

International Semantic Service Selection Contest



S3 Contest Participation Guideline

Version 1
June 1, 2007

Minko Dudev, Patrick Kapahnke, Matthias Klusch, Jozef Mišutka
DFKI GmbH, Saarbrücken, Germany

Table of contents

1. Introduction.....	3
Matchmaker evaluation environment.....	4
How to plug in my matchmaker	4
2. How to participate in the S3 Contest - A Walk-Through Example	6
3. S3 Contest plugin interface documentation.....	8
de.dfki.sme2.IMatchmakerPlugin	8
4. Support and contact.....	10

1. Introduction

About the S3 Contest

While the increasing number of already available semantic Web applications and conventional Web services appears impressive, useful and potentially attractive even to major business stakeholders, there still is a major lack of implemented and deployed tools for semantic Web services retrieval and engineering in practice. As one response to this challenge, the Semantic Service Selection (S3) contest initiative formed at the 5th International Semantic Web Conference ISWC 2006 in Athens, USA. It aims at encouraging the rapid and innovative development of tools for semantic service selection (matching). The S3 contest complements the SWS challenge (www.sws-challenge.org) by its comparative evaluation of SWS discovery tools based on respective SWS retrieval test collections.

The call for participation in any edition of the S3 contest is continuous and open.

The first edition of the contest focuses on OWL-S service matchmaking, though future editions will add other semantic Web service description formats (SAWSDL, WSML) as soon as respective test collections also for these formats are available.

Evaluation of semantic Web service matchmakers will base on classic performance metrics recall/precision, F1, average query response time, and a given service retrieval test collection. Each valid entry will be peer evaluated by members of the S3 contest jury and the S3 contest organisation committee.

This is a continuous call for participation in the S3 contest.

Participation in this contest is NOT dependent on the registration to any event (workshop, conference). Every submission that is complete (see section 2) will be considered valid and evaluated.

S3 Contest Organisation Committee

Matthias Klusch (DFKI, Germany)

Alain Leger (France Telecom Research, France)

David Martin (SRI International, USA)

Massimo Paolucci (NTT DoCoMo Research Europe, Germany)

Abraham Bernstein (University of Zurich, Switzerland)

Terry Payne (U Southampton, UK)

Aims & scope of the S3 contest in 2007

This year's edition of the contest focuses on the comparative evaluation of the performance of OWL-S service matchmaking tools based on respective test collection. Future editions of the S3 contest will add other semantic Web service description formats (SAWSDL, WSML) as soon as respective service retrieval test collections also for these formats are available.

Performance evaluation metrics include the classic R/P, F1, average query response time and resource usage.

The public final round and award giving of the S3 contest in 2007 will be hosted by the international workshop SMR2 on service matchmaking and resource retrieval at the ISWC 2007 conference on November 11, 2007, in Busan, South Korea.

Open and continuous call for entries to the S3 contest:

Deadline for submission to the S3 Contest in 2007: **November 5, 2007**

However, submissions of entries to the S3 contest that reach us **after this date will be automatically taken for the 2008 edition**.

In case of any question on how to participate and technical support, please contact

- Patrick Kapahnke, Patrick.Kapahnke@dfki.de
- Matthias Klusch, klusch@dfki.de

Matchmaker evaluation environment

For testing and evaluation of service matchmakers, a special Semantic Web Service MatchMaker Evaluation Environment (SME²) will be used. The SME² has been developed at DFKI and consists of two parts: The evaluation tool and its easy to use plugin interface. The interface is required to be used by the participant to plug his/her particular matchmaker into the evaluation environment.

After loading a matchmaker plugin the SME² executes it, determines all necessary information about the retrieval performance of the matchmaker, compares it with others plugged into it, and returns the summary report of the results.

How to plug in my matchmaker

Each matchmaker must create a plugin for the environment. It must implement the SME² plugin interface which is then used to perform the testing queries. The implementation of the plugin interface together with the matchmaker should be provided in binary form (e.g. java archive - jar). Dependencies and the name of the implementing class must be described in an xml specification file. And finally all dependencies must be submitted either in binary form or by URL.

Any additional requirements needed to set up the matchmaker properly must also be described in detail and submitted.

In brief: How to participate in the S3 contest?

