The International Journal of Digital Curation
Issue 1, Volume 6 | 2011

Where the Semantic Web and Web 2.0 Meet Format Risk Management: P2 Registry

David Tarrant, Steve Hitchcock and Les Carr,
School of Electronics and Computer Science
University of Southampton

Abstract

The Web is increasingly becoming a platform for linked data. This means making connections and adding value to data on the Web. As more data becomes openly available and more people are able to use the data, it becomes more powerful. An example is file format registries and the evaluation of format risks. Here the requirement for information is now greater than the effort that any single institution can put into gathering and collating this information. Recognising that more is better, the creators of PRONOM, JHOVE, GDFR and others are joining to lead a new initiative: the Unified Digital Format Registry. Ahead of this effort, a new RDF-based framework for structuring and facilitating file format data from multiple sources, including PRONOM, has demonstrated it is able to produce more links, and thus provide more answers to digital preservation questions - about format risks, applications, viewers and transformations - than the native data alone. This paper will describe this registry, P2, and its services, show how it can be used, and provide examples where it delivers more answers than the contributing resources. The P2 Registry is a reference platform to allow and encourage publication of preservation data, and also an examplar of what can be achieved if more data is published openly online as simple machine-readable documents. This approach calls for the active participation of the digital preservation community to contribute data by simply publishing it openly on the Web as linked data.[1]

[1] This paper is based on the paper given by the authors at the 6th International Conference on Preservation of Digital Objects (iPres 2009), October 2009; received January 2010, published March 2011.

The *International Journal of Digital Curation* is an international journal committed to scholarly excellence and dedicated to the advancement of digital curation across a wide range of sectors. ISSN: 1746-8256 The IJDC is published by UKOLN at the University of Bath and is a publication of the Digital Curation Centre.

Introduction

The World Wide Web is recognised as the fastest growing publication medium of all time, now containing well over 1 trillion URLs (Alpert & Hajaj, 2008) and still growing exponentially according to figures taken from reports by Google, Yahoo and Netcraft. As a result, we face problems in both finding and being able to use all the available data. In this paper we focus on maximising the value of data published on the web, specifically in the area of digital preservation and file format registries. The core outcome of this work is to demonstrate how some emerging web publishing techniques can lead to the ability to construct a set of powerful and flexible services focused on digital preservation.

Publishing of data is one of the core features of the Web 2.0 (O'Reilly, 2007) and Semantic Web (Berners-Lee, Hendler & Lassila, 2001) initiatives, and both have shown that users are willing to share information and collaborate on the Web on a large scale. This can be seen with the success of Wikipedia, and the take up of blogging and social networking services designed to build links between people on the web. These are just a few examples that are now publishing information in machine-readable formats such as RSS or ATOM that can be customised and displayed in ways to suit the consumer of the information.

To process data automatically for structured consumption, machine readability is only the first step. The next step is machine understanding, where not only is the data split into concepts, but these concepts are understood and aligned with other concepts. This is at the core of the Semantic Web. By using techniques from the Semantic Web, this paper demonstrates the simplicity of aligning data available on the web from services such as PRONOM, a file format registry produced by the National Archives in the UK (Brown, 2005) and DBpedia, a linked data version of Wikipedia, such that seemingly complicated searches across these services can now be performed with a single request.

Being able to query data from a disparate set of services requires some form of caching of the data available at those services. The P2 Registry essentially provides this cache by storing data in a model-free, unstructured database on top of which many services are built to manipulate the data. The registry automatically harvests information from various defined data sources that are published in an open and machine-readable fashion. Currently this service is specifically directed towards the file formats of the materials collected in digital repositories such as institutional repositories.

Information in the registry is made available through a set of user and programming interfaces (APIs) that are designed to present information on resolving format risk analysis. By providing both high-level summary interfaces, where the searches are hidden from the user, as well as the search interface itself, ensures that the end user has the greatest level of flexibility when it comes to using the data known by the registry. This paper presents both interfaces and give examples of how the high-level interfaces are constructed from a few simple queries through the API.

This paper is structured to outline the entire process, from good publication techniques on the web, to the construction of the high-level services for the P2 Registry. The first sections look at the background to linked data and the Semantic Web, focussing on the importance of following four simple rules for publishing on the web. Then we look at how techniques from the Semantic Web can be used to provide understanding of linked data and the use of ontologies. We then briefly look at existing technologies which are designed to aid digital preservation, including registries of data pertaining to file formats and related tools. In this section we also look at a few of the services built on top of these registries with the aim to show later how the P2 Registry can compliment these. The main body of this paper outlines the P2 Registry and its interfaces for importing, processing and presenting data. We look at how the P2 Registry is able to directly import and cache linked data in the form of RDF, how this is aligned using a series of simple ontologies and how it is queried using simple searches. Finally, the wider uses of the P2 Registry, its implications for digital preservation and possible further development are considered.

This work began in the JISC Preserv 2 project[2] as a response to the perceived limitations of the available tools for file format analysis, and continues in the JISC KeepIt project[3]. Both projects are concerned with managing and preserving the contents of institutional and digital repositories, with the former focussed on the development of preservation tools and services, while the latter is working with repository managers to apply these tools to exemplar preservation repositories.

The most important aspect of this paper is to emphasise the power of a community and the sheer volume of data it can publish cooperatively. The P2 Registry brings this data together so that it can be used to answer questions on digital preservation.

