g., file icons). Additionally the server provides a standards-compliant SPARQL endpoint.

[1] `http://aperture.sourceforge.net`
[2] `http://www.semanticdesktop.org/ontologies`
[3] `http://wiki.musicbrainz.org/RDF`
[4] `http://acm.rkbexplorer.com`