

Revyu.com: a Reviewing and Rating Site for the Web of Data

Tom Heath and Enrico Motta

Knowledge Media Institute, The Open University,
Walton Hall, Milton Keynes, MK7 6AA, United Kingdom
{t.heath, e.motta}@open.ac.uk

Abstract. Revyu.com is a live, publicly accessible reviewing and rating Web site, designed to be usable by humans whilst transparently generating machine-readable RDF metadata for the Semantic Web, based on their input. The site uses Semantic Web specifications such as RDF and SPARQL, and the latest Linked Data best practices to create a major node in a potentially Web-wide ecosystem of reviews and related data. Throughout the implementation of Revyu design decisions have been made that aim to minimize the burden on users, by maximizing the reuse of external data sources, and allowing less structured human input (in the form of Web2.0-style tagging) from which stronger semantics can later be derived. Links to external sources such as DBpedia are exploited to create human-oriented mashups at the HTML level, whilst links are also made in RDF to ensure Revyu plays a first class role in the blossoming Web of Data. The site is available at <<http://revyu.com>>.

1 Introduction

Revyu.com is a live, publicly usable (and used!) reviewing and rating Web site developed using Semantic Web technologies and standards, and according to Linked Data principles [1] and best practices [2]. Reviews and ratings are widely available on the Web and are one major form of Web2.0-inspired 'user-generated content'. However, despite the availability of reviews through APIs such as Amazon Web Services, this data remains largely in isolated 'silos', and described in formats that hinder its integration and interlinking with data from other sources. This presents considerable barriers to the aggregation of all reviews of a particular item from across the Web. As has been recognised by previous authors [3, 4], the Semantic Web, or Web of Data, provides a technological platform with which to overcome this problem. Revyu takes a significant and concrete step towards this, by exposing reviews using standards such as RDF and SPARQL. In doing so it helps to seed an ecosystem of interlinked reviews, and to bootstrap the Semantic Web as a whole.

2 Revyu Overview

Revyu allows people to review and rate things simply by filling in a Web form. This style of interaction with the site will be familiar to those who have written reviews on

sites such as Epinions¹ or Amazon². Whilst this functionality is not especially novel, as a reviewing application Revyu improves significantly over other work in the area in the following ways: it goes well beyond the closed world 'silos' of sites such as Epinions and TripAdvisor, by exposing reviews in a reusable, machine-readable format; it improves upon the APIs of sites such as Amazon by using a more flexible data format (RDF), allowing more versatile queries via SPARQL, and linking to external data sources; lastly the site takes an open world view of the reviewing process by not constraining users to reviewing items from a fixed database. Anything a user can name can be reviewed, whilst links supplied with the review can disambiguate items thanks to inverse functional properties such as *foaf:homepage*. Consequently reviewers are not restricted to reviews and ratings in one domain, as is the case with Golbeck's FilmTrust [4]. As of August 2007 Revyu has been live for 10 months, attracting 412 reviews from 112 reviewers.

Revyu is built from the ground upwards on Semantic Web technologies. By following Linked Data principles [1] and best practices [2] the site ensures that reviews it hosts can be fully connected into a Web of Data. This approach manifests itself in a number of ways. All site content, in addition to being available in HTML, is also published in RDF/XML that is interlinked with the corresponding HTML pages but available as separate crawlable documents. As we have described elsewhere, this creation and publication of RDF is invisible to the reviewer, enabling novice users to contribute data to the Semantic Web through a familiar, Web2.0-style mode of interaction [5]. To date this approach has yielded over 13,000 RDF triples publicly available on the Semantic Web. Whilst not a large figure by many standards, it is significant that these triples have been generated primarily from direct user input, rather than by data mining, extraction from natural language, or conversion of existing databases.