Linked Data

Organisation of data on the web has proved to be a real problem over the years. As people start to realise the importance of linkable data, however, we are starting to see better use of one the web's simplest technologies, the Uniform Resource Locator (URL). A URL represents the location of "something" on the web. Aligning the principles of the URL with that of giving everything on the web a URI (Uniform Resource Identifier) empowers users to be able to link directly to very specific parts of the web, those which now provide data about "things".

Consider a simple real world example, such as booking a holiday over the internet. You and a travel partner are online, chatting to each other over email and browsing a travel website. You find a nice hotel and then copy and paste the browser link in an email to show your partner. The problem is that the website uses session- and path-based browsing, which means that your partner opens the link to be greeted by a page containing a session error. This could be avoided if the hotel page had a URI that could be referenced independently regardless of the path taken to find it.

[2] http://preserv.eprints.org. Retrieved February 11, 2011.
[3] http://preservation.eprints.org/keepit/. Retrieved February 11, 2011.

Essentially this is the goal of Web 2.0, to put discoverable data online. This means that data should remain online in a static location, be well annotated and also link to other resources. Establishing static URIs for resources is the first of four rules for publishing on the linked data web (Berners-Lee, 2006):

1. Use URIs as names for things;
2. Use HTTP (web) URIs - thus they are also URLs;
3. Provide useful information in useful formats, e.g. RDF;
4. Include links to other URIs.

Rules 2-4 emphasise that if someone goes to your URI on the web then it would not only exist but also tell you something about itself and link to other related items. The way this differs from many current web publishing techniques, however, is the publication of data, either alongside or instead of human-readable web pages.

In a series of tutorials, Heath explains how to publish linked data on the web (Heath, 2009; Bizer et al., 2008). In these he also looks at the many ways to serialise and provide data about "things". Once again the key is the use of URIs and URLs for identification and location. Figure 1 outlines the basic principles behind data publishing on the web, where we start with a URI representing a "thing". From this URI many serialisations can be accessed which expose the same data in different forms. For instance a plain HTML page would be the default location displayed by web browsers. Alternative (HTTP code 303) versions (or serialisations) of the same data such as XML, RSS or RDF versions might also be offered.

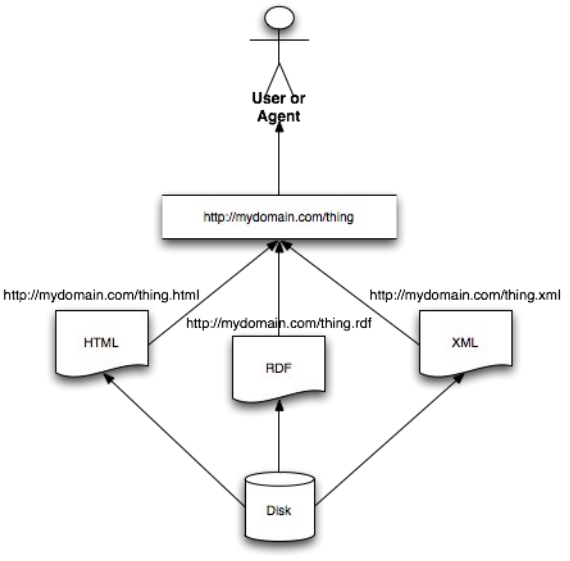

Figure 1. URIs and URLs, Alternative Serialisations.

If we look at linked data from the view of digital preservation, specifically file formats, each format would be represented by a URI. From this URI you would be able to view information about the format, get alternative versions of the information, such as in XML, and, most importantly, be able to follow links to other similar file types or types with similar properties. To a certain extent PRONOM and DBpedia provide this functionality. For this reason these two services were used as a starting point for the P2 Registry.

The Semantic Web

While Web 2.0 has focused on getting readable data from the web in a linked data fashion, the Semantic Web realises that, even at this point, there is still a problem with data deluge. The Semantic Web introduces the requirement for data publishers to add context and encourages the creation of formal descriptions for concepts, terms, and relationships within each knowledge domain. Systems could thus be envisaged which have understanding of real world concepts, as outlined in Berners-Lee's original Semantic Web paper (Berners-Lee et al., 2001). Much like the structure used in a relational database model, the Semantic Web encourages publication of a glossary or terminology with the data such that the "model" can be understood and thus kept constant.

"Model" is a term that occurs often in this as well as many other fields of research. A model is a specification for data publishing such that everyone understands all the terms used. As a simple example, take the term Title. This could mean a name prefix (e.g., Mr, Mrs, Dr), or it could mean the title of the resource (e.g., the name of a publication). Without a well-defined model which explains the usage then there is no way of telling. To differentiate between usages of the same term, XML and RDF introduce the idea of the namespace prefix to a term. A good example here is the Dublin Core specification (dc) which provides "dc:title", which is defined as: "A name given to the resource."

To keep things simple, the Semantic Web encourages information to be published as simple "triples", where two items are related via a third element that describes the relation (Manola, Miller & McBride, 2004). In this space, the triple is constructed from three URIs representing the subject, predicate and object, respectively – linking this to our model example, if the subject was the publication, a possible predicate would be *dc:title* and the object would be the text string representing the title.

At the core of the Semantic Web effort is the Resource Description Framework (RDF): a markup language that extends XML. Through these extensions it makes namespaces mandatory for both resources/objects as well as for the predicates that link them. Figure 2 shows a simple set of triples describing some characteristics of a file format showing the namespacing with a ":" separator. In this representation ovals represent URIs, squares represent plain text nodes and predicates are represented by the arrows which join the nodes.

