Maskkot – An Entity-centric Annotation Platform

Armando Stellato[1], Heiko Stoermer[2], Stefano Bortoli[2], Noemi Scarpato[1], Andrea Turbati[1], Paolo Bouquet[2], Maria Teresa Pazienza[1]

[1] ART Research Group, Dept. of Computer Science,
Systems and Production (DISP) University of Rome, Tor Vergata
Via del Politecnico 1, 00133 Rome, Italy
{pazienza, stellato, turbati}@info.uniroma2.it

[2] University of Trento
Department of Engineering and Information Science
Trento, Italy
{bortoli, bouquet, stoermer}@disi.unitn.it

Abstract

The Semantic Web is facing the important challenge to maintain its promise of a real world-wide graph of interconnected resources. Unfortunately, while URIs almost guarantee a direct reference to entities, the relation between the two is not bijective. Many different URI references to same concepts and entities can arise when -- in such a heterogeneous setting as the WWW -- people independently build new ontologies, or populate shared ones with new arbitrarily identified individuals.
The proliferation of URIs is an unwanted, though natural effect strictly bound to the same principles which characterize the Semantic Web; reducing this phenomenon will improve the recall of Semantic Search engines, which could rely on explicit links between heterogeneous information sources.
To address this problem, in this paper we present an integrated environment combining the semantic annotation and ontology building features available in the Semantic Turkey web browser extension, with globally unique identifiers for entities provided by the okkam Entity Name System, thus realizing a valuable resource for preventing diffusion of multiple URIs on the (Semantic) Web.

1. Introduction

Whichever name it will assume, be it Web 3.0, Giant Global Graph, or any other, the Semantic Web will have to face an important challenge to maintain its promise of a real world-wide graph of interconnected resources: their identity and retrieval. If entities are uniquely identified in the Web, then anyone can make statements about them, thus incrementing their intensional description and contributing to their success and retrievability.
Unfortunately, while URIs almost guarantee a direct reference to entities, the relation between the two is not bijective. Many different URI references to same concepts and entities can easily arise when, in such a heterogeneous setting as the WWW, people independently build new ontologies, or populate shared ones with new arbitrarily named individuals.
In this work we present an integrated framework combining the semantic annotation and ontology building features provided by the Semantic Turkey web browser extension, with global, unique identifiers for entities provided by the OKKAM Entity Name System (ENS). The integration has been carried out through the development of a dedicated ENS extension for Semantic Turkey – maskkot -- which extends its ordinary ontology building functionalities with entity search over the okkam service. Through maskkot, users can create, extend and/or populate ontologies with individuals, while maintaining reference to their okkam unique identifiers, giving life to a virtuos cycle in which they may contribute and/or get additional information from the ENS-empowered semantic search engine, and at the same time fighting a proliferation of identifiers for the same entity.

It is a commonly accepted fact that whenever a computer system needs to describe an object (or ``entity'' as we call it), it needs to create some kind of identifier in the system which is then regarded as the placeholder or proxy for this object. This holds true for e.g. database systems, but is of special significance for the Semantic Web, where the notion of the Uniform Resource Identifier (URI) fulfills exactly this task, but has the additional goal of linking information about entities in a distributed but global fashion. In this way, distributed information sources are supposed to become integrateable on the fly, to create links between pieces of information that were previously not linked before, enabling systems to answer queries that were previously impossible to answer.
The Semantic Web in its current state suffers from several weaknesses which we are trying to address in this paper:

- A lack for convenient, user-friendly tools for semantic annotation of Web content. While solutions exist and are described in more detail in section 2, we believe that means for semantic annotation should be given as pervasively as possible, and should be (almost) as easy as creating a bookmark.
- A proliferation of URIs for entities. As we have argued in (Bouquet, Stoermer, & Bazzanella, 2008), to date no scalable and open service is available to make possible and to support a *consistent reuse* of identifiers for entities, and this undermines the practical possibility of a seamless integration of distributed knowledge into a global knowledge space.

The work presented in this paper attempts to contribute to the state of the art in the Semantic Web on several levels: firstly, by providing an intuitive tool for the creation of semantic content; secondly, by making sure that the semantic annotations created are also globally aligned on

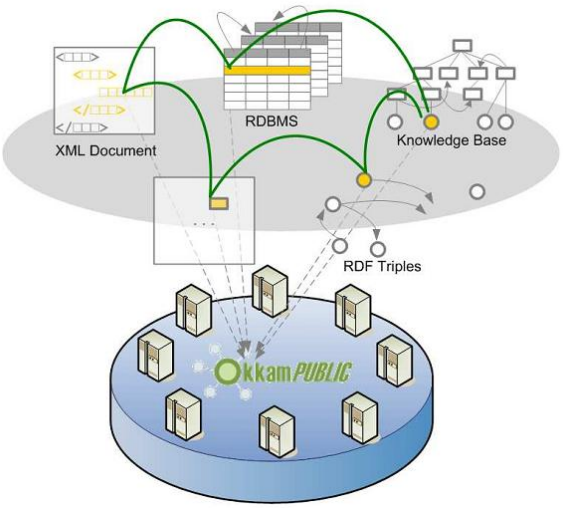

Figure 1: Entities in different information sources and formats, annotated with unique identifiers issued by the okkam ENS

the identifiers for entities, enabling seamless, syntactical integration of data without the need for complex ex-post alignment mechanisms; and thirdly, by contributing to an ever-expanding public space of entity identifiers which offers significant positive network externality effects[1]

2. Related Work

The maskkot integrated platform is rather original in its combination of ontology editing/annotation/semantic browsing functionalities supported by an entity identification service. We therefore report here relevant past works related to the most relevant features characterizing the presented tool.

