Device Description Repository System Based on DDR Simple API

Yong-Soon Cho, Young-Il Lee, Hoe-Kyung Jung, *Member, KIMICS*

Abstract— Recently, improved capabilities of the mobile device is represented in a demand for same level services with the desktop device services but the service that is developed for desktop device is not compatible with mobile devices. To fulfill these demands, it is needed to provide services with considering features of mobile devices. This means that CP(Contents Provider) must do contents transformation in order to make suitable contents on mobile device. For managing some information that is required to do contents transformation, we need the DDR(Device Description Repository) which can store and search a variety of device information to grasp the constraints on the mobile device as compared with desktop devices.

Also, defining standard API is required to offer a service regardless of platforms. Hereupon W3C(World Wide Web Consortium) introduced DDR Simple API. However that specifies the limited functions of DDR, which is inevitable to be added for more precise search services.

In this paper, we expanded DDR Simple API and implemented DDR that supports DDL(Device Description Language) conversion and storing and searching device information.

Index Terms— Mobile, DDR, DDR Simple API

I. INTRODUCTION

Because of the improvement in the mobile device, the service provided for desktop devices is turning to the mobile environment. New technology which treats this issue has received plenty of attention, for example full-browsing, and yet cannot reasonably consume complete services on account of limited capabilities of mobile devices. To solve this problem, there's some research in progress. Its purpose is to serve optimized contents to mobile devices by making full use of characteristics of mobile devices.

As a part of these efforts, DDL(Device Description Language) based on XML was announced to share mobile device information between requesters in platform independent manner. But building up DDR system is needed to store and search device information described by DDL. Also, DDR internal standard interface is required to be defined to offer platform independent service when exchange device information with requesters. And another problem is that DDR Simple API describes only basic functions. As a result, the device information from DDR System is inaccurate and useless. So, we need to add other interfaces to existing DDR Simple API.

In response, we designed and implemented DDR based on DDR Requirements and DDR Simple API from W3C. Additionally, we research converting DDL each other in order to address interoperability challenges of DDL and secure device information. The last thing is extended interface to provide the users with accurate search service.

Manuscript received April 11, 2009 ; Revised May 20, 2009.

Yong-Soon Cho is with the Department of Computer Engineering, Paichai University, Daejeon, 302-735, Korea (Tel: +82-42-520-5640, Fax: +82-42-520-5405, Email: inadvanceof@pcu.ac.kr)

Hoe-Kyung Jung(Corresponding author) is with the Department of Computer Engineering, Paichai University, Daejeon, 302-735, Korea (Tel: +82-42-520-5640, Fax: +82-42-520-5405, Email: hkjung@pcu.ac.kr)

II. RELATED WORK

2.1 Device Description Language

DDL is a description of mobile device capabilities. This is referred on contents transformation to create or adapt the contents for only mobile devices. The prominent examples are UAProf(User Agent Profile) of OMA(Open Mobile Alliance) and WURFL(Wireless Universal Resource File) of wurfl.

UAProf that is based on CC/PP has more strict structure than WURFL. UAProf describes mobile device information using 7 components. There are HardwarePlatform, SoftwarePlatform, BrowserUA, etworkCharacteristics, WapCharacteristics,

PushCharacteristics and MmsCharacteristics. Each component has properties composed mobile device information.

WURFL has a concept of grouping properties. In WURFL structure, there are many groups to describe mobile device properties. The classification criterion of groups is a function of the property.

Because UAProf and WURFL use different description framework, the research about mutual converting of these DLLs is needed to secure device information from all.

2.2 DDR Simple API

As the publication announced by W3C, DDR Simple API specifies interfaces used at DDR system. Using the API in this publication, it is possible that DDR offer same service to all requesters without any restriction.

Basic interface is composed to Evidence, PropertyName, PropertyValue, PropertyValues, Service, Servicefactory interface, and these interfaces are doing the search function by each property, a group of properties and all of the properties.