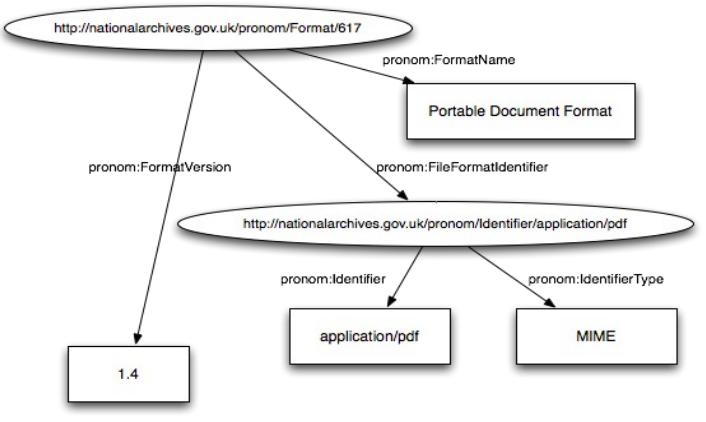

Figure 2. RDF Graph Relating to File Format Data.

While the Semantic Web does not define a limited set of predicates and relations to use, there are a set of well established ontologies (glossaries of terms) – we have already mentioned Dublin Core – that allow mappings between data as well as define the concepts themselves. For the purposes of this work we focus on some of the key terms provided by the RDF, RDFS (RDF-Schema) and OWL (Web Ontology Language) namespaces:

- rdf:type - The subject is an instance of a class (URI linker);
- rdfs:label & rdfs:comment - Human readable fields (Text);
- rdfs:subClassOf - The subject is a subclass of another class (URI linker);
- rdfs:domain & rdfs:range - The domain and range of values for this subject. (URI linker);
- owl:sameAs - The subject URI can be considered to represent the same as object URI.

The advantage of using such glossaries is there are many applications developed to use semantically annotated data in RDF which can understand these concepts. These caching stores are essentially databases that are not constrained by any data model but still provide the query interface. The model builds itself as data is added or imported into the database, and because the underlying store understands terms such as owl:sameAs at the lowest level then queries are able to return implicit results.

The disadvantage of not having a set number of vocabularies from which predicates can be drawn is that it is inevitable people will end up inventing a new predicate which has exactly the same meaning as an existing one. This means that for the data to be used an alignment process has to take place. This kind of process is commonplace already on the web as people import data into their own model. However, this loses the concepts established by the original model as well as any provenance information. The Semantic Web encourages the alignment of concepts and terms by using the owl:sameAs property to simply state that one “thing” (including predicates) is **exactly** the same as another.

Since the original Semantic Web effort it has become clear to many, including Carroll et al (2005), that triples are simply not enough to express all the information required for provenance and trust. In order for an axiom (a fact expressed through the use of a triple) to be ratified, you require a further piece of information which details the context; thus you now have a quad. Take the triple “Dave worksFor UniversityOfSouthampton”. This is a fact which may only be true for a limited amount of time: if Dave moved jobs would he then work for two organisations according to our data? Introducing the fourth data item allows the data publishers to define the context in which the triple (or set of triples) is valid – it is the provenance of that triple. Figure 3 shows how this context object can be used not only to track the provence of a triple but also as a subject of a triple itself such that the context can even be defined. Figure 3 is a real life example about one of the authors of this publication. As all URIs are persistent you should be able to follow the example given here in a browser. For clarity the quad relation is represented by a bolder arrow which has no predicate assigned to it.

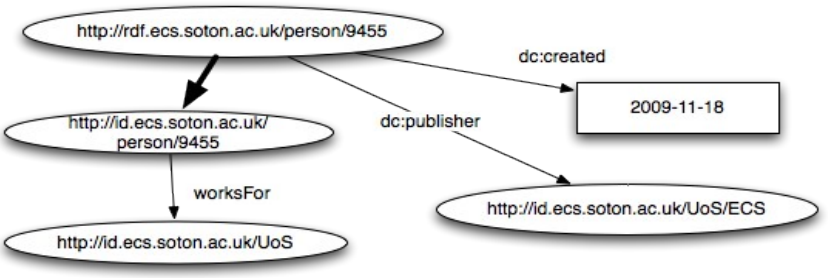

Figure 3. RDF Graph (as Quads).

With the realisation that the quad is required to enable full context-aware queries, software and protocols used to load and query RDF-based data have also moved on. Currently, the Talis platform (as used by data.gov.uk) and the 4store software (the new name for 3store) have the capability to store and provide access to data stored as quads. For the purposes of this work we use the 4store platform, which has been shown to be a highly scalable platform, designed to handle $15x10^9$ quads and allow query in "google time" (Harris et al., 2009).

To query the caching store we are using SPARQL (Simple Protocol and RDF Query Language), which is a World Wide Web Consortium (W3C) recommendation (Seaborne & Prud'hommeaux, 2008). SPARQL is much like SQL in syntax with extensions to support data in the triple-based format. SPARQL is bundled with the 4store product, which also provides a web-based API for accessing the services and performing queries remotely.

Digital Preservation

Digital preservation is becoming of greater concern as we see many resources born in digital form only. A major part of digital preservation lies in the keeping the file bitstreams intact either on disk, tape or even in the cloud (Tarrant, Brody & Carr, 2009). The other key aspect of digital preservation realises that even if the original bitstream or file is accessible in 20 years time, there is a risk that there will be no software able to read or accurately render that file type.

The first stage in active file preservation is file format identification, and this includes specific revisions and characteristics of the file. Several introductions to file formats and their selection for different situations have been put together by Abrams (2007), this report also goes some way to introducing the importance of significant properties. Significant properties of a file are a local list of the important characteristics of a file. For example, in a word document the track changes may be an extremely important piece of metadata which is lost when a PDF is made. Wilson (2007) gives an excellent introduction and background to this area, which, in the end, is simply another set of metadata relating to file formats.

