

Personal Reader Agent: Personalized Access to Configurable Web Services

F. Abel, I. Brunkhorst, N. Henze, D. Krause, K. Mushtaq, P. Nasirifard and K. Tomaschewski

Distributed Systems Institute, Semantic Web Group, University of Hanover

Appelstraße 4, 30167 Hanover, Germany

readerteam@kbs.uni-hannover.de

Abstract

The Personal Reader Framework enables the design, realization and maintenance of personalized Web Content Reader. In this architecture personalized access to web content is realized by various Web Services - we call them *Personalization Services*. With our new approach of Configurable Web Services we allow users to configure these Personalization Services. Such configurations can be stored and reused at a later time. The interface between Users and Configurable Web Services is realized in a Personal Reader Agent. This Agent allows selection, configuration and calling of the Web Services and further provides personalization functionalities like reuse of stored configurations which suit the users interests.

1 Introduction

Within the Personal Reader project we already developed Web Content Reader like the *Personal Publication Reader* [Baumgartner *et al.*, 2005] which allows browsing publications in an embedded context. We also utilized and extended the SWAD-E Semantic Portal software [Reynolds *et al.*, 2005] to provide a Personal Semantic Portal [Henze and Abel, 2005]. Whereas these approaches are fixed in terms of the type of data that is provided, we now introduce a more generic approach: *Configurable Web Services* and the *Personal Reader Agent*. The Personal Reader Agent is a Web Application which enables users to select, configure and call Configurable Web Services. These Semantic Web Services need a detailed description of how they can be configured and how they are accessible. According to this description the Personal Reader Agent generates an interface that allows to adjust the Web Services. Personalization functionalities, like reuse of stored configurations of Web Services which suit the users interest, lead to an adaptive, personal Agent.

2 Personal Reader Agent

The Personal Reader Agent is on the one hand a kind of wizard that allows to select, configure and call Configurable Web Services and on the other hand it enables users to manage and reuse their saved configurations (*personalized access support*).

2.1 Configurable Web Services

Each Configurable Web Service has a detailed RDF description which defines parameters that can be used to ad-

just the Web Service (*Configurable description*). An example is the *My Ear Music Web Service*: This service allows users to configure parameters like *music category* or *maximum duration of songs*. It results in a *podcasting feed* containing items that are aggregated from arbitrary feeds but fulfill the adjusted parameters. A formal definition of the Web Service's configurable parameters, thus a *Configurable Web Service*, has the advantage that the process of configuring the Web Service can be abstracted and different configurations can be stored and reused. These two aspects are covered by the Personal Reader Agent.

2.2 Demonstration

Within a normal workflow the Personal Reader Agent offers the following steps:

1. **Discovery and Selection:** In this step the Agent requests human readable descriptions of the Configurable Web Services that are registered at our UDDI. Afterwards these descriptions are prepared for a selection by the user.
2. **Configuration:** After the first step the Agent reads in the Configurable descriptions of the selected Web Services and generates HTML forms so that the user can perform the configuration (see figure 1).
3. **Web Service Call:** After all selected Web Services are configured without violating the restrictions defined in the corresponding Configurable descriptions (e.g. *maxNumberOfInputs*, *type*, ...), the Agent is ready to call the Web Services. In this step the user further has the opportunity to save the configuration (see figure 2).
4. **Presenting the results:** This step is not part of the Agent application but rather a task that can be done by a common RDF browser or an application that provides a special view for certain RDF data, e.g. *MyEar View* visualizes podcasting feeds (see figure 3).

Stored configurations and a corresponding user model build the fundament to enable users to:

- **reuse their own configurations:** In order to allow users a faster access to the Configurable Web Services they can call these services also with a saved configuration as illustrated in figure 4.
- **reuse recommended configurations of other users:** The Agent allows the listing of configurations that might be relevant for a user. To determine relevant configurations the Agent utilizes relations between users that are defined by an ontology that models persons and their involvements in working groups. If two

persons (*User A* and *User B*) are involved in the same working group then the Agent suggests that configurations made by *User A* are also interesting for *User B*.

3 Conclusion

In this demonstration paper we describe how the Personal Reader Framework has been extended by a Personal Reader Agent for configuring Web Services according to a user's needs. The Personal Reader Agent provides a user interface and corresponding functionality to discover, select, configure, call and access Web Services.

At present the Personal Reader Agent is deployed as a prototype accessible via <http://www.personal-reader.de/agent/>. More informations can be found at <http://www.personal-reader.de/agent/documentation.pdf>.

