A NEW CONCEPT OF THE SEARCH ENGINE FOR THE WEB API

Yuka Obu[1], Minoru Sasaki[2] and Tatsuhiro Yonekura[2]
[1]*Graduate School of Science and Engeneering, Ibaraki University, Ibaraki, Japan*

[2]*Department of Computer and Information Sciences, Ibaraki University, Ibaraki, Japan*

Keywords: Web service, Web API, Software as a Service, Standardization of the Web.

Abstract: The World Wide Web has become a platform for many kinds of Web services. Millions of Web users have interests on the Web services, and there are still a lot of possibilities in it. We have focused on the capability of the Web, and been exploring the new demands and services on the World Wide Web. In this paper, from the viewpoint of the standardization of the Web API and the Web service that searches the best Web API for user's demands, we propose a framework of information that is required for the standardization of the Web API, and finally propose the Web service that searches Web APIs for *mash-up* to create a new Web service.

1 INTRODUCTION

The World Wide Web (Web hereafter) has become a platform for various kinds of Web services. Over the latest years, many applications – map viewer, calendar, e-mail, travel reservation, catalogue, shopping, game, and so on - have been developed on the Web. Accordingly, huge amount of Web users have great interest on the Web service, and there are still a lot of possibilities in it. We have focused on the capability of the Web, and been exploring the new demands and services on the Web.

Nowadays, users have a demand to use the Web service as they want, and that demand is leading to the development of new functions and new services. Software as a Service, which is referred as SaaS, and the Web API are examples of new technology for the Web services. SaaS is a software that user can use via the Internet. One of the advantages of SaaS is that users do not need to install any software to use the function of them and they can use only the function they need. Web API is open to the public, and users can create their own Web service or Web site as they prefer by using the APIs. Especially, the Web service which is created by utilyzing more than one API is called mash-up site. User can also construct various types of SaaS as mash-up sites by using the Web API. Therefore, if there is no SaaS that meets user's demand, user can create personal one. Though the Web API can meet the users' demands flexibly, because it is constructed by using JavaScript or XML, it is difficult for the non-technical oriented users to operate the Web API. Moreover, because the Web API is not standardized in terms of the protocol, even the technical oriented users need some time to create new Web sites by using the Web API.

There is another problem to use the Web API. That is, how does user know which is the best API for creating a new Web service? There are many APIs on the Web. The output to the input is slightly different according to the API. The grammar of the API is not standardized. Therefore, to select which API to use comes to introduce costs to the user.

In this paper, from the viewpoint of the standardization of the Web API and the Web service that searches the best Web API, we propose a framework of information that is required for the standardization of the Web API. And we also propose the concept of the Web service that searches Web API and creates a new Web service using it.

2 RELATED RESEARCHES

There are some kinds of the Web service of creating new Web site by using the Web API. For example, Yahoo Pipes (Yahoo, 2007), PopFly (Microsoft, 2007), Google gears (Google, 2007) are new Web services that user can use the Web API on them. Google gears is an open source browser extension that enables web applications to provide offline

Obu Y., Sasaki M. and Yonekura T. (2008).
A NEW CONCEPT OF THE SEARCH ENGINE FOR THE WEB API.
In *Proceedings of the Fourth International Conference on Web Information Systems and Technologies*, pages 343-346
DOI: 10.5220/0001531503430346

functionality using the JavaScript APIs. By using this API, user can create new applications that can store and serve application resources locally. Yahoo Pipes and Microsoft PopFly are composition tools to aggregate, manipulate, and mash-up content from around the web. Some commands can be combined together to create output that meets users' demands. One of the features of them is that the user can compose the new web site by using GUI based operation. These examples are new technology on the Web. But in order to use them, user must have somewhat of knowledge about JavaScript and Web APIs. In other words, if a non-technical user wants to use these services, it takes a while to learn the usage of them.

For non-technical oriented users, visual programming tools that allow user to create programming code easily have been developed. Islay, an interactive animation-authoring tool based on the state-transition diagram, was first proposed in 2005 (Okamoto, Kamada, Nakao, 2005) (Okamoto, Shimomura, Kamada, Yonekura, 2006). Islay uses the state-transition diagram to make authoring interactive animation intuitively comprehensible. By using the authoring tool, non-programmers can define the animation of the characters. As a side effect, the user may learn how to define dynamic objects while having fun with animation. The GUEST (Graphical User interface Editor by State diagram) (Obu, Yamamoto, Yonekura, 2007) is a Firefox (Mozilla org. 2007) extension that enables users to define the behavior of the browser using a state-transition diagram. Using the GUEST, the user can define behaviors easily even if he or she has no programming experience.