But this publication does not define any properties and does not mandate the use of any particular Vocabulary of such properties. The DDR system can refer any vocabularies when it services device descriptions.

2.3 DDR Core Vocabulary

DDR Core Vocabulary defines the device property information which is essential for the contents creation of the mobile device and its aspects. There are a variety of properties that is needed to create or adapt the contents. Among them, some properties played a key role are described. This publication defines two aspects, webBrowser and device, which are collections of related properties. webBrowser aspect contains markupSupport, stylesheetSupport, imageFormatSupport, cookieSupport and others. The device aspect contains displayColorDepth, inputDevices, model, version and others.

2.4 DDR Requirements

W3C announced this publication to specify the general requirements that must be satisfied by DDR. Also, some use-case is described, which is a flow between the requester and provider and their role. The general requirements are showed in following.

[DDR API] Expose an API through which device descriptions can be retrieved
[DDR-MEASUREMENT] If a value returned by the DDR is an indication of measurement (such as width or resolution) then the units of measurement are expected to be unambiguous to the requesting party
[DDR-CONTEXT-KEY] Attempt to return a device description (or element thereof) based on identification of the hardware and/or software that the description relates to (the 'context key')
[DDR-INSUFFICIENT-CONTEXT-EXCEPTION] Handling the exception as detailed in the DDR API specification.
[DDR-VALUE-RANGE] return a range of all known values of a particular device capability for a given context key if requested to do so
[DDR-CONTEXT-AMBIGUITY] In case insufficient information is given
-Return a proximity value
-Return all result values
-End the service by exception handling

IV. DESIGN OF DDR SYSTEM

To provide mobile device information, we designed DDR System based on extended DDR interface which is added more functions at the DDR Simple API from W3C. And we do fulfill the DDR requirements at the aspect of the service functionality.

Basically, the DDR system in our paper comprises three sections. The following is a brief description of each section. First, the DDL Converting System is to secure device information from multiple DDL. Second section is the Device Description Search System for a diversity of search functions. Finally, there is the Web Interface which offers the interface which is that the users could search DDL. Figure 1 is architecture of DDR system.

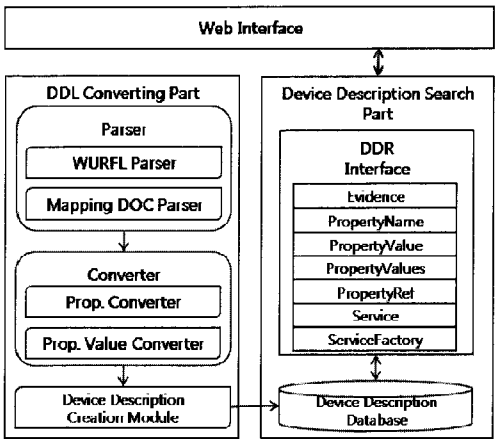


Fig. 1 The architecture of the Device Description Repository System

3.1 DDL Converting Part

The DDL Converting System is designed to handle interoperation problem between DDLs more than two. This problem occurs because they have different formats that describe the properties and its values of mobile device descriptions. This part converts the device properties into a format of target DDL's property name and value. This means that the DDR system can use all of the properties from more than two vocabularies.

In this paper, this system supports the way that transforms WURFL into UAProf because WURFL already has an enormous amount of device information. And we can add other modules to treat any other DDL by define Mapping files between these DDLs. This can affect the scalability of the DDR System.

With this view, there are some needed steps. One of these steps is a use analysis of properties in WURFL. The result of this analysis is used to match WURFL to UAProf. And some properties need adjustment process of their values.

3.1.1 Parser

The parser is a set of WURFL parser and mapping document parser. WURFL parser operates on extracting device information from the WURFL files. And mapping document parser's role is parsing Mapping Document. Mapping document has relations between WURFL properties and UAProf properties. WURFL parser uses the WURFL official API.