From the analysis and background work we find that metadata about file types, their characteristics and properties is very important in digital preservation. In turn, this led to many projects, summarised by Knight (2007), being established to collect, store and use this information. One of the most widely known registries is PRONOM (Brown, 2005) which is focussing on resources which are collected from UK governmental departments.

With data available through registries, such as PRONOM, there are already a great many services utilising this data. PRONOM-ROAR is an extension to the Registry of Open Access Repositories (ROAR) which remotely scans and analyses file types in a repository in order to provide that repository with a Preserv profile (Brody et al., 2008). Moving forward, the next problem is risk analysis and migration, and while risk analysis is a young research area, migration services built on top of the metadata in registries are becoming more common (Ramalho et al., 2008).

The problem now is that there are too many gaps in the current registries where information pertaining to file formats is either not present or incomplete. The aim of the P2 Registry is to show that, by sourcing data from the wider community, we can fill some of these gaps and in turn encourage publication of more data to fill further gaps. In turn, services using this data then become much richer in their capabilities.

The P2 Registry

The P2 Registry[4] is essentially a Semantic Web system backed by a model-free, unstructured RDF registry upon which ontologies and profiles can be applied to manipulate the data. Thus the P2 Registry is using many of the technologies described in the previous sections. The key additions come in the form of the data harvester, which uses a set of import plugins to adjust data on import, if needed, and the high-level interfaces that make the whole system easier to use and more powerful for the digital preservation community.

The registry automatically harvests information from various defined information sources that are published in an open and machine-readable form. Currently, this service is specifically directed towards the file formats of the materials collected in digital repositories. Helpfully, DBpedia publishes data in RDF so no changes are needed to import this data into the P2 system. On the other hand, PRONOM data is only currently (at November 2009) available in XML, and has to be parsed through an import plug-in.

The purpose of the import plugins is to normalise (remove any imposed structure) and translate source data into a set of triples represented in RDF. In the case of data from PRONOM, the importer also constructs the glossary of terms used by PRONOM to represent relations between objects. With the aim of keeping the data loosely coupled, there were no restrictions applied on the glossary, as any conflicts - such as two terms meaning the same thing - could be aligned later using the *owl:sameAs* property. Doing this also demonstrates the total flexibility of the model-free caching store. Of course, it would be beneficial if when systems talk about “things”, e.g. authors, they use the same globally unique URI. Unfortunately, this doesn’t happen, which is why a manual alignment with the *owl:sameAs* predicate is required. This alignment has to be a manual process – if it could be done automatically we wouldn’t need the sameAs predicate.

Having imported the data into the registry from our sources via whichever means is most applicable, the next stage is to link the data. This step is only required because no links were found between the data on DBpedia and PRONOM. By using the set of predicates outlined earlier, one of the first relations to be established was the link

[4] P2-Registry is available at http://p2-registry.ecs.soton.ac.uk (November 2010).

between a specific file format version in PRONOM and its generic parent data imported from DBpedia. To link the data, a series of search interfaces have been developed to help the user to find or provide semantic relations linking two objects.

Figure 4 shows the simple use of subclasses of various PDF formats, which was added to all PDF variations, where applicable. Having done this, the number of tools found that could read a PDF format (1.4) jumped from 19 in the PRONOM registry to 70 in total in the P2 system.

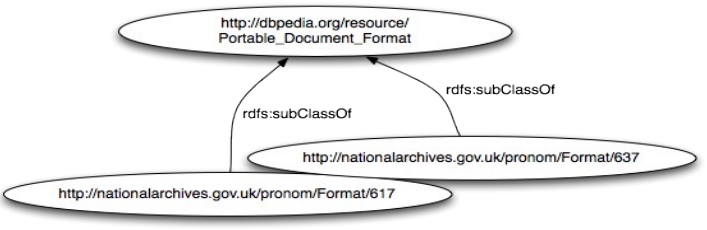

Figure 4. Simple Linking Example of Data Using RDFS.

Since the P2 Registry does not have a set data model, it can import any amount of data from many sources and allow the data to be queried directly. With the system operating using quads you can also easily see who or what was the source for each piece of information returned. Thus, trust models can begin to be applied. You can even query the P2 system and choose to exclude certain data sources.

Another key feature of the registry is the ability to import arbitrary ontologies that can be used both to infer new facts from existing information as well as to align (in the case where two concepts are similar or the same in nature) information already in the registry. For example, an ontology had to be added to the P2 Registry to obtain the result above where the number of tools found capable of reading PDF from a single query was greater than in the original registries.

Information imported from the original registries on software tools was specific about what the tools could do, e.g. open, save, create, render, print, etc. Performing a single query to find all tools requires the addition of further information to group these operations into one category or class. Figure 5 shows the part of the ontology constructed and added to the system to group the "SoftwareLink" class. Now it is possible to ask the registry for all software tools which have a "SoftwareLink" to the format in question. Due to all the subclasses created, PDF 1.4 will transparently include all information relevant to all PDF versions, and tools of all types will be returned by the query.

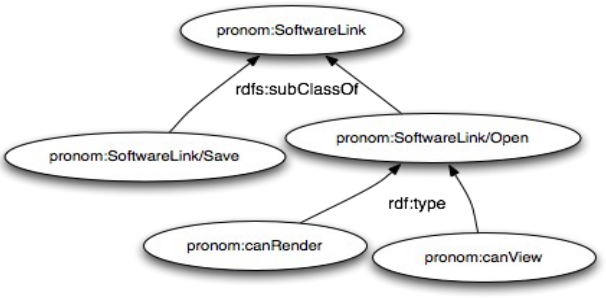

Figure 5. Software Category Subclassing in RDF.

