Service Choreography Meets the Web of Data Via Micro-Data

Xi Bai and Dave Robertson
School of Informatics, University of Edinburgh, EH8 9AB, UK.
xi.bai@ed.ac.uk, dr@inf.ed.ac.uk

Abstract
Several solutions exist for semantically describing Web Services (WSs) from the perspective of orchestration but little is known about how semantics benefit WS choreography. The most extreme example of a choreography problem occurs in peer-to-peer systems where shared semantics of data may need to be established via services interactions. We present a solution to this problem by sharing micro-data via interaction models. No pre-unified ontology is required in our approach so peers can make use of existing heterogeneous resources having been described in the RDF data model flexibly and compatibly. The experimental results indicate that our approach semantically enhances WS choreography in a lightweight way which complies with principles of Linked Data and republished Interaction Models (IMs) can further facilitate the progress of the Web of data as well as the formation of peer communities generated through peers’ interactions.

Introduction
Choreography and orchestration are two perspectives from which researchers currently investigate Web Service (WS) integration. The former describes the collaborative interactions between services and the latter describes service composition from the perspective of a single peer. Busi et al. defined a notion of conformance between choreography and orchestration which allows them to state when an orchestrated system conforms to given choreography (Busi et al. 2005). From the perspective of orchestration, several Semantic Web Service (SWS) description languages (Martin et al. 2004) (Akkiraju 2005) (Farrell and Lausen 2007) have been proposed but not adequate attention has been paid to SWS choreography. WS-CDL (Kavantzas et al. 2004) is a language for describing WS choreography and like other WS description languages, it lacks an support for semantics.

From the perspective of choreography, peers interact with each other in terms of specific protocols and the OpenKnowledge ¹ project has developed a peer-to-peer knowledge sharing system in which choreography is described in an Interaction Model (IM, encoded in Lightweight Coordination Calculus (LCC) (Robertson 2004)). An IM is a set of clauses defining the behaviors associated with roles within peer interactions. Here, a role describes the necessary actions for one of the peers who will take part in the interaction. An IM actually describes a WS model. Although LCC provides a formal way of describing choreography, since it is not a typed language, currently there is no semantic information associated with any interactions. As shown in the next section, without the support of semantics it is difficult if not impossible to match the user’s query with a helpful service (IM) so the discovery recall is affected due to this lack. It is not suitable for revising the LCC interpreter and making it parse data types as ontological concepts because of following two reasons: firstly, LCC should stay lightweight in accordance with the distributed knowledge sharing environment; secondly, ontological information is boundless and there is no parser that is able to cover all ontologies. Keyword-based IM publication (Kotoulas and Siebes 2007) limits the service discovery in the OpenKnowledge system due to the synonymity and the ambiguity of phrases. On the other hand, because of limited computation capabilities that mobile devices have, relatively lightweight service description solutions should be offered to service publishers. Nowadays, more users take search engines as service discovery agents. Within the movement of Linked Data (Berners-Lee 2006), there will be a tendency to publish service descriptions complying with Link Data principles and facilitating the progress of the Web of data.

In this paper, we propose a semantically enhanced approach for republishing IMs. Unlike the conventional way of providing peers with SPARQL endpoints to make them query other peers’ Knowledge Bases (KB), by employing micro-data, our approach makes peers not only publish their IMs with semantics on normal Web pages but also just expose knowledge that they want to be advertised and keep sensitive knowledge confidential. The reason for publishing IMs on Web pages in HTML/XHTML is because more and more search engines have begun to index Web pages with micro-data nowadays (e.g., Yahoo! SearchMonkey and Google Rich Snippets). These search engines can be taken as natural discovery services for the OpenKnowledge system. On the other hand, micro-data-embedded Web pages are both human-readable and machine-readable, so users

¹http://www.openk.org/
who want to create new IMs by revising existing ones can thoroughly understand the content on Web pages. Since
good resources described using arbitrary RDF vocabularies can
be embedded in the Web pages, peers can employ any ontol-
ogies to publish IMs and suitable matchmakers to align
their profiles with IMs published by other peers so as to harness heterogeneous resources. When they do not know the
ontologies used in a specific IM, RDF search engines like Sindice
2 and mapping services like sameAs3 can provide recommendations. This way of republishing IMs complies with the four principles of Linked Data and will facilitate the progress of the Web of data.