References

- [Baumgartner *et al.*, 2005] R. Baumgartner, N. Henze and M. Herzog The Personal Publication Reader: Illustrating Web Data Extraction, Personalization and Reasoning for the Semantic Web. *European Semantic Web Conference ESWC 2005*, Heraklion, Greece, 2005.
- [Henze and Abel, 2005] N. Henze and F. Abel. User Awareness and Personalization in Semantic Portals. *4th International Semantic Web Conference*, Galway, Ireland, 2005.
- [Reynolds *et al.*, 2005] D. Reynolds, P. Shabajee, S. Cayzer and D. Steer. *Semantic Portals Demonstrator - Lessons Learnt*. SWAD-Europe deliverable 12.1.7, 2005.

A Figures

Step 2: Configuration of Webservice

Configure the selected Webservice to your needs...

MyEar Configuration

Configurable Things of MyEar Service

Query Keywords

Type in your keywords (separated with blanks) that should be within the podcasting item, e.g.: Jazz Swing

Value:

Enter at least 1 values.

itunes category

An itunes:category... [Info: This restriction may lead to few considered items or in extreme case also to timeouts!]

Duration

Specify a range for the length of the audio files. If you are searching for radio-like shows then you should enter a high Minimum Duration. Otherwise if you are searching for single songs then we recommend you to enter a low Maximum Duration.
Info: Some podcasting producer do not specify the duration of their audio files. If you do not want to pass over these podcasts then leave both fields blank.

Maximum Duration
The maximum duration of audio files (in minutes).

Minimum Duration

Figure 1: Configuration of Web Services

• Click on the "confirm + save"-Button to confirm and save your entries. Saving allows you to re-use your configuration at a later date.

MyEar Configuration - Values you entered/selected

Duration	Maximum Duration: 20 Minimum Duration: -
Maximum Number of Google API Calls	2
itunes category	Music
Query Keywords	Jazz

If you click the confirm-Button the Webservices you selected and configured are executed. Otherwise click the back-Button to edit your inputs.

Save your configuration of the Webservice

Saved configurations can be used at a later date to call the Webservice in a personalized way. It is also possible to adjust your configuration later. You have to enter at least a significant name for your configuration. Further you can enter a short description and select whether other users can utilize your configuration or not (standard is not).

Name: (e.g. MyEar - Jazz)

Description: (e.g. With this configuration the MyEar-Webservice returns a Jazz-Podcasting Feed...)

publish?: (if checked then other users can utilize your configuration)

Progress

1 Selected WebServices | 2 Configured WebServices | Ready to call WebServices | 3 Show Results

Figure 2: Saving Configurations - Entry of meta description about a configured Web Service

Personal Music Aggregation
My Ear

[about] [browse]

Browse the results of the MyEar Webservice

Starfrosch - Jazz
Channel: <http://www.starfrosch.ch/pod/jazz.xml>

Pamela's Parade - Les ventilateurs [insub02]

My Ear Player

Actual Playing:

Pamela's Parade wurde 2000 in Genf als Drum and Sax Duo gegründet, später kam eine Posaune und ein Doublebass dazu.

Pamela's Parade ist inzwischen ein Jazz Quartet beeinflusst von Folk und Free Jazz oder pulsierendem Funk.

<http://www.dinose.net/insubordinations/>

Download

Date: Wed Feb 22 12:30:15 CET 2006
URL: http://www.archive.org/download/INSUB02/insub02_Pamelas_Parade_01_Les_ventilateur...
Duration: 35339074

In The Groove, Jazz and Beyond
Channel: http://www.lasternet.com/mto/egroove/ITG_podcast.xml
A weekly Jazz radio show that airs on WJUS 91.7FM, Storrs CT. From the Jazz masters of past and present to emerging new artists performing jazz, fusion and funk. No smooth jazz here!

Great 80's
Here is some great piano jazz from both well known masters as well as little known talented jazz pianists.
Date: Sun Apr 23 05:37:07 CEST 2006
URL: http://podcast.sonic.com/audiofiles/kenamel/1120/rg_20060422.mp3
Duration: 41300000

Swiss is in the air for April 16, 2006

Figure 3: MyEar View

Your Configured Web Services

List of Web Services you have configured using the Personal Reader Agent is shown below. You have the opportunity to view, edit or delete your Configured Web Services. Further you can re-use a Configured Web Service by clicking on call.

Name	Description	Is Public?	Actions
Die Testkonfiguration	Dies ist eine Testkonfiguration, die nichts bewirkt...	false	<input type="button" value="view"/> <input type="button" value="edit"/> <input type="button" value="delete"/> <input type="button" value="call"/>
MyEar - Jazz	A simple search for podcasting feeds related with Jazz...	true	<input type="button" value="view"/> <input type="button" value="edit"/> <input type="button" value="delete"/> <input type="button" value="call"/>
Rock - MyEar	A bissel Rock...	true	<input type="button" value="view"/> <input type="button" value="edit"/> <input type="button" value="delete"/> <input type="button" value="call"/>

Figure 4: List of configured Web Services