Empowering Developers' Community Consuming APIs: An Approach for Collaborative API Annotation for Semantically Structured Format

Vidhi Jain
Birla Institute of Technology and Science, Pilani
Department of Computer Science
Vidya Vihar, Pilani Campus,
333031 Rajasthan, India
+91 9928813321
f2014113@pilani.bits-pilani.ac.in

Matthias Frank
FZI Forschungzentrum Informatik
Information Process Engineering
Haid-und-Neu-Straße 10-14,
76131 Karlsruhe, Germany
+49 721 9654858
frank@fzi.de

1. INTRODUCTION

With proliferation of Web APIs, the demand to automatically aggregate, invoke and analyse these services is increasing. Exploring and exploiting the myriad of Web APIs, the developers are required to build heterogeneous data-oriented service compositions, called mashups to power Agile application development [2]. In order to aggregate data from different provenances, the Client applications need to be specifically coded : (i) For each API (ii) By every developer consuming the APIs.

Most of the Web API documentations are human readable only. It is time consuming as each API consumer needs to go through the entire documentation, understand the API and then configure a custom tailored software. Rather a smart generic client is required to automate this process. We focus on answering the following research questions:

RQ1 How to model suitable vocabulary for semantic annotation of API documentation?

RQ2 How to ensure automatic processing of collaboratively annotated API documentations by Smart Clients?

This defines our vision: bridge the gap between the existing Web API usage scenario and the dream to make them machine processable. In Section 2, we discuss the state-of-the-art technology for semantic annotation for the developers. We explain the approach for collaborative semantic annotation in Section 3 . In Section 4, we conclude with our contributions and the avenues for future work.

2. BACKGROUND AND RELATED WORK

While research has been for machine readable API documentations, the onus is mostly on the Designer of the API, for instance, RESTful API Modeling Language (RAML)[1], Apiary[2] and OpenAPI Specification (Swagger)[3]. Unfortunately, most Designers have yet not implemented these solutions to the existing APIs. Bringing a drastic design change is a critical task, especially when their consistent functioning affects a substantial fraction of the economy [5]. This makes it quite difficult to standardise any of the solutions for all the Designers.

For the providers, APIs have to be provided only once. For the consumers, they are called many times over. ASSAM [6] is WSDL annotator tool for API consumers to semantically annotate a Web service. But it requires prerequisite knowledge about ontology classes. In our solution, the complex interplay of semantics is abstracted out from the annotators through Semantic Forms.

SWEET [11] is a lightweight Web application to create semantic descriptions of Web APIs, based upon hRESTS microformat and the MicroWSMO microformat. While this tool involves a lot of manual task as a single user Web Application, we reduce this burden through collaborative annotations and re-use of existing ones in our approach.

3. APPROACH AND UNIQUENESS

We present a collaborative approach for annotation of the Web API Documentation, DocApi[4]. It is an open-source platform, powered by Semantic MediaWiki[5] for the developer community to update, maintain and reuse API descriptions. The developers can annotate in Semantic Forms structured on Hydra vocabulary to create machine processable hypermedia triple relations from existing human readable descriptions. In the following sections, we discuss the approach and significance of each aspect.

3.1 Collaborative API Annotation

Many successful web sites thrive on the wisdom of the crowd [3]. To transform the Web API documentations from unstructured textual pages to processable and queryable format, we empower the developer community to collaboratively annotate or re-use them. All developers, consuming the API, will need the same document since it has to be finally be interpreted by machines (which do it in the same way always).

Through the customized SMW platform, the developers can

[1] http://raml.org/
[2] https://apiary.io/
[3] https://openapis.org/

[4] DocApi https://amazonas.fzi.de/docapi/
[5] SMW https://www.semantic-mediawiki.org

socially collaborate by keeping track of changes, allowing comments and discussion on every single part of API annotations. They can further assign tasks and honor the activity of users.[10]. There has been a successful history for collaborative approach for documentation [1] but it has not been used to annotate publicly available Web APIs yet.