These parsers make collection object that stores device description to do cross reference from each vocabulary. The data from these parsers passed to Converter is used as base data on converting process to secure appropriate device information.

3.1.2 Converter

Converter refers to data passed from parser when performing conversion of properties and adjustment of property values. This process is needed because of the difference in description formats between DDLs. The converter gathers relation properties and their values, and creates new properties and their values which are appropriate at target DDL formats.

In this process, the converter uses the relation between DDLs. This relation is about properties and its values of device from each vocabulary include value format and type. So we designed XML schema to describe mapping relation between WURFL and UAProf. This scheme is presented in figure 2.

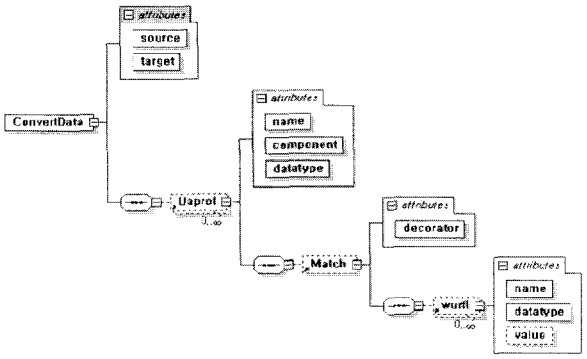


Fig. 2 XML Schema of DDL mapping Relation between UAProf and WURFL

3.2 Device Description Search Part

Device Description Search System can search device information upon request, which is composed of device description database and DDR interface.

3.2.1 Device Description Database

Device Description Database has real data of mobile device as it is stored by UAProf format. These data can be reorganized to UAProf by DDR interface. There are 4 tables to store device descriptions, which are consist of uaprof_devices, uaprof_capabilities, vocabulary, vocabularyIRI. The following is description of each table in detail.

- uaprof_devices – uaprof_devices table stores mobile device list. All mobile devices in this table have a number for 'device_ID' to distinguish from other devices. Also this number is referred from uaprof_capabilities table.
- uaprof_capabilities – uaprof_capabilities table stores many pairs of property names and property values of mobile devices. 'device_ID' is used to group the device properties by the device.
- vocabulary – vocabulary table has DDL information. For instance, components and properties. Now that only supports UAProf to service but it is possible to add other vocabulary that has more information about mobile device.
- vocabularyIRI – vocabularyIRI table is for namespace information to check DDL. Each DDL has specific namespace.

3.2.2 DDR interface Part

Our DDR System is based on DDR Simple API(W3C) and has some extended parts to more accurate and reliable search services. The followings are each interface's role in DDR System except extended part.

- Evidence interface – is to manage and define delivery context. Delivery context is a clue that is used to search the device descriptions. For example, user-agent, model and so on can be a clue to determine delivery context. It must support Evidence consisting of HTTP Header name and value pairs.
- PropertyName interface – is to search one property that is a characteristic of an aspect that can affect the web experience or consumption of contents on the mobile device.
- PropertyValue interface – is to create object that has the property value. This object models a PropertyRef with property value and sets the type of return value by defined methods.
- PropertyValues interface – is to create object that has the property values. This object models a PropertyRef with property values and offers the search service by the aspect or component.
- PropertyRef interface – is an interface that has properties information related with its aspect with the namespace. Using this interface, DDR system can search the properties in specific aspect. Because our DDR system uses UAProf, users can be provided with properties by component groups.
- Service interface – This interface created by ServiceFactory interface controls a series of DDR System services. Each request from user is dealt with one Service interface. It refers to other interfaces to search device information using Factory Methods, Query Methods, Information Methods, and Initialization.
- ServiceFactory interface – When the user requests device information, ServiceFactory create a Service interface's object with analyzing search conditions.

