RSS CONVERSION SERVICE FOR INTEGRATING WEB INFORMATION RESOURCES AND HOME NETWORK SYSTEM

Hiroshi Igaki, Hiroyuki Sakamoto and Masahide Nakamura
Department of Computer Science and Systems Engineering, Graduate School of Engineering
Kobe University, 1-1 Rokkodai-cho, Nada-ku, Kobe 657-8501, Japan

Keywords: Home Network System, RSS, Mashup, Web Service.

Abstract: Home Network System(HNS), which connects home appliances and sensors to networks, has been researched actively. HNS makes it possible to remote-control and monitor home appliances. However, use of these services has been limited to home appliances and sensors in HNS.In this paper, we present integrated services combining RSS documents and HNS with our RSS conversion service. In our service, the hyperlink in each item on an RSS document is replaced to a WebAPI of home appliance control. This converted RSS(integrated RSS) can control home appliances by clicking items in an RSS reader.

1 INTRODUCTION

Ubiquitous and embedded technologies are emerging widely as new fundamental technological frameworks for a home network system. Based on the technologies, many smart devices including sensors and home appliances in home network system have network capability to access their functions from other appliances or softwares. With using such devices, various applications have been being developed. One of the applications is an appliance integration service. The integration service provides a user with new added value by combination of multiple devices in HNS. An automatic door(Nabtesco Corp., 2008) and a sensor light(Residential Landscape Lighting & Design, 2008) with a human-detect sensor are typical examples with using multiple devices.

As Web 2.0 technology spreads, mashup services with higher added value combining the various information resources exhibited on the Web have become popular. Then, we aim at development of more sophisticated integration services combining with not only home appliances but the information resources on the Web as a next step of HNS integration services. For example, ”stock price information service” which tells a user a certain stock price fluctuation by combining the alarm device and stock information, and ”energy-saving air-conditioner, ventilation service” which performs optimal opening and closing of windows, ventilating fans, and air-conditioner control based on the weather information on the Web. To integrate home network system and Web information resources, we use RSS(Berkman Center for Internet & Society at Harvard Law School, 2007) documents and HNS-WebAPIs.

In our research, we focus on a `<link>` element in the RSS document. The link element usually has a hyperlink to access detailed information of each `<item>` in the document. By replacing the hyperlink by WebAPI of home appliance control(HNS-WebAPI), new integrated services using both web information resources and home network system can be developed.

In our proposed *RSS conversion service*, based on a certain pre-defined conversion rule, replacement of the hyperlink of adequate items in the RSS document is performed. The conversion rule contains a HNS-WebAPI and an item extractor to extract adequate items in the RSS document. Based on the rule, our service specifies items in RSS documents, and replaces a hyperlink of the link element into a HNS-WebAPI.

In this paper, we explain about our RSS conversion service based on the conversion rule.

Igaki H., Sakamoto H. and Nakamura M. (2009).
RSS CONVERSION SERVICE FOR INTEGRATING WEB INFORMATION RESOURCES AND HOME NETWORK SYSTEM.
In *Proceedings of the International Conference on e-Business*, pages 131-134
DOI: 10.5220/0002194001310134

2 PRELIMINARY

2.1 Home Network System

A Home Network System(in short, HNS) consists of one or more *networked appliances* connected to a LAN at home. Each networked appliance has a set of *control APIs*, so that the user or external software agents can control the appliance via the network (e.g., (DLNA, 2007)(ECHONET Consortium, 2004)).

In this paper, we assume that such control APIs are exhibited as WebAPIs. The term ”WebAPI” here indicates an API that can be accessed over HTTP. (Nakamura et al., 2008) shows examples of WebAPIs which are exposed by a legacy appliance wrapped in a Web Service . Hereafter we call such WebAPIs in HNS *HNS-WebAPIs*.

2.2 RSS as Web Information Resources

RSS is a Web content syndication format(Berkman Center for Internet & Society at Harvard Law School, 2007). Adopting RSS to the information resources on the Web ,such as blog entries, news headlines, weather information and TV programs enables users to access the resources in a standardized way.

In a typical format of an RSS document, <channel> element indicates outline of information resources, and the <item> expresses each information item itself. For example, in the case of an RSS document containing blog entries, name, URL, and summary of the blog are described in the <channel>, and each <item> indicates each entry. Then, each <item> contains a <title> element, summary(a <description> element), a hyperlink to the detailed information (a <link> element), and date of the entry etc.. A user can browse various kinds of RSS documents on the web through an RSS reader(News On Feeds, 2008). Similarly, many applications can produce RSS documents to mashup with other resources or web services.