3.2 Role of Semantics

The Semantic Web enables people to create data that machines on the web can exchange and share. It is needed for machines to move between the Web of links and nodes in the same fashion as the human user. The coordination between systems is executed through the use of well described services, that are discovered and selected based on requirements, then orchestrated, adapted or integrated. [4]

A Semantic MediaWiki combines the best of semantics and wiki platform: it lets multiple people, at multiple places, cooperate in making documents and data, along with the semantic programs that retrieve, organize and present this data [8]. The basic structure of semantic data in SMW is inspired by RDF and OWL [7]. The knowledge of semantics is not a pre-requisite for annotation due to abstraction by Semantic Forms. The input to the forms is used to generate RDF Feed for the machines to understand.

3.3 Modelling Vocabulary for Semantic Annotations

The structure of the RDF is designed using Hydra vocabulary as the backbone. It is lightweight vocabulary to describe REST APIs in machine readable affordances which enables generic interaction among machines. This helps to design hypermedia driven Web APIs. We intend that the API consumers are able to use this vocabulary for even the existing APIs.

Through the Figure 1, we explain how we identify the contextual correspondence between the object oriented structure of Hydra to the wiki world of SMW.

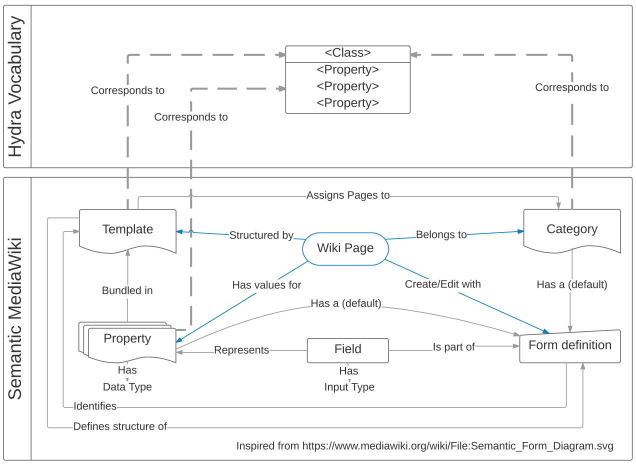

Figure 1: Context of Hydra Vocabulary and SMW explained

In the Figure 2, we show how the Semantic forms are structured in the SMW. Each block represents a Semantic Form, along with its field-names and their data-type. This figure shows how each meta-tag of the Hydra vocabulary [9] can be annotated through the Semantic Forms in Semantic MediaWiki.

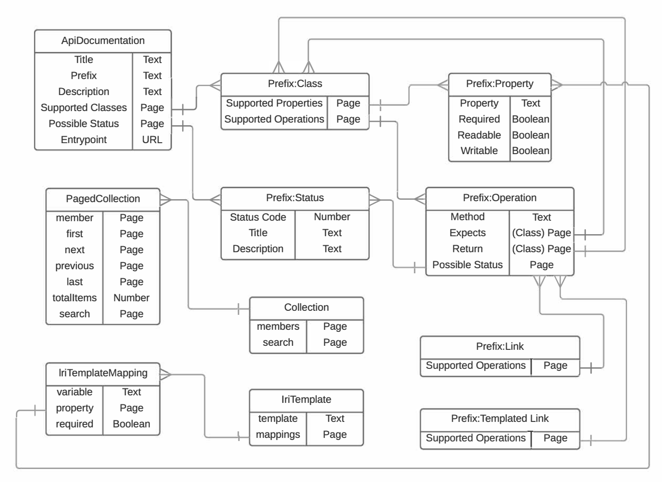

Figure 2: Modelling Hydra Vocabulary into Semantic MediaWiki

4. RESULTS AND CONTRIBUTIONS

The proposal is a stepping stone to bridge the gap between the dream of Web 3.0, about machine to machine communication, and the way APIs are being used today. The RDF data generated after annotation can be validated and visualised by W3C Validator[6]. Our use-case is to integrate the provenance information from different APIs in an automated way to feed into Decision Support System for project BigGIS[7].