2.1. Semantic Browsing and Semantic/Social Bookmarking/Annotation

One of the first examples of Semantic Browser can be probably traced back to the Haystack (Quan & Karger, May, 2004) web client. Developed at the MIT laboratories, Haystack was conceived as an application that could be used to browse arbitrary Semantic Web information in much the same fashion as a Web browser can be used to navigate the Web. Standard point-and-click semantics let the user navigate over aggregation of RDF repositories from different arbitrary locations. The application had been built as an extension for the popular Integrated Development Environment Eclipse[2] this choice facilitated extension of the tool thanks to Eclipse flexible plug-in mechanism, but required the user to adopt Eclipse as a platform for browsing the web and collecting data from it: a strong requirement for the user, who would just prefer to rely on his trusted personal web browser and try out other features which are not too invasive for his usual way of working.

Such a less invasive approach was followed by Magpie (Dzbor, Domingue, & Motta, 2003), that was deployed as a plug-in for the Microsoft Internet Explorer Web Browser. Magpie allowed for semantic browsing, and perceived it as a parallel navigational style to complement the "exposed" web content (i.e. free text) by an associated, dynamic semantic layer (which was derived from one or more ontologies semantically describing typical content in a particular domain). Magpie also allows for collaborative semantic web browsing, in that different persons may gather information from the same web resource and exchange it on the basis of a common ontology. Subsequent work on Magpie (Dzbor, Motta, & Domingue, 2004) extended the platform more and more towards the vision of the Semantic Web as "an open web of interoperable applications" (Berners-Lee, Hendler, & Lassila, 2001), by allowing bi-directional exchange of information among users and services, which can be opportunistically located and composed, both manually (web services) or automatically (semantic web services).

From (part of) the same authors of Haystack, comes Piggy-Bank (Huynh, Mazzocchi, & Karger, 2005), an extension for the Firefox web browser that lets Web users extract individual information items from within web pages and save them in RDF, replete with metadata. Piggy Bank then lets users make use of these items right inside the same web browser. These items, collected from different sites, can then be browsed, searched, sorted, and organized, regardless of their origins and types. Piggy-Bank users may also rely on Semantic Bank, a web server application that lets them share the Semantic Web information they have collected, enabling, as for Magpie, collaborative efforts to build sophisticated Semantic Web information repositories from daily navigation through their enhanced web browser.

2.2. Identity and Reference

When attempting to give an identity to ``things" in a way that makes them describable in the Semantic Web (i.e. choosing or creating a URI as a placeholder for them), we can encounter three different approaches:

2.2.1. Local Identification

This is unfortunately -- as mentioned in the introduction -- the common practice at the moment: new URIs for entities are created on the fly, because they are regarded as a mere technical necessity to be able to make RDF statements. Such identifiers do not consider a scope that goes beyond the local knowledge base, and even the Semantic Web community itself has been following this practice, e.g. in the case of authors of Semantic Web conferences as we have shown in (Bouquet, Stoermer, & Bazzanella, 2008).

2.2.2. Vertical Identification

Vertical approaches usually refer to a certain domain of interest, for which an organization is issueing identifiers for entities. Examples include publications (DOI[3]), geographical locations (Geonames[4] or Yahoo! Internet Locations[5]), life science entities(LSID[6]), and many more. The issue with these vertical approaches is the findability of the identifiers: if we are creating RDF statements about entities from many different domains, how do we find out

[1] See (Liebowitz & Margolis, 1998) or http://en.wikipedia.org/wiki/Network_effect for an introduction.
[2] http://www.eclipse.org/
[3] http://www.doi.org
[4] http://www.geonames.org
[5] http://developer.yahoo.com/geo/
[6] http://lsids.sourceforge.net/