3 INTEGRATION INFORMATION RESOURCES AND HOME NETWORK SYSTEM

Recently, very various and worthy information resources such as news headline, weather forecasts, stock quotations, TV program listings, and Podcasting, etc. are exhibited as RSS documents. We suppose novel integrated services are realized by combining these information resources and home appliances. In this paper, we assume the following services as new integrated services combining information resources and home appliances.

- RSS-TV Service

  If a user selects a certain TV program under televising from a TV program listing RSS through an RSS reader, the program can be watched by the user’s TV in HNS.

- Podcasting Play Service

  In this service, a MP3 Stereo Component plays a podcasted item selected by the user on RSS reader.

In the following section, we introduce about our RSS Conversion Service to realize these kinds of service.

4 RSS CONVERSION SERVICE

We focused on the <link> element which is sub element of the <item> in the RSS document in order to integrate an RSS document and HNS. Originally, the <link> element includes a hyperlink to the detailed information of each information item. A user can read a title, summary, and published date etc. of the items in the RSS documents. If the user wants to acquire more detailed information, he/she moves to URL of the hyperlink. Therefore, the general RSS reader offers a GUI which a user can travel to the hyperlinked information by clicking on the title of the item.

Then, we propose *an RSS conversion service* which changes the hyperlink in the <link> element into arbitrary HNS-WebAPI like Figure 1. Our service enables users to invoke a suitable HNS-WebAPI by clicking title of a specific information item. In order to make an RSS document and an HNS-WebAPI cooperate appropriately, our service uses *a conversion rule* which contains search conditions for extraction

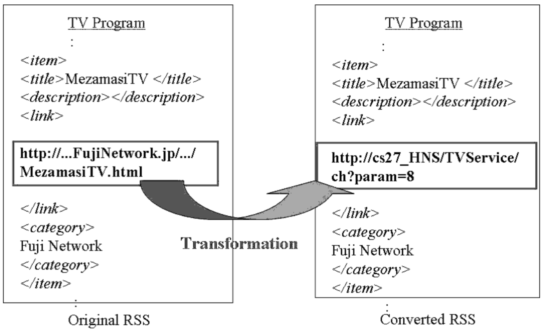


Figure 1: RSS Conversion Service.

Table 1: Example of Conversion Rules.

ConversionRule1		
	TargetElement:	category
	Search Query:	ABC-TV
	HNS-WebAPI:	http://myHNS/TVService/ch?param0=6
ConversionRule2		
	TargetElement:	category
	Search Query:	Fuji Network
	HNS-WebAPI:	http://myHNS/TVService/ch?param0=10

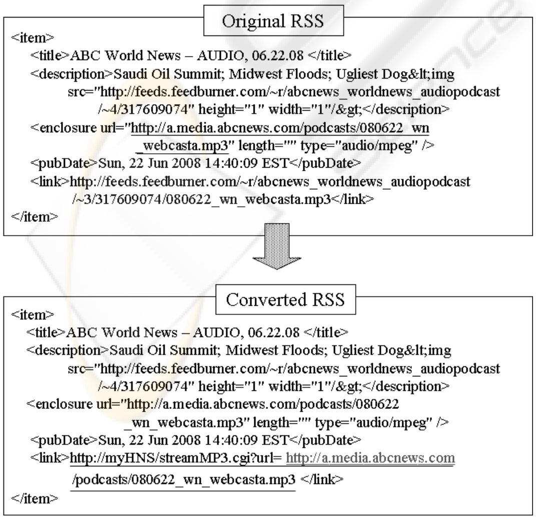
of suitable items in an RSS document and an adequate HNS-WebAPI. In the following section, we denote about the conversion rule for our service.

4.1 Conversion Rules

We define a conversion rule as a combination of *an item extractor* extracting adequate items and a HNS-WebAPI. The item extractor consists of *a target element* and *a search query*. The target element shows arbitrary element name included in each `<item>` such as `<title>`, `<description>`, etc. in an RSS document. The search query using regular expressions is given to find adequate items based on the character string in the target element.

If this item extractor and an HNS-WebAPI as a conversion rule are given, `<link>` which is subelement of each `<item>` in an RSS document is replaced to a suitable HNS-WebAPI based on the rule. A user can invoke the HNS-WebAPI by clicking a title of the item on the RSS reader.

Table 1 shows an example of a conversion rule for the RSS-TV Service. The TV program listing RSS to which the conversion rule is applied includes a TV station name in the `<category>` of each item. In the conversion rule, a target element ”category” and a search query ”ABC-TV” are specified as an item extractor. Then, a HNS-WebAPI ”http://myhome/TVService/ch?param0=6” which sets channel of a TV to 6ch is given in the rule. By preparing conversion rules with the similar scheme on other TV programs, every TV programs and TV WebAPIs which set channel of the TV are associated.