Create your entry to the S3 contests which consists of the following files

- a. Library of matchmaker plugin
- b. plugin.xml
- c. Additional libraries your project depends on either in binary format or as a set of URLs where they can be downloaded
- d. Full contact details (Name, Affiliation, email, phone) and name of the matchmaker

Please look at the detailed **walk-through example** below of how to create an entry to the S3 contest in the following section. You may also look at the example Java code of a matchmaker plugin given in this package.

Send your entry at any time

In a zip file via email to klusch@dfki.de **or** on a CDROM via postal mail to Dr. Matthias Klusch, DFKI, Stuhlsatzenhausweg 3, D-66123 Saarbruecken, Germany

2. How to participate in the S3 Contest - A Walk-Through Example

The SME² evaluation tool provides a very simple but generic matchmaker plugin interface. To test a matchmaker the evaluation environment needs two things

- a java library implementing the plugin interface and
- a specification file with additional information about the matchmaker.

In the following, we will describe the process of how you can realize the plugin of your matchmaker to the evaluation environment in more detail:

1. **Download** the plugin interface either as interface source or interface library

2. **Preparation**

- 2a. interface source - unpack and copy the content of *src* directory to source directory of your matchmaker project

- 2b. interface library - unpack and copy the content of *lib* directory to library directory of your matchmaker project (do not forget to add it to your project libraries, if not done automatically)

4. **Implement the plugin interface** (de.dfki.sme2.IMatchmakerPlugin) either in a new class or extend an already existing class.

NOTE: The class must have a public default constructor.

5. **Build the project and create a java library (.jar)**

6. **Create plugin.xml** which must contain

- a. name of the library containing the class which implements IMatchmakerPlugin (e.g. plugin.jar)
- b. the full path of the class implementing IMatchmakerPlugin (e.g. org.matchmaker.SME2Plugin)
- c. path to one or more directories with additional libraries (e.g. libs)

Example of plugin.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<plugin>
  <jar>plugin.jar</jar>
  <class>org.matchmaker.SME2Plugin</class>
  <directory>./libs</directory>
</plugin>
```

NOTE: The schema for xml validation can be downloaded [here](#).

7. **Send us your entry to the S3 contest** that consists of the following files:

- a. Library
- b. plugin.xml
- c. Additional libraries your project depends on either in binary format or as a set of URLs where they can be downloaded
- d. Full contact details (Name, Affiliation, email, phone) and name of the matchmaker

Please check the library versions carefully before sending!

NOTE: If your matchmaker does require any additional support, please send us also a very detailed step-by-step description how to set up your matchmaker.

3. S3 Contest plugin interface documentation

de.dfki.sme2.IMatchmakerPlugin

public interface IMatchmakerPlugin

SME² matchmaker evaluation environment plugin interface.

The evaluation environment loads a plugin and instantiates the class implementing this interface (the class is specified in **plugin.xml**).

The **input(java.net.URL)** method is called to let the matchmaker initialize and parse the advertised services. This can be a time consuming operation and will not be counted in the average query response time (aqrt). However, it will be counted into the total execution time.

After the initialization is complete the matchmaker should be ready to process queries. Queries are performed by calling **query(java.net.URL)** method.

Method Detail

void
input (java.net.URL serviceURL)

Initializes matchmaker.

Advertised services identified by **serviceURL** should be processed in this method. After returning from this method the matchmaker should be initialized and ready to accept queries.

Parameters:

serviceURL - URL identifying advertised services. It can be either a local / remote file or a local directory.

java.util.Hashtable<java.net.URL,java.util.Vector<java.net.URL>>
query (java.net.URL queryURL)

Performs a query returning matched services, sorted by matching factor.

This function must perform one or more queries identified by queryURL parameter. The result is a container of < query, container of matched services > pairs.

The container must be sorted by matching factor. First item in the container must be the nearest match etc.

Parameters:

queryURL - URL identifying queries. It can be either a local / remote file or a local directory.

Returns:

Container of queries and query results. The results must be sorted by matching factor. The first item in the container must be the nearest match.

4. Support and contact

For technical support, please contact:

- Jozef Mišutka at jmisutka@gmail.com
- Patrick Kapahnke at Patrick.kapahnke@dfki.de

For general inquiries on the contest, please contact:

- Dr. Matthias Klusch at klusch@dfki.de, <http://www.dfki.de/~klusch>