Figure 6 shows the SPARQL query, passed to the registry through the query interface, that returns the list of software able to manipulate a specific format. As can be seen, a SPARQL query consists of sets of triples where variables are represented by anything preceded with a question mark. In this case we are looking for ?x, where ?x is the URI of the software and has a SoftwareName ?name. ?x is then related to our format (617) through a SoftwareLink predicate. We could add the quad identifier to this query, but it is not needed in this case because here we are looking for a complete set of answers from all our data sources.

```
select distinct ?name ?y where
{
        ?x ?y <http://nationalarchives.gov.uk/pronom/Format/617> .
        ?y rdf:type <http://nationalarchives.gov.uk/pronom/SoftwareLink> .
        ?x pronom:SoftwareName ?name
}
```

Figure 6. SPARQL query to find software compatible with PDF v1.4.

SPARQL provides the base level interface to the data contained in the registry. It is possible and necessary to construct a number of higher-level interfaces to allow easier manipulation and browsing of the data.

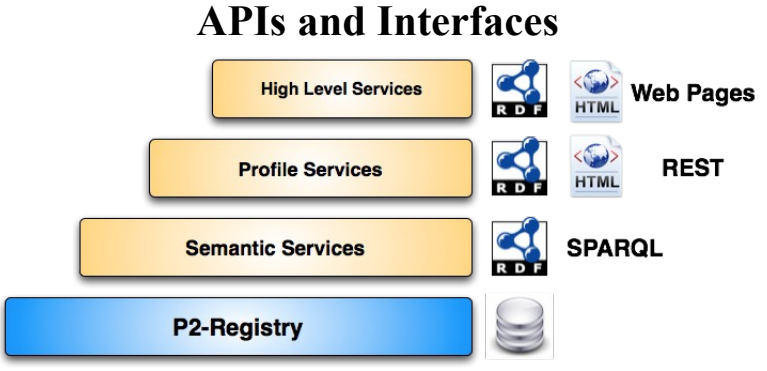

Figure 7. P2 Registry Design Layer Cake.

This section looks at the overall design and set of interfaces available at each layer in the P2 Registry. Figure 7 shows the layer cake which became the specification for the design of the registry. At the lowest level the P2 Registry is a caching database. On top of this sits a SPARQL query service with direct access to the RDF stored within the registry. This semantic layer is designed purely for use by other services and agents which can harvest the data and results of queries for their own use.

Above this layer sits a set of services that perform some form of manipulation on the data before use. This translation will either be serialisation to provide the same data in different formats, e.g. XML and HTML, or summation services that combine data to build new profiles of concepts and objects in the registry. Taking a lead from the linked data guidelines by Heath (2009) (Figure 1), the P2 Registry exposes URIs with related URLs to obtain the same data in different formats, such as HTML, XML and RDF. The profiling layer is also where RESTful services have been built to manage the registry and to import further data.

Finally, the high-level services are designed to hide the data while providing key information and interacting directly with physical users. Although some RDF can still be obtained at this level, these services are designed simply to demonstrate what can be done on top of the other services. High-level interfaces available include a data browser, which uses the hubs and authorities algorithm to rank "URIs"; a risk-profile analysis service, which uses a set of rules added to the registry to provide a risk score for a particular format; and a migration pathway interface that provides information on tools that can translate one format to another.

Searching

Even with the registry focussed on data specific to digital preservation, in particular file formats, there are still just under 44,000 statements in the current registry. Among these statements is data from PRONOM, plus any available data from DBpedia related to the PDF formats. Searching thus becomes a key activity in order to gain familiarity with the contents of the registry, as such there also needs to be a way to order search results putting the most relevant first.

The P2 Registry provides a simple triple-based search interface, where each URI has been ranked to produce search results of greatest relevance. Using the hubs and authorities algorithm (Kleinberg, 1999), each URI is viewed as a node linking to many other nodes/URIs, and search results return the highest scoring of each. Basically, by searching we are looking for a central node in the graph, thus a search for PDF will return the PDF MIMEtype node.

Risk Analysis

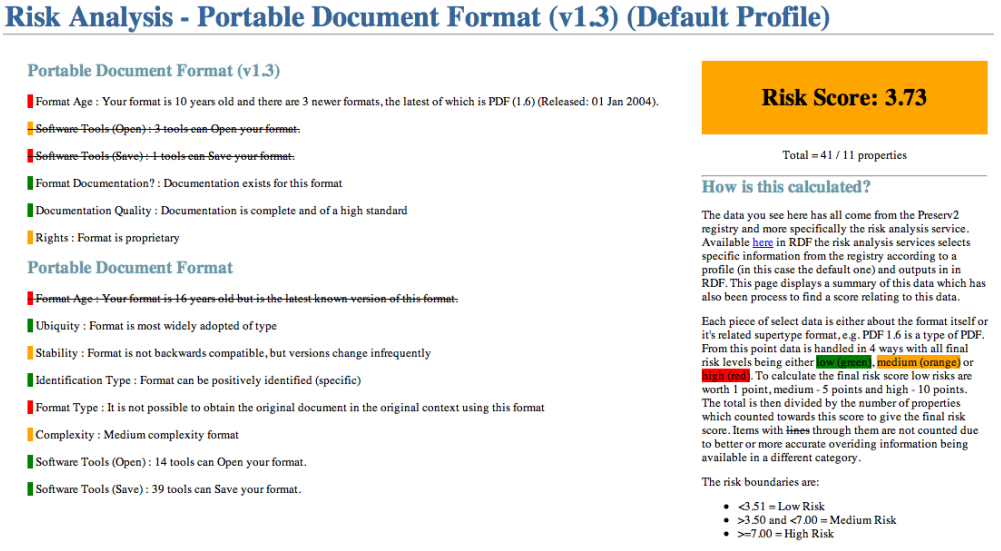

Figure 8. High Level Risk Analysis for P2 Registry.