4.1 Contributions

The solution takes us towards a more standard way of API documentation so that they are both human and machine processable. The contributions of my work are:

- Analysis of existing annotation tools for the developers consuming the APIs
- Contextual Modelling of Hydra vocabulary in Semantic MediaWiki Forms for annotation.

4.2 Future Work

The open-ended research extensions of our work are:

- Support for enhanced extensions, discussion or talk pages, advance search for better API usage on customised SMW.
- Design of Smart Clients for automatic processing of the API description.

[6]W3C Validator `https://www.w3.org/RDF/Validator/`
[7]BigGIS `https://www.fzi.de/forschung/projekt-details/biggis/`

Through meta-tagging API descriptions, smarter generic clients and interoperability of Web APIs can be achieved. The machine processable and queryable API descriptions will help the developers to build and test generic Client architecture. It can be used to aggregate the data from several API provenances and feed to a dynamic decision support system. Thus, complex integration tasks can be pipe-lined with minimum human interference as machines will understand the valid state transitions possible. This will eventually improve service composition as minimal implementation logic will be required.

5. REFERENCES

[1] S. Auer, S. Dietzold, J. Lehmann, and T. Riechert. OntoWiki: A Tool for Social, Semantic Collaboration of Structured Knowledge. In *Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge (CKC 2007) at the 16th International World Wide Web Conference (WWW2007) Banff, Canada*, volume 273 of *CEUR Workshop Proceedings*. CEUR-WS, 2007.

[2] D. Bianchini, V. D. Antonellis, and M. Melchiori. A Multi-perspective Framework for Web API Search in Enterprise Mashup Design. In *Proceedings of the 25th International Conference on Advanced Information Systems Engineering*, CAiSE'13, pages 353–368, Berlin, Heidelberg, 2013. Springer-Verlag.

[3] G. Correndo and H. Alani. Survey of tools for collaborative knowledge construction and sharing. In *Proceedings of the 2007 International Conference on Web Intelligence and Intelligent Agent Technology - Workshops, Silicon Valley, CA, USA*, pages 7–10. IEEE Computer Society, 2007.

[4] D. Fensel, M. Kerrigan, and M. Zaremba. *Implementing semantic web services: The SESA framework*. Springer, Berlin, 2008.

[5] M. Henning. API design matters. *Commun. ACM*, 52(5):46–56, 2009.

[6] A. Heß, E. Johnston, and N. Kushmerick. ASSAM: A Tool for Semi-automatically Annotating Semantic Web Services. In *The Semantic Web - ISWC 2004: Third International Semantic Web Conference,Hiroshima,Japan, November 7-11, 2004. Proceedings*, pages 320–334, 2004.

[7] M. Krötzsch and D. Vrandecic. Semantic Wikipedia. In A. Blumauer and T. Pellegrini, editors, *Social Semantic Web: Web 2.0 - Was nun?*, X.media.press, pages 393–421. Springer, 2009.

[8] M. Krötzsch, D. Vrandecic, and M. Völkel. Semantic MediaWiki. In *SEMWEB*, 2006.

[9] M. Lanthaler and C. Guetl. Hydra: A Vocabulary for Hypermedia-Driven Web APIs . In *Proceedings of the WWW2013 Workshop on Linked Data on the Web, Rio de Janeiro, Brazil, May*, volume 996 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2013.

[10] R. O. Lloyd Rutledge, editor. *Applying and Extending Semantic Wikis for Semantic Web Courses,*, 2011.

[11] M. Maleshkova, C. Pedrinaci, and J. Domingue. Semantic Annotation of Web APIs with SWEET. In G. A. Grimnes, S. Auer, and G. T. Williams, editors, *Proceedings of the Sixth Workshop on Scripting and Development for the Semantic Web, Crete, Greece*, volume 699 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2011.