In addition to review data, RDF describing reviewers, reviewed items, and tags they assign to these is published on the site. These descriptions use the FOAF [6] and Tag [7] ontologies, as well as properties and classes from RDFS and OWL. This data can also be retrieved programmatically via the Revyu SPARQL endpoint³, allowing third parties to access Revyu data for reuse in their own applications. Whilst in some ways analogous to Web2.0 APIs that provide remote query capabilities, SPARQL endpoints afford many advantages to the developer: for example, common libraries can be used to query multiple RDF graphs yet return the results as one resultset, effectively allowing joins over multiple data sources. In the following section we will detail the technical infrastructure underlying Revyu, and discuss decisions made in implementing the system.

3 Revyu Architecture and Implementation

Revyu is implemented in PHP, and runs on a regular Apache web server. The RDF API for PHP (RAP) [8] provides RDF processing capabilities, whilst RDF data is

¹ <http://www.epinions.com/>

² <http://www.amazon.com/>

³ <http://revyu.com/sparql/welcome>

persisted to a de-normalised MySQL database following the RAP database schema. The Revyu SPARQL endpoint relies on the RAP SPARQL engine, which operates against the same MySQL-based triplestore.

From the outset Revyu was designed to adhere to the four 'commandments' of Linked Data outlined by Berners-Lee [1]: using URIs as names for things, using HTTP URIs so people can look up those names, providing useful information when someone looks up a URI, and linking to other URIs so more things can be discovered.

All things represented on Revyu are assigned URIs: reviews, people, reviewed things, tags assigned to things, and even the bundles that represent tags assigned by one person at one point in time. Providing URIs for all these things gives many items a presence on the Semantic Web which they would not have otherwise, and enables any third party to refer to these items in other RDF statements. This opens the way for links between Revyu and other data sets, thereby helping to lay the foundations for a Web of Data.

All URIs in the Revyu URI-space can be dereferenced. Attempts to dereference the URIs of non-information resources receive an HTTP303 "See Other" response containing the URI of a document that describes the resource. This adheres to the W3C Technical Architecture Group's finding on the httpRange-14 issue [9], and serves to reinforce the distinction between a resource and a description of that resource. Content negotiation is also performed on Revyu URIs, whereby the user agent receives a description of the resource in either HTML or RDF depending on the value of the *Accept* header sent in the initial HTTP request.

4 Deriving Semantics from Tagging Data

When creating Revyu, a significant decision was taken to not require users to classify the items they were reviewing, but instead to associate keyword tags with the item. This decision was taken for several reasons: firstly there was seen to be a lack of sufficiently comprehensive classifications of items that users may want to review; secondly, requiring all users to subscribe to a single classification scheme for reviewed items seemed unnecessarily constraining and against the spirit of the Semantic Web; thirdly, providing a usable interface through which non-specialists could classify items using arbitrary types discovered in ontologies on the Semantic Web was seen as unfeasible; and lastly, the coverage provided by ontologies readily available on the Web was deemed insufficient to describe all items that might be reviewed, therefore potentially resulting in a more closed world of reviewed items.

The recent availability of Yago [10] class definitions via DBpedia [11] has gone some way to addressing these issues, and we will be investigating use of these classes in future work. However we believe that tagging retains the appropriate balance of usability whilst also providing sufficient data from which stronger semantics can be derived. At present we use tagging data in two ways: to identify basic semantic relationships between tags and to derive type information about a reviewed item.

Tags that are frequently associated with the same item are assumed to be related in some way. In the HTML pages about each tag, tags that co-occur above a certain threshold are displayed to the user. This threshold is set low for HTML output, as

human readers of the page are unlikely to infer erroneous information based on these relationships. In contrast however, relationships exposed in RDF descriptions of tags (using the *skos:related* property) are based on a more conservative threshold, in order to avoid erroneous inferences based on these assertions. In ongoing work we are investigating the derivation of more precise relationships (such as superclass/subclass) between tags, based on tagging data.