Web API Search (Web API Search, 2007) and *API Compare-and-Matching Service* (Metadata, 2007) is a search engine for the Web API. On the *Web API Search*, user inputs some keywords, and then retrieval results from the database are displayed. *API Compare-and-Matching Service* provides functions of the Web API search related to the key word and the Web API. It is possible to review the Web APIs by leaving history of inspected API. By using this search engine, user can create an outline of the planning document of the mash-up Web service. These services are helpful to find a suitable Web API. But the result is a kind of catalogue. Therefore the user has to understand the functions of the Web API and how to use it in order to create new Web site or new mash-up site. Moreover, if the user needs some Web APIs, user must consider about the relationship between the APIs.

3 NEW CONCEPT OF SEARCH ENGINE FOR WEB API

3.1 New Search Engine for the Web API

The current search engines for the Web API mentioned above are a kind of searching the catalogue of the Web APIs. The users input the keyword according to input, output, or function that they need. And they needs to have technical knowledge and comprehend the usage and functions of the API. Accordingly, we propose a new concept of a search engine that has more high performance.

Our aim is that user can search the Web API without considering about the details regarding functions and codings. Our new concept takes only one step to find the best Web API, while the current system needs user to compare the APIs.

In our new concept, the search engine has to be more semantically. Hence, some data for the system are defined as below.

(1) **Input Data**: Data that user input to the Web API. For example, Japanese, English, and so on.

(2) **Inputdata Type**: Data type of input. For example, sentence, a word, and so on.

(3) **Process**: The function of the Web API. For example, translating sentences, looking up the words from a dictinary, and so on.

(4) **Output Data**: Data that the user needs.For example, English, German, and so on.

(5) **Output Data Type**: Data type of output. For example, sentence, words, and so on.

By using these data above, the system can find the best Web API. Let's assume that we create the translation service from Japanese sentence to German sentence. Input data, input data type, process, output, and output data type are Japanese, sentence, translation, Germany, and sentence, respectively. Then the system show results the translation Web API. Moreover, if there is no Web API to translate Japanese sentence to German sentence, we might employ the two possible APIs: the one that translates sentence from Japanese to English, and the other that translates English to German. Then the system searches these two steps from input data listed above (Figure 1).

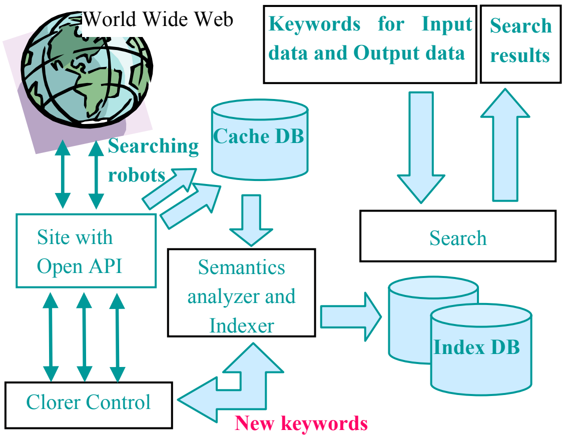


Figure 1: Outline of the search engine for the Web API.

Figure 1 is an outline of the search engine for the Web API. Semantics analyzer and indexer create the index database (Index DB). Indexer extracts keywords from the Web API, and input data, API, and keyword are registered into the index DB as an inex. In the index DB, some attribute is also registred. This attribute indicates what kind of input data is used for what kind of Web API.

3.2 Standardization of the Web API

In order to realize the system which is mentioned above, the standardization of the Web API is necessary in terms of their protocol. Semantics analyzer in Figure 1 may handle to decide types of data that the API uses, and data format to input or output to be defined. And it needs the format that represents how the API processes from the input to the output.

The standardization may lead the Web APIs to mash-up each other more flexibly, and it becomes benefit for users and developper of the Web API. And it will be the great forward push of the new World Wide Web.

3.3 An Example of the Web Service to Create Mash-up Site with Web API

We are designing a new Web service that searches the Web APIs and creates a new mash-up site. Figure 2 shows an outline of the service. The flow of processing of the service is as follows.

(1) User inputs data that is needed for the new mash-up site.

(2) The search engine searches the Web API.

(3) The result is shown as a state-transition diagram.

