Micro-Data Injection

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 of the IM to be republished or published.

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

c. Annotating Module provides a drag-and-drop way for publishers who want to attach IM element data with semantic annotations (see Figure 2).

d. Revising Module helps IM publishers delete, modify and replace existing annotations before IMs are finally republished.

In a peer-to-peer network, peers are autonomous and each of them has both server and client capabilities. From the perspective of choreography, peers collaborate through interactions and use LCC to describe the choreographies inside the peer-to-peer network. Although our paper employs LCC, the specific choice of the process language is not essential to the core arguments of this paper. An example in which a client purchases a product referenced by a product code from a shop using his or her credit card is depicted in LCC as follows:

```xml
<r(client, initial, 1, 1) (SHOP, necessary, 1)>
   <a(client(PC, CC), C)>
       <buy(PC, CC) ⇒ payby(CC)⟩ (shop, C)
       ⟨lookup(CC) then receipt(CC)⟩
   </a(client(PC, CC), C)
</r(client, initial, 1, 1)>
```

### Micro-Data Consumption

Republied 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 most important components for the OpenKnowledge 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 can also collaborate with theirs in terms of how or who it input query. Here, “query” means key words or URLs. Since URLs are hard to remember and some URLs are heterogeneous, users can use URLs discovered by RDF search engines like Sindice or URIs through the DBpedia Lookup service [2]. Republied IMs can be semantically enhanced and be harvested embedded RDFs, the DDS can provide a more precise and more extensible querying service thanks to dereferenciable URIs on the Web of Linked Data.

Another way of consuming embedded RDFs happens when an IM is executed after all required roles are filled by peers. 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 discovery 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 from Edinburgh to San Francisco. In these cases, the ORCA at the side of the service provider will take charge of running the Web pages and by matching the harvested RDF data against the user’s query, the coordinator will find out the information that meets the user’s requirement.

### Experiment

As mentioned above, one way of harmonising republied IMs is to assist the DDS in discovering appropriate IMs that meet the peers’ requirements. Suppose a peer P 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. By querying these triplets harvested by an RDF parser, P can check if it can play a specific role in this IM. Moreover, P will also have a chance to query the above harvested triplets to find out if the IM has belonged to some communities and then P can apply for a membership and join those communities if it is happy with that. Suppose the URI for this IM is denoted by <uri>, the named graph containing P’s profile is denoted by <profile>, the URI for the graph containing harvested triplets 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:

```sparql
PREFIX lcc: <http://lcc.openknowledge.net#>
PREFIX dbpedia: <http://dbpedia.org/resource/>

SELECT * 
FROM IM
WHERE GRAPH <profile> (?P a lcc:Profile) .
  FILTER (?P lcc:hasProfileURI <uri>) .
  FILTER (?P lcc:hasRole <role>) .

  OPTIONAL { GRAPH <profile> (?P a lcc:Community) .
               FILTER (?P lcc:hasCommunity URI <communityURI>)
               FILTER (str(?P) = str(<profile>)) } .
```

Another way of making use of republied IMs occurs in their execution processors. In order to test this usage, we created and published an IM for the job vacancy service on UK Civil Service Website [3]. 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 3. Job vacancy pages from this site are based on a unified format 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 1.7s (ms 180ms). Then the RDFa data are harvested from these page and the corresponding time cost for each harvest is described in Figure 4.

The number of harvested RDF triples for each document is depicted in Figure 5, from which we can see that most of pages contain the same numbers of RDF triples (from 43 triples to 53 triples).

### Conclusions

In this paper, we add a semantic layer to the IM by republishing it using the micro-data-embedded Web page. Republied IMs can assist peers in discovering and collaborating peers meet their requirements more precisely due to unambiguous URIs of resources. Moreover, IM republiation cooperates with the principles of Linked Data and will further contribute to and benefit from the Web of data. On the other hand, the IM republiation provides a more accessible and controllable way of transferring knowledge in the distributed environment.