We currently derive type information from tagging data in two domains, books and films, relying on external data sources to help ensure accurate results. Firstly, where items are tagged 'book' we parse Web links provided by the reviewer that relate to the item, and attempt to extract ISBN numbers embedded in these links. Where we are able to extract an ISBN number in this fashion we conclude that the reviewed item is in fact a book, and assert a corresponding *rdf:type* statement into the triplestore.

If an item has been tagged 'film' or 'movie', we execute a query against the DBpedia SPARQL endpoint⁴ in order to find any entries of type *yago:Film* that have the same name as the reviewed item. If a match is found then we conclude this item is in fact a film, and add an *rdf:type* statement to this effect to the triplestore. These type statements for both books and films are exposed in the RDF descriptions of items on Revyu, and also used as the basis for showing additional relevant data in the HTML pages about an item, as detailed in the following section.

5 Production and Consumption of Linked Data

Validating Revyu data against external sources not only allows the derivation of more reliable type information than would be possible using tags alone, it also allows items on Revyu to be linked with others from heterogeneous external data sources such as DBpedia⁵, Open Guides⁶, and FOAF data. Where matches are found, we use the *owl:sameAs* property to assert that two URIs identify the same resource. Publishing these links in RDF helps create a Web of Data rather than simply isolated islands of RDF; Revyu data is *in the Web*, not just *on the Web*.

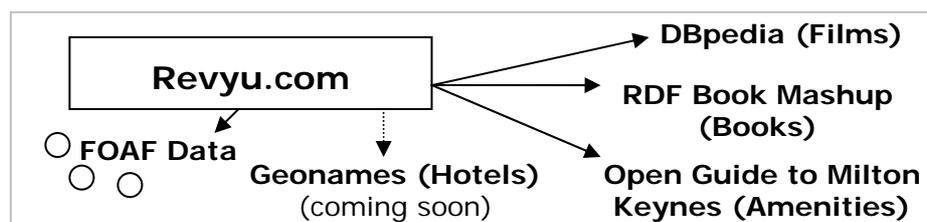


Fig. 1. Links from Revyu.com to external data sets

We actively exploit the links we set between Revyu and external data sources, to enhance the experience of our users without placing an additional burden on

⁴ <http://dbpedia.org/sparql>

⁵ <http://dbpedia.org/>

⁶ <http://openguides.org/>

reviewers by requiring them to supply additional information about the reviewed item. For example, where *owl:sameAs* statements exist linking films on Revyu to their entry in DBpedia, we retrieve additional information about the film, such as the URI of the film's promotional poster, and the name of the director. This information is displayed on the Revyu HTML page about the film (as shown in Fig. 2), thereby enhancing the value of the site for users without requiring this information to be manually entered into Revyu. Similarly we use *owl:sameAs* links between Revyu and the RDF Book Mashup [12] as the basis for retrieving book cover and author information which is also then displayed on the Revyu HTML page about the book (see ⁷ for an example).

In the RDF descriptions of items we take a slightly different approach to that taken with HTML output, choosing to simply expose the links between items without republishing RDF data from external sources. This approach could be described as using Semantic Web data to produce Web2.0-style mashups at the human-readable, HTML level, whilst also mashing up (i.e. linking) data at the RDF level. Not only does this Linked Data approach to mashups reduce issues with licensing of data for republication, it is also a more Web-like approach; duplicating data is of much lesser value than linking to it, and the user agent of the future should be able to 'look ahead' to linked items and merge data accordingly.

It should be noted that we do not claim that the Revyu Web2.0-style mashups represent something that could not have been achieved using conventional Web2.0 approaches. However, the following features distinguish our approach: the simultaneous publishing of data-oriented and human-oriented mashups, so that the data integration effort we have invested is not lost but can be reused by other parties; the ability to easily integrate additional heterogeneous sources using RDF; and the substantially reduced development costs in producing human-oriented mashups through use of Semantic Web technologies.