Original RSS

```
<item>
   <title>ABC World News – AUDIO, 06.22.08 </title>
   <description>Saudi Oil Summit; Midwest Floods; Ugliest Dog&lt;img
           src="http://feeds.feedburner.com/~r/abcnews_worldnews_audiopodcast
                 /~4/317609074" height="1" width="1"/&gt;</description>
   <enclosure url="http://a.media.abcnews.com/podcasts/080622_wn
                          _webcasta.mp3" length="" type="audio/mpeg" />
   <pubDate>Sun, 22 Jun 2008 14:40:09 EST</pubDate>
   <link>http://feeds.feedburner.com/~r/abcnews_worldnews_audiopodcast
              /~3/317609074/080622_wn_webcasta.mp3</link>
</item>
```

Converted RSS

```
<item>
  <title>ABC World News – AUDIO, 06.22.08 </title>
  <description>Saudi Oil Summit; Midwest Floods; Ugliest Dog&lt;img
           src="http://feeds.feedburner.com/~r/abcnews_worldnews_audiopodcast
                 /~4/317609074" height="1" width="1"/&gt;</description>
  <enclosure url="http://a.media.abcnews.com/podcasts/080622
                        _wn_webcasta.mp3" length="" type="audio/mpeg" />
   <pubDate>Sun, 22 Jun 2008 14:40:09 EST</pubDate>
   <link>http://myHNS/streamMP3.cgi?url= http://a.media.abcnews.com
              /podcasts/080622_wn_webcasta.mp3 </link>
</item>
```

Figure 2: Conversion of Podcasting-RSS in Podcasting Play Service.

Table 2: A Conversion Rule for Podcasting Play Service.

ConversionRule		
	TargetElement:	enclosure@type
	Search Query:	audio/mpeg
	HNS-WebAPI:	http://myHNS/streamMP3.cgi?url= *enclosure@url*

4.2 Tag Variable for Dynamic HNS-WebAPI

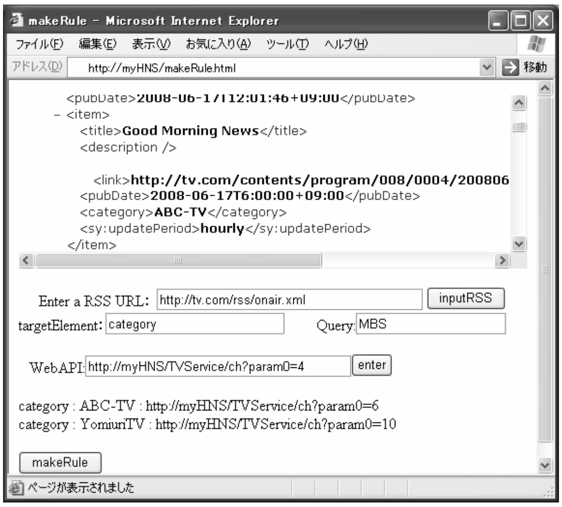
We define *a tag variable* as a variable whose value can be specified with any element value or attribute in `<item>`. The tag variable is used for expressing a flexible HNS-WebAPI in a conversion rule. When our RSS Conversion service replaces data of the `<link>`, the tag variable is replaced with data of specified element based on a specified RSS document.

For example, a Podcasting RSS contains a hyperlink to an MP3 file in ”url” attribute of an `<enclosure>` element in each item. A MP3 Stereo component WebAPI takes url of mp3 file as an argument. In the conversion rule of Podcasting Play Service(denoted in 3), if argument of the HNS-WebAPI is given statically, only pre-given MP3 file can be played by the HNS-WebAPI. To integrate such the RSS document and the HNS-WebAPI flexibly, url of an MP3 file must be given to the MP3 Stereo component WebAPI dynamically.

Table 2 is a conversion rule for the Podcasting Play Service. In this rule, the HNS-WebAPI has a tag variable ”$enclosure@url$”. Figure 2 shows one of the `<item>` elements of a podcasting RSS. Like this figure, the tag variable ”$enclosure@url$” is replaced to the url attribute of the enclosure element in the `<item>`.

