Service Orchestration for Linking Open Data: Applying a SOA principle to the Web of Data

Christian Sadilek[1], Rainer Simon[1], and Bernhard Haslhofer[2]

[1] Austrian Institute of Technology, Digital Memory Engineering Group, Donau-City-Str. 1,
1220 Vienna, Austria
{christian.sadilek, rainer.simon}@ait.ac.at

[2] University of Vienna, Department of Distributed and Multimedia Systems, Liebiggasse 4/3-4,
1010 Vienna, Austria
bernhard.haslhofer@univie.ac.at

Abstract. The rising popularity of RESTful Web services has recently motivated the extension of existing service orchestration engines to support the composition of services that do not rely on machine-readable descriptions. At the same time, within the Linking Open Data initiative, data sets are published conforming to the Linked Data principles, which can be naturally achieved by exposing RDF data through RESTful interfaces. In this position paper, we motivate the use of service orchestration to define workflows for interlinking open data. We introduce the design of an abstract workflow for the semantic enrichment of such data with the purpose of providing an integrated view on otherwise isolated data sources. Finally, based on this abstract workflow we present early work on a concrete implementation and report on our experiences.

1 Introduction

The W3C Linking Open Data (LOD) community project[1] aims to create a Web of interlinked datasets based on the Resource Description Framework (RDF) and following the Linked Data principles[2]. These principles demand that exposed data objects on the Web are identified with dereferenceable HTTP URIs, providing both human and machine readable representations and include links to other resources. These demands are inherently supported for data objects exposed through services that follow the REST architectural style [1], also referred to as RESTful Web services [2]. Likewise, Battle and Benson point out that RDF is capable of semantically describing and aligning data from disparate sources, but the lack of standard or agreed-upon access methods prevents the widespread use of such data, and consequently argue that the REST methodology is a natural fit for Semantic Web operations as it integrates well with its resource paradigm [3]. The emerging number of frameworks (e.g. RESTEasy[3], Restlet[4]), APIs (e.g. Amazon[5], Yahoo[6]) and programming standards [4] indicate the recently growing popularity

[1] http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData/

[2] http://www.w3.org/DesignIssues/LinkedData.html

[3] http://jboss.org/resteasy

[4] http://www.restlet.org/

[5] http://aws.amazon.com

[6] http://developer.yahoo.com

of RESTful Web services. This popularity has led to extensions of existing service orchestration engines like Apache ODE[7] to incorporate this type of services, which are not built upon machine-readable descriptions. As a fundamental building block of service oriented architectures, service orchestration is an approach to service composition that describes how Web services interact [5]. It enables the definition of executable workflows as combination of Web services to support compositionality and reuse without demanding locally deployed services [6]. In this paper, we argue that due to these benefits, executable workflows can simplify and further facilitate the interlinking of open data. We present the design and implementation of a workflow for semantic enrichment of unstructured content from existing applications to integrate it in the Web of Data.

2 Background and Related Work

Service orchestration for Semantic Web services is also addressed by Nitzsche et al. presenting an extension to BPEL[8] using a WSDL-less interaction model [7]. The extension, called BPEL4SWS, motivated the implementation of an execution engine for semantic business processes [8]. Other approaches are based on templates [9] or a novel programming language [10]. For our work, we are aiming to use standard-conforming BPEL instances without process description extensions to facilitate the support for different service orchestration engines which we argue is crucial to maintain reusability.

A system that employs Named Entity Recognition (NER) and human feedback to link news content to DBpedia[9] is shown by Kobilarov et al. in [11]. A framework named Silk that enables discovering RDF links using a declarative language for specifying link conditions is presented by Bizer et al. in [12], and Raimond et al. introduce an algorithm for automatic interlinking of music datasets in [13]. Further, a solution for finding equivalent geospatial entities in the Web of Data for creating links between them is given by Auer et al. in [14].

We present a semi-automatic approach which incorporates workflows and human feedback. Our work is complementary to the aforementioned approaches as it also operates on extracted entities within different data sources in order to discover relationships between them. In fact, if Web service interfaces were implemented on top of these existing components they could be employed as services within workflows for interlinking open data. Within our workflows an optional step is carried out prior to link discovery to identify named entities of different types in unstructured content using existing entity recognition services.

3 Workflow Design

The abstract workflow depicted in Fig. 1 is suitable for interlinking open data. The workflow exposes a service that takes a query string as parameter and initiates the workflow's execution. This query string is used to formulate a search request for a LOD

[7] http://ode.apache.org

[8] www.oasis-open.org/committees/wsbpel

[9] http://dbpedia.org

source to retrieve the source data set. The first activity of the *Link Resources* scope invokes an *Entity Recognition Service* in order to analyze textual fields and return named entities found in the text. This activity can be skipped if link discovery can be carried out based on existing entities in the source data. The subsequent invocation of the *Link Discovery Service* aims to retrieve dereferenceable URIs for the identified entities. As RDF/XML is assumed to be the representation format, it is possible to execute XPath[10] expressions against the source data and to formulate a loop for iterating through all resources in the source data set. The loop iterations can safely be executed in parallel because iterations operate on distinct resources without any shared mutable state. However, avoiding the overutilization of external services is an argument for sequential loop iteration. The described two-step approach to linking resources offers a certain flexibility. For example: A Geoparser[11] could be employed as an *Entity Recognition Service* to parse unstructured content for location names and return the discovered entities embedded in XML. Then a service like GeoNames[12] could be used as a *Link Discovery Service* to return dereferenceable URIs for the corresponding locations. We will describe an additional application of this approach in greater detail in the next section.