In the remainder of the paper, we first present a method for modeling interactions between peers in a distributed en-
vironment using LCC. We then give an introduction to the
employed vocabulary about IM and peer-to-peer communities. This vocabulary can assist IM publishers in associating
micro-data with IM elements and is also extensible and re-
placeable. We detail our approaches for IM republication and
IM consumption as well as their design and preliminary implementation. Finally, we give experiments with our ap-
proach and case studies which indicate that republished IMs can help peers discover desired services and further establish peer communities.

LCC-based Choreography Description
In a peer-to-peer network, peers are equal to each others and
each of them has both server and client capabilities. From
the perspective of choreography, peers collaborate through
interactions and we use LCC to describe the choreographies inside the peer-to-peer network. Although our paper em-

dows LCC, the specific choice of the process language is not essential to the core arguments of this paper. The syntax
of LCC is described in (Robertson 2004) and will not be iter-
a
dated here. We have this choreography description language now but how do peers share these IMs in the peer-to-peer
environment? Figure 1 depicts an example that a client pur-
chases a product referenced by a product code from a shop using his or her credit card.

However, since there is no explanation about any elements (e.g., roles, messages and constraints) in this IM, it is diffi-
cult if not impossible for peers who want to purchase or sell a product to automatically recognize whether this IM is ex-
actly the one they really need. The IM in Figure 1 is neither human-readable nor machine-readable. We will not under-
stand what CC denotes unless we read the comment at the
top of the figure. Although the IM publisher can use more explicit name of argument like CreditCard instead of CC, it is still not machine-readable, because a service will not know that CreditCard denotes a thing that is of type credit card.
Published IMs should be human-readable because users who are familiar with LCC may also revise an exist-
ing IM when it does not meet their requirements. Suppose a user originally does not know that this IM can provide the service he or she desires. Suppose he or she creates a query by typing in keywords like buy, credit card. Since there is no

string in this IM can match these keywords, this query will be ignored by the keyword-based discovery service. Need-
less to say, IMs without semantical enhancement will affect the discovery recall. Service discovery and reuse are both hampered by the lack of semantical information about IMs. In this paper, we propose an approach for semantically en-
hancing interactions between peers from the perspective of choreography using micro-data. Figure 2 gives the excerpt of proportional source codes of a Web page on which the trade IM described in Figure 1 has been republished. Our approach for fulfilling this semantical enhancement will be detailed and further discussed in the following sections.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{Simple trade IM in LCC}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure2.png}
\caption{Republished trade IM in XHTML}
\end{figure}
Web-Based Peer-to-Peer Topology

The vocabulary used for republishing IMs has two important notions are Peer and Community. The prime community can be constructed by automatically discovering peer groups (Bai, Cheng, and Robertson 2009). This vocabulary named as OPENK 4 has been established using 17 classes and 38 properties. In OPENK, three important notions are openk:Peer, openk:P2PCommunity and openk:IM which are coined in terms of concepts involved in the OpenKnowledge system. Since the FOAF (Bojars and Breslin 2007) vocabulary is good at describing persons, activities and relationships, we use it to describe peers and their relationships. We also try our best to keep the extensibility of the peer community notion using the SIOC (Brickley and Miller 2007) vocabulary. IM ontology was created in terms of semantics inside the OpenKnowledge kernel language (LCC), which gives us an intimate connection between interaction processes and the community ontology. Each peer has a profile that is composed of triples about its features such as which roles this peer can play and which OKCs this peer can provide. An OKC (OpenKnowledge Component) is a plug-in component that contains methods used for solving constraints in the IM (sometimes human interventions are also involved in the process of constraint solving). Based on four principles of Linked Data, the Web is essentially very close to a Peer-to-Peer network because each user on the Web can digest existing data as a consumer and publish new data and interlink it with other data as a contributor for the formation of the Web of data. Figure 3 gives an illustration about the Web-based peer-to-peer network. Each peer has a KB composed of a vocabulary, a peer profile, an IM repository and an OKC repository. Users log on to peers or join a specific community via their user accounts.