Figure 8 shows a high-level risk analysis profile for a particular format (PDF 1.3). This profile collates information from the registry's risk analysis service as RDF and then calculates a risk score according to a simple ruleset implemented on top of the registry. This ruleset consists of the data used to construct the profile, either defined by inclusion or exclusion, and then defines how to process the returned values. In Figure 8 we see information presented pertaining to the format type, including documentation, stability, age, as well as the number of manipulation tools.

In this work we are not promoting a new risk assessment technique, simply a process through which all the data pertaining to any format can be gathered and processed automatically. Others, including Rog and van Wijk (2003), Pearson and Webb (2008), and Brown (2003), look in more depth at the criteria and types of assessment which could be critical to analysing risks pertaining to a particular file format. These include both factual data, which can be simply looked up in a registry service (if the data exists), as well as subjective information that is local to the organisation, such as technical knowledge of the format. Interestingly, from a simple merge of data from PRONOM and DBPedia, the P2 Registry can answer many of the factual questions considered by these authors.

As an example, consider the "number of tools?" category. Earlier, we presented an ontology and query which enabled us to group all the types of tools together and thus perform one query to retrieve a list of all the tools available. Obviously doing a count on the number of items in this list would answer the question. However, a user may want to go back and consider the types of tools and thus split this category further. This is why it is critical to expose raw data and not processed data. Processing the data is performed in one of the REST interfaces which sits on top of the registry – it is anticipated that the REST layers are purely for reference. Implementing your own risk profiles and processing techniques will allow an organisation to take its own specific requirements into consideration.

Once all the raw data is gathered from the registry by the processor, a risk score is generated by translating returned data into a low, medium or high risk classification – boundaries are customisable via the profile which the processor uses. Each category is given a score: 1 for low risk, 5 and 10 for medium and high risk, respectively. Then the average of the results is the output risk score. To obtain an overall risk, colour boundaries have been set which translate the final average back into a category. In the case of our PDF example anything below 3.51 is low risk, between 3.5 and 7 denotes medium risk and above 7 represents high risk. Such policies are clearly subjective and it would be wise for data consumers to consider their policy carefully before blindly using this simple example. The P2 Registry has been designed to show the ease with which services can be built on top of the raw data. As such, this policy has its own namespace in the registry so it can easily be ignored.

Earlier, in the JISC Preserv 2 project, a pilot format identification and risk analysis interface was implemented in the EPrints repository software, as shown in Figure 9. The P2 Registry already supports the same set of services to provision information to this interface. However, further modifications are planned to link the EPrints interface back to the registry. This would allow users to browse the contents of the registry from within the EPrints software and see how the risk scores have been constructed.

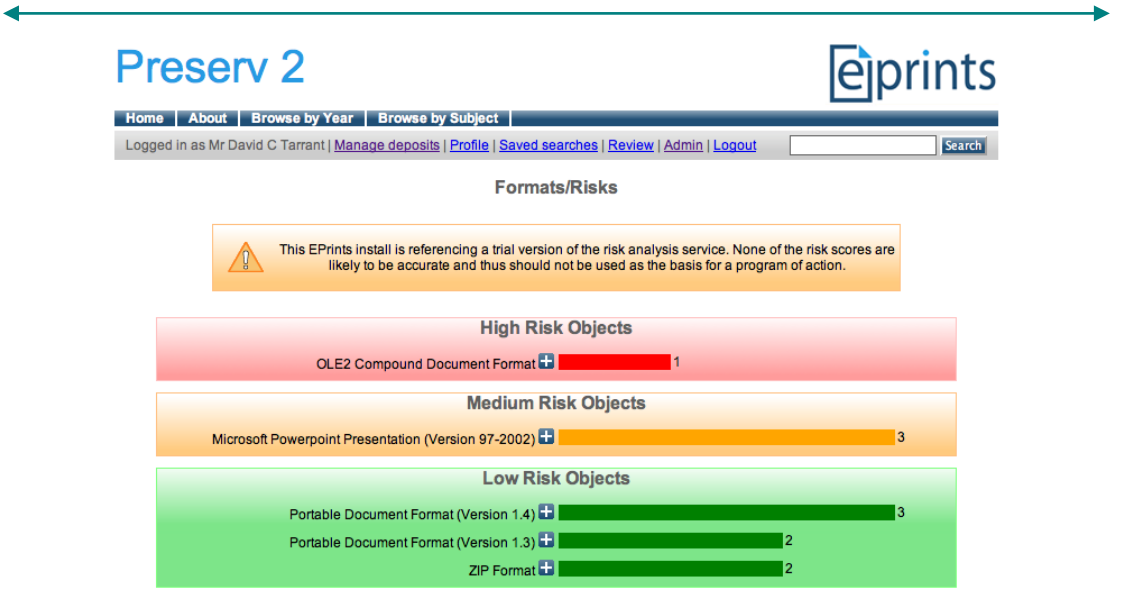

Figure 9. Risk Analysis Interface in ePrints.