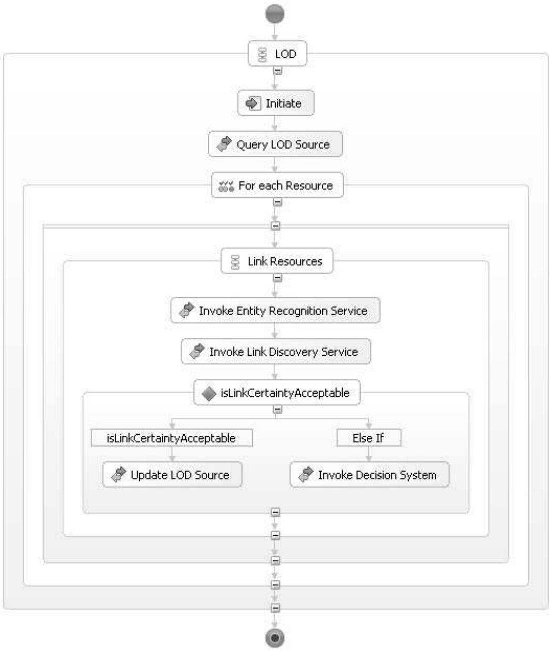
LOD
Initiate
Query LOD Source
For each Resource
Link Resources
Invoke Entity Recognition Service
Invoke Link Discovery Service
isLinkCertaintyAcceptable
isLinkCertaintyAcceptable
Else If
Update LOD Source
Invoke Decision System

Fig. 1. Abstract workflow for interlinking open data. Assign activities and result polling have been omitted for brevity.

[10] http://www.w3.org/TR/xpath
[11] http://geoparser.digmap.eu/
[12] http://www.geonames.org

If the returned URIs are complemented by values that express the certainty of the matching concept, or in the case that multiple URIs are returned for ambiguous entities, a decision will have to be made before assigning the link and updating the resource to ensure the semantic validity. This decision is based on the condition of the activity *isLinkCertaintyAcceptable*. It could for example be based on a threshold for the determined certainty or on an XPath count operation for the number of links in the provided service output. If the condition evaluates to *true* the link to the URI will be created and assigned to the resource. Finally, the resource is modified by sending an update request to the LOD source. This is done by either sending an SPARQL/Update[13] statement, if provided by the *Link Discovery Service*, or using a HTTP PUT request to send the updated triples as RDF/XML representation. In case the condition evaluates to *false*, a separate service will be invoked that collects decisions to be made collaboratively by users. In this case the resource will be updated as soon as the positive decision has been made.

For the invocation of asynchronous operations on RESTful Web services we have decided to use a result polling mechanism as an alternative to callbacks based on message correlation. The latter requires the introduction of state to any implementation case which violates the key REST principle of stateless communication. It is therefore assumed that for such invocations on RESTful Web services a task or job identifier is synchronously returned which in turn can be used to frequently poll for the result within a loop activity as shown in Fig. 2. This activity is terminated as the result becomes available.

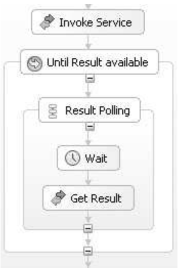

Fig. 2. Invocation of a RESTful Web service with result polling as an alternative to callbacks based on message correlation.

4 Workflow Implementation

The prototypical implementation of the abstract workflow presented in the previous chapter makes use of a LOD source for multimedia annotations. This data source is utilized by a set of Web applications that enable users to collaboratively associate their knowledge with multimedia objects and fragments in the form of unstructured text. The

[13] http://www.w3.org/Submission/SPARQL-Update/

use case we implement is to semantically enrich these user-contributed annotations by linking them to matching geographic resources using the *rdfs:seeAlso* predicate with the purpose of being able to provide an integrated view on these resources. Although the workflow is currently executed on a batch job basis, the annotation application's architecture would allow to configure workflow invocations on certain pre-defined application events (e.g. when an annotation is created, referred to or updated).

The workflow is implemented in BPEL 2.0 using Apache ODE 1.3.3 as a service orchestration engine. ODE includes a set of WSDL 1.1 extensions to describe invocations of RESTful Web service operations. The workflow is querying the recently added annotations at the RESTful annotation backend. The user contributed content is then sent to OpenCalais[14], a service that creates semantic metadata based on unstructured text. The OpenCalais Web service returns the metadata as resources represented in RDF/XML. Therefore, the relevant links can be extracted and added to the annotation model. OpenCalais serves as both an *Entity Recognition Service* and a *Link Discovery Service* here. The provided *resource type* and *relevance* fields can be used to decide which links should be stored directly and which should be accepted by users first and therefore need to be sent to the decision system. In this example, we have decided to accept all links to cities and countries (of types *http://s.opencalais.com/1/type/er/Geo/City* and *http://s.opencalais.com/1/type/er/Geo/Country*) with a relevance score above 0.7. It is only a matter of adapting the selection criteria to change the workflow to identify and link to entities of a different domain or category like person for instance.