Figure 3: Peer-to-peer network topology

Linking Choreography to the Web of Data

In this section, we present our semantically enhanced approach for republishing IMs for the purpose of linking them to the Web of data. The corresponding republishing tool is designed and implemented, which assists IM publishers in annotating IMs and semi-automatically generating Web pages with micro-data. Methods for consuming micro-data are given based on the OpenKnowledge system. Micro-data provides a lightweight way of embedding semantic information into Web pages. Several formats such as eRDF, Microformat and RDFa were proposed. eRDF does not support full RDF. Microformat requires users to create new data models for new formats. RDFa allows users to embed any resources complying with the RDF model and has been a W3C recommendation. Here, we use RDFa in our approach and make it carry the semantic that an IM element has.

Micro-Data Injection

In our approach, the micro-data embedding strategy is employed in within the IM publishing process. An IM defines the behaviors of roles involved in the interaction, so users can use whatever ontologies they prefer to enhance and republish IMs in a cut-and-paste way. On the other hand, when peers receive IMs published by others using heterogeneous ontologies, they can also use local ontology matchmakers to fulfill the mapping task. Conventionally, users are allowed to query RDF repositories through exposed SPARQL endpoint. Using micro-data-embedded Web pages to publish IMs and exchange them with other peers, IM publishers can keep their own private knowledge confidential and just expose the knowledge about IMs they want other peers to know. Currently, our republishing tool provides following functionalities: LCC editing, peer profile browsing, IM annotating, annotation revising and annotation issuing. These functionalities are achieved by designing and implementing the following modules:

a. **Preprocessing Module** is used for assisting a peer in importing the local profile the IM it expects to be published.

b. **Profile Browsing Module** is dedicated to display auxiliary information derived from peers’ profiles.

c. **Annotating Module** provides a drag&drop way for publishers who want to attach IM elements with semantics.

d. **Revising Module** helps IM publishers delete, modify and replace existing annotations before the final publication.

e. **Issuing Module** has two functionalities, one of which is harvesting embedded RDFa of the IM as new knowledge using RDFa parsers 5 and adding it to the peer’s local profile. The other task is generating Web page and publishing it on the Distributed Discovery Service (DDS).

The next section will present the preliminary implementation for assisting users in republishing IMs based on the above design as well as an example for republishing an IM that describes a peer sends a greeting text to another peer and the recipient then displays the received text.

---

4http://homepages.inf.ed.ac.uk/s0896253/openk.owl

5http://rdfa.info/rdfa-implementations
**Micro-Data Consumption**

Republished IMs can be consumed in various ways and one of them is to assist the DDS in discovering desired IMs. The DDS is one of the most important components for the Open-Knowledge system. IMs are located on different peers in a distributed way and the DDS is in charge of discovering IMs that meet the user’s requirement and other peers the user can collaborate with in terms of his or her input query. Here, “query” means key words or URIs. Since URIs are hard to remember and some URIs are heterogeneous, users can use URIs discovered by RDF search engines like Sindice or URIs curated by DBpedia through the Lookup service. Republished IMs have been semantically enhanced and by harvesting embedded RDFa, the DDS can provide a more precise and more extensible query processing thanks to dereferenceable URIs on the Web of Linked Data.