3.2.3 Extended interface

This DDR System has an extended interface of DDR Simple API that can provide improved search services. The differences from existing interfaces are range mode and text mode. A basic interface can't be satisfactory for users who want to find a model list in a specific condition. Taking account of this factor, range mode can make a result without model identifier like a user-agent. This result can be used to develop the contents for a group of devices or some devices that has specific features. Also, users can find mobile device information using text mode. It can be used to find the devices that have specific property value. This course is described in figure 3.

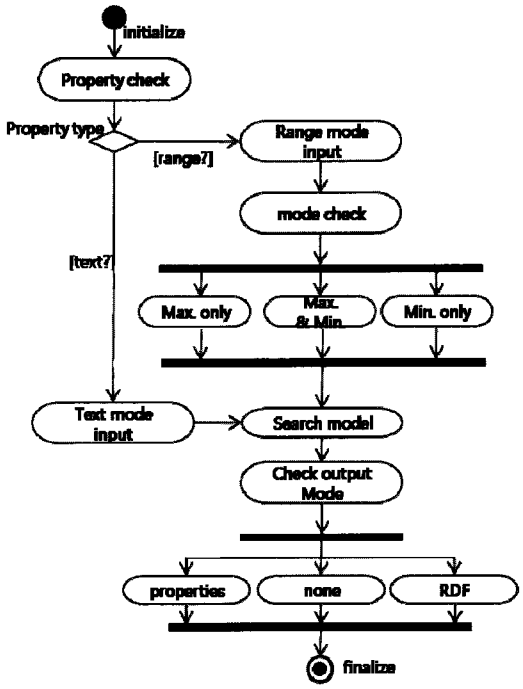


Fig. 3 The Diagram of processing in extended interface

At first step, the system checks property types whether range type or text type. If it is range mode, the system checks the mode values of minimum and maximum. Otherwise checked only text value. The next step is finding mobile device models using inputted conditions. At last the result is outputted by table or RDF file format.

3.3 Web Interface

Web Interface offers the user the search interface to get mobile device information. This interface is designed based on DDR system interface. It means that users can utilize all functions of DDR system by this Web interface.

IV. IMPLEMENTATION

We implement the DDR System in Linux Ubuntu 8.04 with JDK 5.0. And the Web interface is running on apache tomcat 5.5 environments. This system is based on the DDR Simple API, W3C, and offers mobile device descriptions to requesters. The vocabularies of this system are WURFL and UAProf. Also it supports mutual converting functions for these vocabulary data.

Figure 4 shows Web interface which we provide users to find appropriate mobile device information.

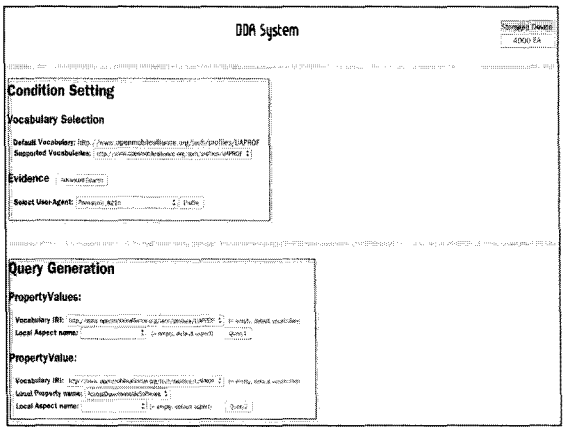


Fig. 4 Web interface of DDR System

Web interface of DDR System is composed to Condition Setting part and Query Generation part.

Condition Setting part is to select device model and DDL, which is used in Evidence interface. Query Generation part creates a query statement referring to set information by Condition Setting part. Also, the users can generate the query to diverse options that is aspect mode, one-property mode, many-property mode and so on.