5 Conclusions

In this paper, we have exemplified how orchestration of RESTful Web services can facilitate the interlinking of open data. We have defined a basic abstract workflow that utilizes named entity recognition and link discovery services, and is augmented by a decision stage in which ambiguous link occurrences can be resolved through human intervention. Furthermore, we have motivated that LOD sources should support updates of resources using either an RDF/XML representation or the SPARQL/Update language to enable the collaborative interlinking of open datasets. We have not aimed to create semantic bridges for existing services by either mapping fields and parameters to an ontology or by wrapping or even recreating services to map REST operations to Semantic Web operations. The only requirement for the services used was the existence of an XML Schema[15] to define how to access input and output data.

The initial motivation for our work was that we currently see a gap between the vast amount of content created on the Web in partly unstructured, distributed and collaborative manner and the Web of Data which is built on the notion of publishing structured data using machine readable languages. We have shown that this challenge can be addressed by employing established technologies for Enterprise Information Systems: By offering an infrastructure that can access content from various sources and tie together different named entity recognition and link discovery services quickly and effortlessly, they can unlock content from existing applications for the domain of the Semantic Web.

[14] http://www.opencalais.com

[15] http://www.w3.org/XML/Schema

For future work, we will focus on improving the decision system to better support the (collaborative) human interaction within the workflows. We plan to experiment with a broad range of services using different service orchestration engines and workflow languages to research how to efficiently modify records in the Web of Data using executable workflows and to find ways to integrate these workflows into existing applications.

6 Acknowledgements

This work has been partially supported by the EuropeanaConnect Targeted Project for digital libraries, as part of the eContentplus Program of the European Commission.

References

1. Fielding, R.T.: Architectural styles and the design of network-based software architectures. PhD thesis, University of California, Irvine (2000)
2. Richardson, L., Ruby, S.: RESTful Web Services. O'Reilly (2007)
3. Battle, R., Benson, E.: Bridging the semantic Web and Web 2.0 with Representational State Transfer (REST). J. Web Sem. **6** (2008) 61–69
4. Hadley, M., Sandoz, P.: JAX-RS: The Java API for RESTful Web Services. Java Specification Request (JSR) 311 (2007)
5. Peltz, C.: Web Services Orchestration and Choreography. Computer **36** (2003) 46–52
6. Chen, L., Wassermann, B., Emmerich, W., Foster, H.: Web service orchestration with BPEL. In: ICSE '06: Proceedings of the 28th international conference on Software engineering, New York, NY, USA, ACM (2006) 1071–1072
7. Nitzsche, J., van Lessen, T., Karastoyanova, D., Leymann, F.: BPEL for Semantic Web Services (BPEL4SWS). In: On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops, Springer (2007) 179–188
8. Lessen, T., Nitzsche, J., Dimitrov, M., Konstantinov, M., Karastoyanova, D., Cekov, L., Leymann, F.: An Execution Engine for Semantic Business Processes. Service-Oriented Computing - ICSOC 2007 Workshops: ICSOC 2007, International Workshops, Vienna, Austria, September 17, 2007, Revised Selected Papers (2009) 200–211
9. Sirin, E., Parsia, B., Hendler, J.: Template-based Composition of Semantic Web Services. In: AAAI Fall Symposium on Agents and the Semantic Web, Virginia, USA (2005)
10. Hong-Hua, C., Shi, Y., De-Hui, D., Yang, X.: Orchestrating Semantic Web Services with Semantic Programming Language. Semantic Computing and Systems, IEEE International Workshop on **0** (2008) 101–106
11. Kobilarov, G., Scott, T., Raimond, Y., Oliver, S., Sizemore, C., Smethurst, M., Bizer, C., Lee, R.: Media Meets Semantic Web — How the BBC Uses DBpedia and Linked Data to Make Connections. In: ESWC 2009 Heraklion: Proceedings of the 6th European Semantic Web Conference on The Semantic Web, Berlin, Heidelberg, Springer-Verlag (2009) 723–737
12. Bizer, C., Volz, J., Kobilarov, G., Gaedke, M.: Silk - A Link Discovery Framework for the Web of Data. In: 18th International World Wide Web Conference. (2009)
13. Raimond, Y., Sutton, C., Sandler, M.: Automatic Interlinking of Music Datasets on the Semantic Web. In: WWW 2008 Workshop: Linked Data on the Web (LDOW2008), Beijing, China (2008)
14. Auer, S., Lehmann, J., Hellmann, S.: LinkedGeoData - Adding a Spatial Dimension to the Web of Data. In: Proc. of 7th International Semantic Web Conference (ISWC). (2009)