Another way of consuming embedded RDFa happens when an IM is executed after all required roles are filled by peers. It is notable that one role may be filled with more than one peer and peers are allowed to select partners to collaborate with based on the Peer Ranking mechanism of the OpenKnowledge system. Thereafter, a peer will be randomly selected as the coordinator by the DDS, which takes charge of bootstrapping the interaction and executing the LCC protocol. Within the process of running an IM, users normally need to query the data embedded in Web pages. However, this query is different from the aforementioned query that triggers discoveries of appropriate IMs and collaborative peers. Here, the query will be more specific and more targeted such as a tourist asking an airline service for a cheap flight with a concrete destination and relative departure time and arrival time or an unemployed man asking a job vacancy service for a job with a reasonable salary and a location close to where he or she lives. In these cases, the OKCs at the side of the service provider will take charge of parsing the Web pages (e.g., Web pages describing available jobs that have been published by a job vacancy Website using RDFa). By matching the harvested RDF data against with the user’s query, the coordinator will find out the information that meets the user’s requirement.

**Experiment and Case Study**

We used PCs with Intel Pentium Duo 3.0GHz CPUs and 1 GB of RAM to do the experiment. As mentioned above, one way of harnessing republished IMs is to assist the DDS in discovering appropriate IMs that meet the peers’ requirements. In the peer-to-peer network, the atomic interaction occurs between two peers. We took two PCs with the same performance as two peers and each of them has been installed with an Apache server that can communicate with the server running on the other peer. This actually makes each machine have both client and server capabilities. IMs owned by each peer have been published on the DDS and also stored in the local IM repository. Suppose a peer sends a query (both URI-based queries and phrase-based queries are supported) to a DDS peer denoted by \( D \). By matching the query against with micro-data embedded in published IMs, \( D \) can select most relevant IMs which are likely to meet the requirement of a user who has logged on to \( P \) and created that query. Several matching and ranking strategies can be adopted here, but the discuss of this is out of scope for this paper. When a specific IM is being browsed by \( P \), the \( P \)’s RDFa parser (e.g., this parser has been installed on \( P \)’s Web browser) can automatically parse the IM page into RDF triples. By querying these triples, \( P \) can check if it can play a specific role in this IM. If \( P \) can provide all OKCs which this IM requires a role to provide, \( P \) is able to subscribe to this IM and play this role when the IM is executed. Moreover, \( P \) will also have a chance to query the above harvested triples to find out if this IM has belonged to some communities and then \( P \) can apply for a membership and join these communities if it is happy with that. Suppose the URI for this IM is denoted by \( im\_uri \); the named graph containing \( P \)’s profile is denoted by \( IM\_triples \); the URI for the graph containing harvested triples is denoted by \( IM \). Then following SPARQL query will help the peer figure out which role it can play and which potential community it may join:

```
SELECT ?role ?community
FROM (IM)
FROM NAMED (IM\_triples)
WHERE {
  GRAPH (IM\_triples) {?P a openk:Peer. ?role a openk:Role. ?P openk:can\_play ?role.}
  ?community a openk:Community.
  ⟨im\_uri⟩ openk:has\_role ?role.
  ⟨im\_uri⟩ openk:belongs\_to ?community.
}
```

Another way of making use of republished IMs occurs in their execution processes. In order to analyze this usage, we created and published an IM for the job vacancy service on UK Civil Service Website. This IM can help people not only find out appropriate job information services but also get desired jobs from these services. Since each job vacancy is published on a Web page with embedded micro-data (RDFa for this Website), when the IM is executed, users looking for jobs will query the service and get the job pages back for further checking if they can play roles in this IM. Using the Atom feed provided by this Website, we finally retrieved 338 pages corresponding to 338 jobs. Suppose all pages are attached with a unique ID (0 to 337), the time cost of page retrievals is described in Figure 4. Job vacancy pages from this site are based on a unified form that is actually a template by which the site manager can easily create and update job information. Therefore, the time cost interval for most pages is \([171\ ms, 188\ ms] \). Then the RDFa data are harvested from these page and the corresponding time cost for each harvest is described in Figure 5. In this figure, most of pages are parsed within the time cost interval \([3\ ms, 5\ ms]\). The first page and the last page cost 43\ ms and 27\ ms respectively, which are extraordinarily longer compared with others. This is because much time was spent on the preparation before the loading of retrieved pages and resource disposals after the overall RDFa harvest. We checked extracted triples

---

6http://www.dbpedia.org
7http://lookup.dbpedia.org/api/search.asmx
8http://www.civilservice.gov.uk/jobs/
and found that some triples are duplicated due to the RDFa parse we used. However, the duplicated triples are repeated with equal times for different pages so the duplication will not influence the comparison in our experiment.