When the query generated completely, DDR System presents device information on the screen by tables or returns a RDF file described using UAProf format. Figure 5 is an example of a part of RDF document.

```
<prf:BitsPerPixel>65536</prf:BitsPerPixel>
<prf:ColorCapable>NO</prf:ColorCapable>
<prf:ImageCapable>YES</prf:ImageCapable>
<prf:Model>VS400</prf:Model>
<prf:ScreenSize>128x128</prf:ScreenSize>
<prf:ScreenSizeChar>11x6</prf:ScreenSizeChar>
<prf:SoftKeysCapable>YES</prf:SoftKeysCapable>
<prf:StandardFontProportional>NO</prf:StandardFontProportional>
<prf:Vendor>Asmobile</prf:Vendor>
```

Fig. 5 the document from DDR System

For extended part, figure 6 shows that the extended service gives the user the ways of searching mobile information by range mode or text mode. Once the user requests specific property, the DDR System provides appropriate input windows, range mode or text mode. When the condition is completed, users can get the list of mobile device and their properties. Using this part, the CP can get the device information that has more usability to do contents transformation.

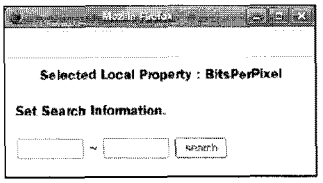


Fig. 6 The input window of Range mode

IV. CONCLUSIONS

Recently, there are many challenges to provide the mobile devices with contents equal to desktop devices because capabilities of the mobile device are getting more and more improved. However, it is difficult for the mobile device to present desktop contents on their own device because of the restriction of mobile devices. So DDL was defined to settle these problems by describing property and value of the mobile device.

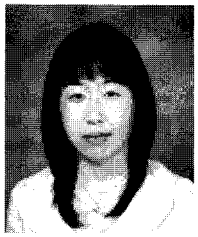
But DDR System is needed to manage mobile device information that is used for contents transformation process.

Therefore we implemented DDR System that stores and search mobile device properties, and extended DDR interface that functions more precious and reliable search service. Also, we propose an alternative method to cope with non-interoperability between DDLs, which is the way of transformation WURFL to UAProf. Through this research, we can contribute to a growth of mobile device contents as it is possible to get accurate device information from DDR System.

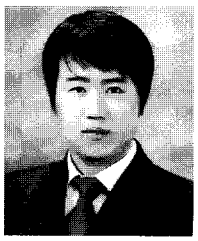
As a future work, we intend to design a framework to exchange the data of device information between contents transformation system and DDR System. Also, conversion from any other DDL is needed to secure more device information data.

REFERENCES

[1] Tae-hyun Kim, "A Study on Mobile Device Information Processing System", 2007.
[2] OMA, User Agent Profile 2.0, 2007
[3] Composite Capability/Preference Profile(CC/PP) : Structure and Vocabularies 2.0, 2007
[4] WURFL, Wireless Universal Resource File, http://wurfl.sourceforge.net
[5] W3C, DDR Requirements 1.0, Working Group Note 2007
[6] W3C, DDR Simple API, Working Draft, 2008
[7] W3C, DDR Core Vocabulary, Working Draft, 2007



Yong-Soon Cho
received the B. S. and M. S. degrees in computer engineering from PaiChai University, in 1999 and 2001, respectively. From 2008 she is in the Ph.D. program. Since 2003, she has worked as a lecturer of Paichai University. Her currently research interests include XML, MPEG-21, IPTV.



Young-Il Lee
received B.S degree in the Department of Computer Engineering from Paichai University, Korea in 2008. He is currently pursuing the M.S degree in the Computer Engineering, Paichai University, Korea, under the supervision of Prof. H. K. Jung(Multimedia Information Engineering Lab). His research interests include mobile device, IPTV, and MPEG-21.

Hoe-Kyung Jung
received B.S degree in 1987, and Ph. D. degree in 1993, in the Department of Computer Engineering from Kwangwoon University, Korea. During 1994- 2005, he worked for ETRI as Researcher. Since 1994, he has worked in the department of Computer Engineering at Paichai University where he now works as a Professor. His current research interests Multimedia Document Architecture Modeling, Information Processing, Web Services, Semantic Web, USN and MPEG-21.