5 DISCUSSION

Figure 3 shows our registration UI for creating conversion rules. This UI program consists of



makeRule - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
アドレス(D)
移動
inputRSS
enter
makeRule
ページが表示されました

http://myHNS/makeRule.html

```
    <pubDate>2008-06-17T12:01:46+09:00</pubDate>
  - <item>
      <title>Good Morning News</title>
      <description />

        <link>http://tv.com/contents/program/008/0004/200806
      <pubDate>2008-06-17T6:00:00+09:00</pubDate>
      <category>ABC-TV</category>
      <sy:updatePeriod>hourly</sy:updatePeriod>
    </item>
```

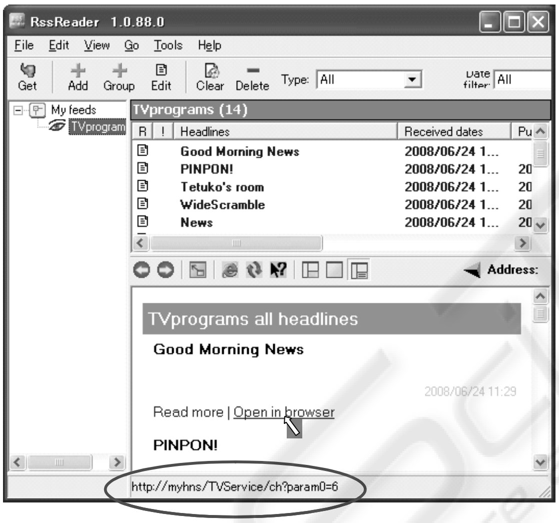
Enter a RSS URL: http://tv.com/rss/onair.xml

targetElement: category   Query: MBS

WebAPI: http://myHNS/TVService/ch?param0=4

category : ABC-TV : http://myHNS/TVService/ch?param0=6
category : YomiuriTV : http://myHNS/TVService/ch?param0=10

Figure 3: UI for Conversion Rule Registration.



TVprograms (14)

R	!	Headlines	Received dates	Pu
🗎		Good Morning News	2008/06/24 1...	
🗎		PINPON!	2008/06/24 1...	20
🗎		Tetuko's room	2008/06/24 1...	20
🗎		WideScramble	2008/06/24 1...	20
🗎		News	2008/06/24 1...	20

TVprograms all headlines

Good Morning News

2008/06/24 11:29

PINPON!

http://myhns/TVService/ch?param0=6

Figure 4: Integrated Services with an RSS reader.

makeRule.html(HTML with JavaScript, LOC:97) and makeRule.cgi(Perl/CGI, LOC:53).

First, the user enters url of an RSS document in this registration UI. The UI displays the contents of inputted RSS document on the upper part of the screen. The user inputs a target element, a search query, and a HNS-WebAPI based on the displayed RSS document. After inputting all conversion rules, a definition file(*rule.txt*) which contains all rules is created by clicking the [makeRule] button.

In the next step, our RSSConverter.cgi(Perl/CGI, LOC:57) applies the definition file containing rules to a specified RSS document. The following url is an example of url for getting converted RSS document. Practically, *RSS_URL* is an actual url of an original RSS document. http://myhome/RSSConverter.cgi?rss=RSS_URL&rule=rule.txt

RSSConverter.cgi applies *rule.txt* which contains conversion rules to the RSS document specified by *RSS_URL*, and creates a new converted RSS document. Tag variables given to the HNS-WebAPI in the rules are also converted, simultaneously. As shown in the Figure 4, a user can browse the converted RSS document, and execute the HNS-WebAPI specified in the conversion rules by clicking each `<item>` element.

Recently, RSS processing services like Yahoo Pipes which can aggregate multiple RSS documents and extract required items are being widely developed. Therefore, information resources to which our RSS conversion service can apply will increase further.

ACKNOWLEDGEMENTS

This research was partially supported by: the Japan Ministry of Education, Culture,Sports,Science and Technology, Grant-in-Aid for Young Scientists (B) (No.21700077, No.20700027).

REFERENCES

Berkman Center for Internet & Society at Harvard Law School (2007). Rss2.0., http://blogs.law.harvard.edu/tech/rss.

DLNA (2007). Digital Living Network Alliance. http://www.dlna.org.

ECHONET Consortium (1997-2004). http://www.echonet.gr.jp/ english/index.htm.

Nabtesco Corp. (2008). Automatic entrance system. http://nabco.nabtesco.com/english/door_index.asp.

Nakamura, M., Tanaka, A., Igaki, H., Tamada, H., and ichi Matsumoto, K. (2008). Constructing home network systems and integrated services using legacy home appliances and web services. *International Journal of Web Services Research*, 5:82–98.

News On Feeds (2008). List of news aggregators. http://www.newsonfeeds.com/faq/aggregators.

Residential Landscape Lighting & Design (2008). Home automation! including product, controllers & systems. http://www.residential-landscape-lighting-design.com/home_lighting_automation.htm.