By matching the user’s query with the micro-data embedded in republished IMs, the previously ignored IM is discovered as shown in Figure 7. Various peer-side applications can be designed and implemented for digesting and reusing the republished IMs in one way or another but further discussion of this is outside the scope of this paper. Figure 7 depicts a peer-side consumer that analyzes the discovered IM in terms of the local profile and informs the peer of which roles it can play and which communities it may join.

Figure 6 gives a snapshot of a publisher republishing an IM that describes a peer sending a greeting sentence to another peer and the recipient displays this sentence. After being republished, the XHTML page will be published on the DDS and stored in the publisher’s local IM repository. As soon as the publisher receives acknowledgement of the publication from the DDSs, it directly subscribes to the IM.

Related Work

From the perspective of orchestration, several approaches have been proposed for semantically describing WS, such as OWL-S (Martin et al. 2004), WSDL-S (Akkiraju 2005) and SAWSDL (Farrell and Lausen 2007). Consequently, several matchmakers such as OWLS-MX (Klusch, Fries, and Sycara 2006) and SAWSDL-MX (Klusch and Kapahnke 2008) came up as well and have been used for fulfilling tasks related to services. However, not adequate attention has been attracted towards semantically enhancing service descriptions from the perspective of choreography. WS-CDL (Kavantzas et al. 2004) has been proposed for describing WS choreography but it lacks an appropriate support for semantics. Moreover, existing SWS choreography description languages such as WSMO (Lara et al. 2004) are relatively heavyweight to portable devices such as mobile phones and personal digital assistants whose computation capabilities have been hampered by limitations of batteries and memories. Micro-data provides us with a lightweight way of adding semantics to Web content for the purpose of making the content both human-readable and machine-readable. Several solutions for embedding micro-data into Web pages have been proposed. Davis proposed eRDF (embedded RDF) (Davis 2005) in Talis, which can be used with any version of HTML but it restricts itself to the existing HTML attributes and does not support full RDF (e.g., there is no data type and no blank node). Microformat (Suda 2006) also makes use of existing HTML and XHTML tags to convey metadata and other attributes but new formats require new data models. RDFa (Adida et al. 2008) was proposed by
Adida and Birbeck, which not only takes advantage of existing HTML attributes but also invents several new attributes in XHTML for its flexibility and disambiguation. It supports full RDF and can reuse data models created for RDF.

The micro-data-embedded strategy is employed in the publishing process of the IM in this paper. Users can use any vocabularies to annotate and republish IMs but existing pervasive vocabularies are recommended due to their acceptances in the SW community. Moreover, more search engines began to crawl and index micro-data-embedded Web pages, which actually provide us a natural platform for creating discovery services for the OpenKnowledge system.

Conclusions

From the perspective of choreography, this paper proposes a semantically enhanced approach for republishing IMs in the OpenKnowledge peer-to-peer knowledge sharing system. We add a semantic layer to the IM by republishing it using the micro-data-embedded Web page. Republished IMs can assist peers in discovering services and collaborating peers meet their requirements more precisely due to unambiguous URIs of resources. Moreover, IM republication complies with the principles of Linked Data and will further contribute to and benefit from the Web of data. On the other hand, the IM republication provides a more secure and controllable way of transferring knowledge in the distributed environment. RDF search engines as well as conventional search engines like Google and Yahoo support micro-data indexing nowadays, but they use different ranking strategies and finding out a synthesis way of making them index IMs and integrating search results is important for our IM discovery service and will be further researched in the next step.

Acknowledgments

The research was supported by the OpenKnowledge project (FP6-027253). I would like to thank Prof. Ewan Klein for his valuable comments.

References