Whilst to date we have waited for new film reviews on Revyu and then attempted to automatically match them with entries in DBpedia, we are currently preparing for import into Revyu 'skeleton' records covering 12,000 films described in DBpedia. These records simply include the title of the film, a statement indicating that this item is of type '*Film*', a number of keyword tags, and links to the corresponding item on DBpedia. Not only will this provide a foundation on which new reviews can be created, it will also ensure that all films being reviewed in the future will already be interlinked with the corresponding DBpedia entry, and thus the Web of Data.

This skeleton record approach has already been followed when linking Revyu to data from the Open Guide to Milton Keynes⁸, a member of the Open Guides family of wiki-based city guides that expose data in RDF. Milton Keynes is a city in south east England, and home of The Open University. Whilst some amenities in the city, such as pubs and restaurants, were already reviewed on Revyu, many more were listed in the Open Guide due to its longer history. Therefore, after identifying items existing in both locations and making the appropriate mappings to avoid duplication, we created skeleton records in Revyu for the remaining items, setting links back to their Open Guide URIs. This has enabled latitude and longitude data for many items to be

⁷ <http://revyu.com/things/the-unwritten-rules-of-phd-research/about/html>

⁸ <http://miltonkeynes.openguides.org/>

retrieved from RDF exposed by the Open Guide, and used to show a Google Map of the items location (see ⁹ for an example). The same approach can also be used to expose address, telephone, and opening time information held in the Open Guide.

Broken Flowers



directed by [Jim Jarmusch](#)

Broken Flowers is a 2005 comedy-drama film directed and written by

Fig. 2. Excerpts from the Revyu HTML page about the film Broken Flowers, showing the film poster, director information, and summary drawn from DBpedia¹⁰

About tom (Tom Heath)



[tom's Home Page](#)

tom's location:
[Borough of Milton Keynes](#)

tom's Interests

[Semantic Web](#)
[Word of Mouth](#)
[Trust](#)
[Beer](#)

Fig. 3. Excerpts from the first author's Revyu profile page, showing data sourced automatically from his external FOAF file¹¹

⁹ <http://revyu.com/things/ye-olde-swan-woughton-on-the-green-milton-keynes/about/html>

¹⁰ <http://revyu.com/things/broken-flowers-film-movie-bill-murray-jim-jarmusch-sharon/about/html>

¹¹ <http://revyu.com/people/tom/about/html>

Similar principles are also applied to user information, such that people registering with the site are not required to provide copious information to populate their user profile. Instead, where they have an existing FOAF description in an external location they may provide its URI, in which case Revyu dereferences this URI and queries the resulting graph for relevant information (such as a photo, location, home page address, and interests), which is then displayed on their profile page, as illustrated in Fig. 3. This approach reduces the burden on the user by not requiring them to manage multiple redundant sets of personal information stored in different locations. Furthermore, where the user has assigned themselves a URI in their FOAF description, Revyu sets *owl:sameAs* links asserting that this URI identifies the same resource as the user's Revyu URI. Users can also state that they know other Revyu reviewers, at which point this relationship is recorded in the triplestore using the *foaf:knows* property, and exposed (privacy settings permitting) in the user's RDF description on the Revyu site. This ensures that social networking data created in one location is not automatically rendered inaccessible to other services.

6 Future Work and Conclusions

In addition to encouraging further user participation in order to increase the value delivered by the site, we plan to integrate Revyu with a number of additional data sets. Most notably we are preparing to create skeleton records in Revyu of 70,000 hotels worldwide, linked to their corresponding entry in the *Geonames* dataset. The same approach will also be used to link Revyu with data from other Open Guides, such as London and Boston. Additional data will be integrated as further relevant sources become available.

It should be noted that our aim in linking to external datasets is not to constrain, but merely to seed, users conceptions of what can be reviewed. As we integrate further data sets we hope to achieve a more automated linking process by investigating generic similarity matching techniques for operation on the wider Semantic Web.