Migration Pathways

The last of the current interfaces implemented on top of the registry exists at the profile services level. These provide useful services, as well as demonstrating another of the RESTful interfaces. The migration pathways service is designed for users who need to translate a file from one format to another, and this service simply performs a single SPARQL query to the registry based upon the user's two defined inputs (the input format and the required output format). The software linking ontology outlined in Figure 5 is an essential part of this, as we can simply use the "open" and "save" classes in the query to represent the operation needed to be performed on the two formats.

As an example, say we want to migrate from PDF 1.0 (format 613) to PDF 1.6 (format 637). Our REST interface can be called by simply browsing to the following URL:

http://p2-registry/migration_pathways?
from=http://nationalarvices.gov.uk/pronom/Format/613&
to=http://nationalarchives.gov.uk/pronom/Format/637

Since the from and to arguments represent the URIs of the two file formats, these can simply be substituted into the query shown below in place of ?in_format and ? out_format, respectively.

```
select distinct ?software where {
        ?software ?predicate1 ?in_format .
        ?predicate1 rdf:type SoftwareLink/Open .
        ?software ?predicate2 ?out_format .
        ?predicate2 rdf:type SoftwareLink/Save
}
```

The above shows a single-step query which will only return software URIs that can both open and save in the input and output formats specified. SPARQL has the ability to take this query to an infinite number of steps, thus including intermediary proxy formats which can be used between input and output formats. This also means

that a migration may end up using more than one software package. You have to be careful how many steps you allow this query to take recursively, as it may never return every answer. By default, the P2 Registry iterates only to two-step pathways involving a maximum of one intermediate format.

As a final note, the P2 Registry's REST services attempt to return URIs representing each result. This way, if someone performs a migration which uses softwareA to go from formatA to formatB, the URI representing this exact operation can also be referenced to allow comments to be added to this particular migration pathway. This additional data can then be returned the next time this result appears, to advise future users of the previous experiences of others. Since the whole system uses quads, tracking the provenance of this data is implicit and thus trustworthiness of the data is easily judged.

Conclusions and Future Directions

Empowering the community of Web users to publish data and knowledge on sites such as Wikipedia has already demonstrated the benefits of collaborative content. Currently, digital preservation has a few islands of knowledge that are beginning to publish in a linked data fashion, but there is still some way to go. By harvesting data from just two of these islands, PRONOM and DBpedia, the P2 Registry has demonstrated the benefits gained in terms of increasing the amount of knowledge available and also shown how easy it is to link this knowledge using techniques from the Semantic Web. By building some simple ontologies we have demonstrated how these islands of data can be linked by simply aligning similar concepts, or even just by saying two concepts are exactly the same.

The P2 Registry's set of high-level interfaces go some way to revealing what sort of services could be built on top of the core SPARQL API. By constructing a set of policies we have demonstrated possible ways the data could be processed and thus generate new knowledge, in this case, knowledge relating specifically to file format risk analysis. From results such as the migration pathways URIs, we envisage that other services can start linking to, and commenting on, these results. Thus, a third party could state that it used a certain migration pathway, as represented by a URI, and rate the quality and experience with this to advise others who may use the service at a later date. This has parallels with the rating of items in online stores, such as eBay and Amazon, and brings the P2 Registry full circle: in the first instance it consumed linked data, and by using ontologies and policies it is able to publish new linked data for others to consume.

It is important to see the P2 Registry as an exemplar of what can be achieved if more data is published openly online as simple machine-readable documents. There have been many other projects that have attempted to solve digital preservation problems by developing complex systems and models. These are valuable but don't scale due to lack of data. As an example, the PANIC project (Hunter & Choudhury, 2006) constructed a full (and rather complex) suite of tools to automatically alert users when file formats were becoming obsolete. This project focused more on building systems than gathering data, although they did build a metadata gathering tool with a tightly controlled model. Unfortunately, any data gathered is not published in a location that is easily accessible. Much like the earlier versions of PRONOM, we

suspect this data is only accessible via a SOAP/WDSL web service, a rather complex way to access data which could simply be represented in HTML/XML.

We envisage that systems, such as that developed from PANIC (Hunter & Choudhury, 2006), AONS (Pearson & Webb, 2008) and PLANETS, including the Plato Preservation Planning Tool (Becker et al., 2008) and PLANETS Testbed (Aitken et al., 2008), still have a valuable place in the future of digital preservation, but without a complete set of shared metadata then all of these are limited in their capability to solve preservation problems. What we need now is data, not another system. The P2 Registry is guilty as well on this front, however, because this too is a system, albeit one designed to show that complex preservation problems can become simple when data is available. The P2 Registry was also designed to show that even the schema/model used to express the data is not important, as long as the documentation is available to explain what the terms mean in that model. Having these two pieces of information, the data and the schema description, is the first crucial stage in the process of aligning this data to solve much more complex digital preservation problems.

We have shown in this paper that by aligning Wikipedia data with PRONOM we can increase the returned list of available softwares that can manipulate a PDF file from 19 (using only PRONOM data) to 70. During this process no extra data was curated; the data was simply aligned using a number of simple ontologies. The advantage of keeping all the data model-free, i.e. not creating your own model to import the data into, is that the full provenance chain is preserved and every user can find out exactly what data comes from where. Even the alignment ontologies have provenance which can be followed such that a user can see if they agree with the alignment; if they don't they can simply ignore it and change the query.

This brings us to the final major advantage of the P2 Registry, the fact that it exposes the query interface. This is parallel to opening up a project's mysql server so that anyone can query it. This is not a common idea, perhaps because it is viewed as a security threat, or maybe the model is so complex that not a lot of people would know how to use it.