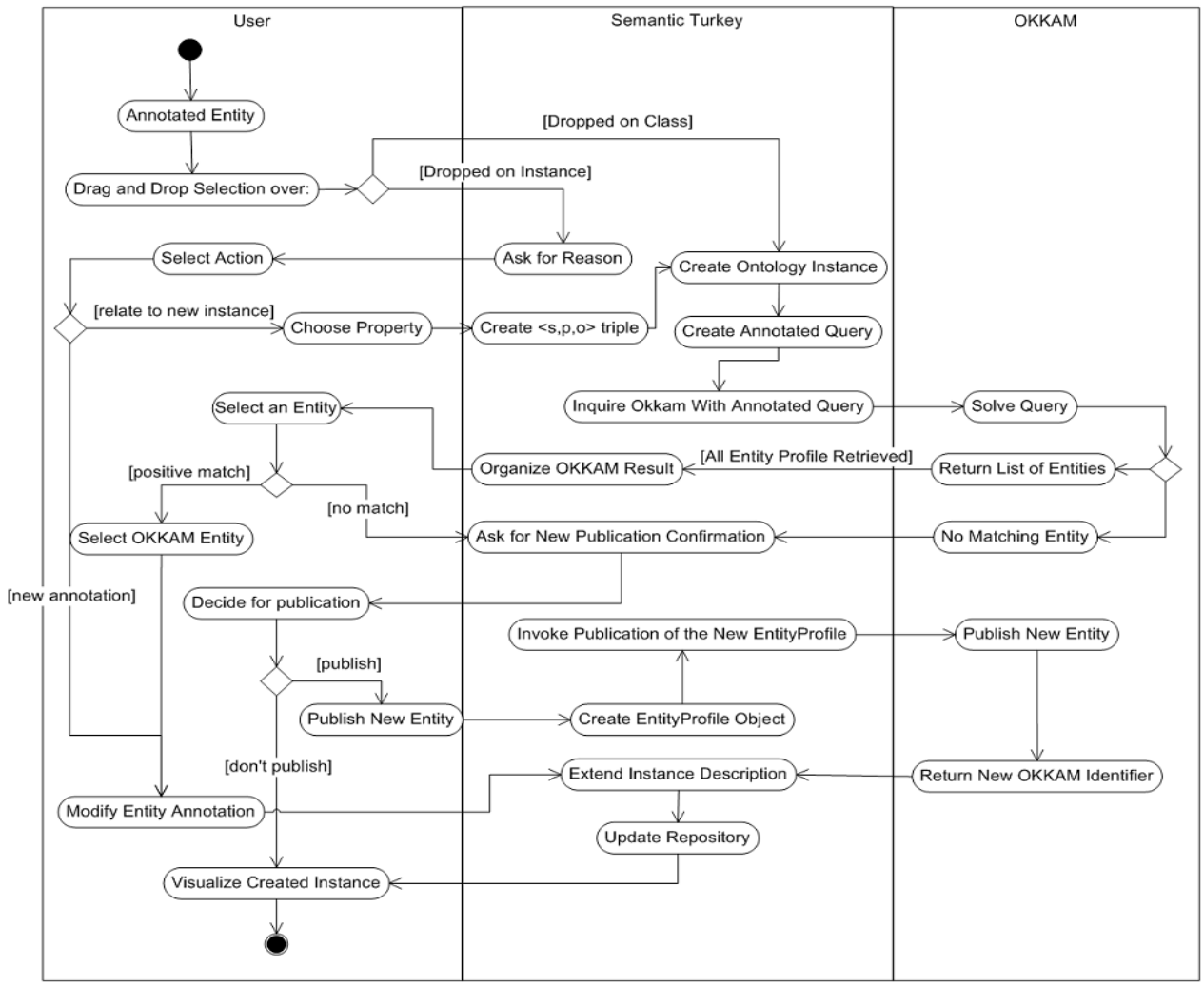

Figure 2: Entity-centric Annotation Activity Diagram

which is the source of an identifier for an entity, and how do we make sure that we chose ``the right'' (i.e. authoritative) one?

2.2.3. Global Identification

Global identification in our sense is a horizontal approach as an attempt to overcome the issues of both the local and the vertical approach. Currently, there are two main streams of activity in the Semantic Web which can be considered relevant in this respect. The first one is the Linking Open Data initiative[7], which pursues an ex-post approach trying basically to discover identity relations between entities that have been given different identifiers in different knowledge bases, but are actually (believed to be) ``the same''. As we have discussed in (Bouquet, Stoermer, Cordioli, & Tummarello, 2008), this approach is viable due to the simple fact that such un-aligned data sources exist, but it has the obvious downside that it does not provide a solution for *avoiding* such a proliferation of identifiers. An orthogonal approach to address all these issues are the efforts around the okkam Entity Name System (ENS), which is described in further detail in section 3.1.

3. Towards an Entity-centric Semantic Annotation Platform

3.1. The Entity Name System (ENS)

The key idea behind the proposal of an ENS is that the Semantic Web can become an open and scalable space for publishing knowledge (in the form of RDF data) only if there will be a reliable (and trustworthy) support for the reuse of URIs. Therefore, at a very general level, the core functionality of the ENS can be characterized as follows: given any representation of an entity (e.g. a bag of keywords, a paragraph of text, a collection of key-value pairs, a graphical depiction, and so on), decide if a URI for this entity is already available in an entity repository (using some method(s) for *entity matching*); if it is, then the ENS will return its URI (or at least a ranked list of candidates), otherwise it will issue a new URI which will be stored in the ENS repository.

As we have argued in (Bouquet, Stoermer, & Bazzanella, An Entity Naming System for the Semantic Web, 2008), issues of entity identification are optimally solved a-priori, across data sources and formats. Instead of creating

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