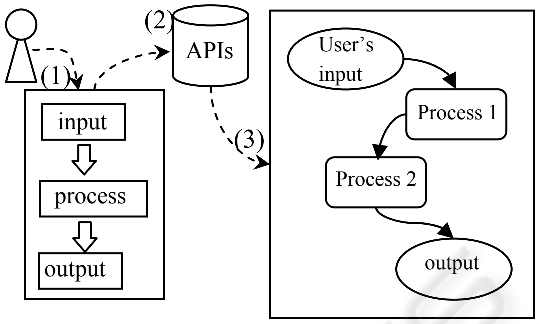
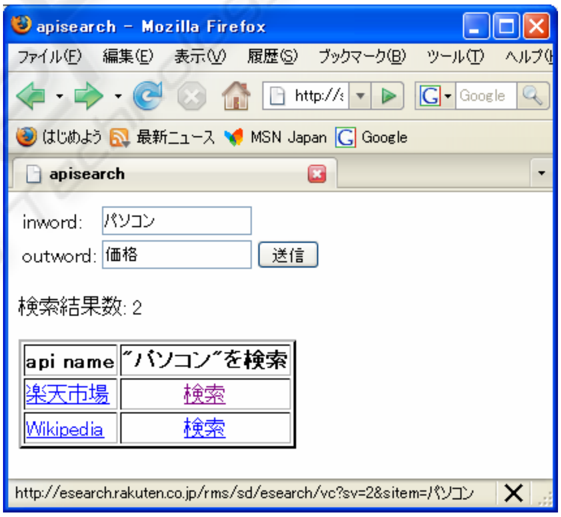


Figure 2: Concept of the service.

This service shows the result by using a state-transition diagram. By using a state-transition diagram, user can comprehend the Web API intuitively, and it is easy to understand the flow of processing and change of the data. When the user arranges the process, he re-edits the state-transition diagram instead of the source code.


apisearch - Mozilla Firefox
ファイル(F) 編集(E) 表示(V) 履歴(S) ブックマーク(B) ツール(T) ヘルプ(H)
http://
Google
はじめよう 最新ニュース MSN Japan Google
apisearch
送信

inword: パソコン

outword: 価格

検索結果数: 2

api name	″パソコン″を検索
楽天市場	検索
Wikipedia	検索

http://esearch.rakuten.co.jp/rms/sd/esearch/vc?sv=2&sitem=パソコン

Figure 3: Screen shot of the current search engine.

Figure 3 is a screen shot of the current search engine of the Web API. In this search engine, there are two text-input edit boxes; these are for input data, and output data of the Web API. When these two data are input and the button is clicked, the Web API search is executed. As a result of the search, some Web APIs that can be suited for the input and output data are listed.

As an example, shown in figure 3, input data is PC, and output data is price. Then the Web API names as results are Rakuten Ichiba (Rakuten, 2007) and Wikipedia (Wikipedia, 2007). This means if the user wants to search price by using PC as an input data, the user can use the Web API of the Rakuten Ichiba, or that of Wikipedia.

4 CONCLUSIONS

We proposed a new concept of the search engine and the standardization of the Web API. And an example of the Web service to create mash-up site with the Web API is proposed. By using our new concept, more semantic search engine may be realized, and many users may be able to find the best Web API they want and create a new mash-up site easily. This will lead the Web APIs to mash-up each other more flexibly, and it becomes benefit for users and developper of the Web API and will be the great forward push of the new World Wide Web.

ACKNOWLEDGEMENTS

We would like to appreciate all of our colleagues participating in this project, namely, Professor Masaru Kamada of Ibaraki Univ. and Assoc. Professor Shusuke Okamoto of Seikei Univ. and Mizuaki Yamamoto of Hitachi Co.Ltd. We give our thanks to Mr. Osamu Miyamoto for his Funds. This study was partially supported by the JSPS Grants-in-Aids No. 18300027.

REFERENCES

Yahoo, 2007. Yahoo Pipes. In *http://pipes.yahoo.com/pipes/*

Microsoft, 2007. PopFly. In *http://www.popfly.ms/*

Google, 2007. Google gears. In *http://gears.google.com/*

Okamoto, S., Kamada, M., Nakao, T., 2005. Proposal of an Interactive Animation Authoring Tool Based on State-transition diagram. *Vo.46, No.SIG 1(PRO24), pp19-27*. Information Processing Society of Japan. in Japanese.

Okamoto, S., Shimomura, T., Kamada, M., Yonekura, T., May 2006. Programming with Islay, an Interactive Animation Authoring Tool. *Vo.47, No.SIG 6(PRO29)*. Information Processing Society of Japan. in Japanese.

Obu, Y., Yamamoto, M., Yonekura, T., Kamada, M., Okamoto, S., 2007. Exploring State-transition diagram-based Web Browser Programming. In *CYBERWORLDS 2007, International Conference on Cyberworlds*. Cyberwrolds Press.

Mozilla org., 2007. Mozilla Firefox. In *http://www.mozilla.or.jo/products/firefox/*

Web API Search, 2007. Web API Search. In *http://kommy.s254.xrea.com/WebAPI_Search.php*

Metadata Incorporated, 2007. API Compare-and-Matching Service. In *http://www.api-match.com/*

Rakuten Ichiba, 2007, In *http://www.rakuten.co.jp/*

Wikipedia, 2007, In http://ja.wikipedia.org/