Whilst frequently suggested as an additional feature, at present there are no concrete plans to import external review data into Revyu, for a number of reasons. Firstly, to the best of our knowledge Revyu is the only site serving reviews as Linked Data according to current best practices, which limits our abilities to interlink Revyu with external review data sets; secondly, little review data is available under a suitable license; lastly, our ongoing research is predicated on the ability to combine review data with social networks, requiring some global identifier (such as *foaf:mbox_sha1sum*) to be available for each reviewer. This is rarely the case with traditional reviewing sites. By providing reviews in a reusable format that is easily integrated and interlinked with other data, Revyu provides core data for our ongoing work into information seeking, recommendation, and trust in social networks on the Web.

In conclusion, in this paper we have described Revyu, a human usable reviewing and rating Web site built on Semantic Web technologies, and fundamentally designed to contribute to the realization of a Web of Data. Whilst superficially not unique in

functionality, the site is rare in its status as a publicly available service in daily use that is oriented towards human users, yet also embodies current best practices in developing for the Semantic Web.

Acknowledgements

This research was partially supported by the Advanced Knowledge Technologies (AKT) and OpenKnowledge (OK) projects. AKT is an Interdisciplinary Research Collaboration (IRC) sponsored by the UK Engineering and Physical Sciences Research Council under grant number GR/N15764/01. OK is sponsored by the European Commission as part of the Information Society Technologies (IST) programme under grant number IST-2001-34038. Peter Coetzee did a superb job of turning data into skeleton records for import into Revyu. Lastly, the Open Guides and DBpedia communities, and the RDF Book Mashup team deserve our special thanks.

References

1. Berners-Lee, T.: Linked Data. <http://www.w3.org/DesignIssues/LinkedData.html> (2006)
2. Bizer, C., Cyganiak, R., Heath, T.: How to Publish Linked Data on the Web. <http://sites.wiwiw.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/> (2007)
3. Guha, R.: Open Rating Systems. In: Proc. 1st Workshop on Friend of a Friend (2004)
4. Golbeck, J., Hendler, J.: FilmTrust: Movie Recommendations using Trust in Web-based Social Networks. In: Proc. IEEE Consumer Communications and Networking Conference (2006)
5. Heath, T., Motta, E.: Ease of Interaction plus Ease of Integration: Combining Web2.0 and the Semantic Web in a Reviewing Site. *Journal of Web Semantics*, 5 (to appear)
6. Brickley, D., Miller, L.: FOAF Vocabulary Specification 0.9. <http://xmlns.com/foaf/0.1/> (2007)
7. Newman, R., Russell, S., Ayers, D.: Tag Ontology. <http://www.holygoat.co.uk/owl/redwood/0.1/tags/> (2005)
8. Oldakowski, R., Bizer, C., Westphal, D.: RAP: RDF API for PHP. In: Proc. 1st Workshop on Scripting for the Semantic Web, 2nd European Semantic Web Conference (ESWC2005) (2005)
9. W3C Technical Architecture Group: httpRange-14: What is the range of the HTTP dereference function? <http://www.w3.org/2001/tag/issues.html#httpRange-14> (2005)
10. Suchanek, F. M., Kasneci, G., Weikum, G.: Yago: A Core of Semantic Knowledge - Unifying WordNet and Wikipedia. In: Proc. 16th International World Wide Web Conference (WWW2007) (2007)
11. Auer, S., Lehmann, J.: What have Innsbruck and Leipzig in common? Extracting Semantics from Wiki Content. In: Proc. 4th European Semantic Web Conference (ESWC2007) (2007)
12. Bizer, C., Cyganiak, R., Gauss, T.: The RDF Book Mashup: From Web APIs to a Web of Data. In: Proc. 3rd Workshop on Scripting for the Semantic Web, at 4th European Semantic Web Conference (ESWC 2007) (2007)