The future of the P2 Registry is two-fold. First, as a reference platform to allow and encourage publication of preservation data as linked data on the Web. Second, better integration with third-party tools, such as EPrints, by enhancing and completing the high-level interfaces. It goes without saying that it would be great to import more data into the registry. At the time of writing (November 2009) the only organisation exposing data online is digitalpreservation.gov from the Library of Congress. As this data is only exposed via human-readable web pages, the effort to parse this into XML/RDF is notably higher than that required for PRONOM. At the same time GDFR/UDFR were not exposing any data, and projects such as PANIC and PLANETS seemed to be collecting data and not exposing it back to the community via the web.

At this stage the P2 Registry has proved to be a promising and helpful platform for bringing together rich sources of linked data on file format risk information and migration pathways. This approach needs the active participation of the digital preservation community to contribute data by simply publishing it openly on the Web as linked data. By demonstrating the range of services that can be built on top of open

data, it is hoped that more parties will be encouraged to make this part of their core activity and business practice, thus allowing the hard work of building preservation data registries to be distributed across the wider community.

Since the original publication of this paper, the National Archives UK has committed to exposing its file format data, as imported into the P2 Registry, as linked data[5]. This means that the rough importer, which makes lots of assumptions when it imports the data from the PRONOM XML format, is no longer required and the link between the PRONOM data and that from DBPedia can be done directly and not via the proxy established by the P2 Registry. This is excellent news for the community and hopefully the start of a much bigger web of linked data pertaining to digital preservation.

References

Abrams, S. (2007). *File formats*. Installment of DCC, Digital Duration Manual (1). Digital Curation Center.

Aitken, B., Helwig, P., Jackson, AN., Lindley, A., Nicchiarelli, E., & Ross, S. (2008). The planets testbed: Science for digital preservation. *The Code4Lib Journal (3)*. Retrieved March 4, 2011, from http://journal.code4lib.org/articles/83.

Alpert, J., & Hajaj, N. (2008). *We knew the web was big*. Retrieved November 13, 2009, from http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html.

Becker, C., Kulovits, H., Rauber, A., & Hofman, H. (2008). Plato: a service oriented decision support system for preservation planning. *Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries.* Pittsburgh, USA

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. *Scientific American 284, (5):34-53*. Nature: USA.

Berners-Lee, T. (2006). *Linked data*. W3C Design Issues. Retrieved November 13, 2009, from http://www.w3.org/DesignIssues/LinkedData.html.

Bizer, C., Heath, T., Idehen, K., & Berners-Lee, T. (2008). Linked data on the web. *Workshop at the 17th International World Wide Web Conference.* Beijing, China

Brody, T., Carr, L., Hey, J., Brown, A. & Hitchcock, S. (2008). PRONOM-ROAR: Adding format profiles to a repository registry to inform preservation services. *International Journal of Digital Curation 2, (2).*

[5] Linked Data and PRONOM – http://labs.nationalarchives.gov.uk/wordpress/index.php/2010/10/linked-data-and-pronom.

Brown, A. (2003). *Selecting File Formats for Long-Term Preservation.* The National Archives (UK): Digital Preservation Guidance Note (1).

Brown, A. (2005). Automating preservation: New developments in the PRONOM service. *RLG DigiNews 9, (2)*. Research Libraries Group.

Carroll, J.J., Bizer, C., Hayes, P., & Stickler, P. (2005). Named graphs, provenance and trust. *Proceedings of the 14th international World Wide Web Conference.* ACM, ISBN 1595930469: 613-622.

Harris, S., Lamb, N., & Shadbolt, N. (2009). 4store: The design and implementation of a clustered RDF store. *Proceedings of the 5th International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS2009).* Washington DC, USA

Heath, T. (2009). An introduction to linked data. *Proceedings of the Semantic Web Summer School (SSSW2009).* Cercedilla, Spain

Hunter, J., & Choudhury, S. (2006). PANIC: an integrated approach to the preservation of composite digital objects using Semantic Web services. *International Journal on Digital Libraries 6, (2):174-183.* Springer

Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM 46, (5)*. ACM

Knight, G. (2007). *File format typing and format registries*. Sherpa-DP project report.

Manola, F., Miller, E., & McBride, B. (2004). *RDF primer*. W3C recommendation. Retrieved November 13, 2009, from http://www.w3.org/TR/rdf-primer/.

O'Reilly, T. (2007). What is web 2.0: Design patterns and business models for the next generation of software. O'Reilly Media.

Pearson, D., & Webb, C. (2008). Defining File Format Obsolescence: A Risky Journey. *International Journal of Digital Curation 3, (1).*

Ramalho, J., Ferreira, M., Faria, L., Castro, R., Barbedo, F., & Corujo, L. (2008). RODA and CRiB a service-oriented digital repository. *Proceedings of iPRES 2008 Fifth International Conference on Preservation of Digital Objects.* London, UK

Rog, J., & van Wijk, C. (2008) *Evaluating File Formats for Long-term Preservation,* Koninklijke Bibliotheek 2:12-14.

Seaborne, A., & Prud'hommeaux, E. (2008). *SPARQL query language for RDF*. W3C Recommendation. Retrieved November 13, 2009, from http://www.w3.org/TR/rdf-sparql-query/.

Tarrant, D., Brody, T., & Carr, L. (2009). From the desktop to the cloud: Leveraging hybrid storage architectures in your repository. *Proceedings of Open Repositories Conference 2009.* Atlanta, USA

Wilson, A. (2007). *Significant Properties Report*. InSPECT Work Package 2.2.