



Contribution ID: 79

Type: **Oral**

## Enabling Large Data Transfers Between Web Services

*Thursday 15 April 2010 09:40 (20 minutes)*

Despite numerous benefits, many Web Services (WS) face problems with respect to data transport, either because SOAP doesn't offer a scalable way of transporting large data-sets or because orchestration workflows (WF) don't move data around efficiently. In this paper we address both problems with the development of the ProxyWS. This is a WS utilizing protocols offered by the Virtual Resource System(VRS)(Java API used by the VBrowser), to enable other WS to transfer and access large datasets without modifying WS nor the underlying environment.

### Detailed analysis

There is currently an abundance of deployed (legacy) WS using SOAP, which fail to produce access and return large datasets. Moreover, orchestration WF causes WS to pass messages containing data back through the WF engine. To address these problems we introduce the ProxyWS: a WS that is able to access data from remote resources (GridFTP, LFC, etc.), thanks to the VRS, and also transport larger data produced by WS, both legacy and new. For the ProxyWS to be able to provide larger data transfers to legacy WS, it has to be deployed on the same Axis-based container, just like a normal WS. This enables clients to make proxy calls to the ProxyWS instead of a legacy WS. As a consequence the ProxyWS returns a SOAP message containing a URI referring to the data location. For new implementations the ProxyWS is used as an API that can create data streams from remote data resources and other WS using the ProxyWS. This approach proved to be the most scalable since WS can process data as they are generated from producing WS. Thus with the introduction of the ProxyWS we are able to provide a separate channel for data transfers, that allows for more scalable SOA-based applications.

### Conclusions and Future Work

We have presented the ProxyWS, which may be used to support large data transfers for legacy and new WS. We have verified its performance to deliver large datasets on two real-life tasks: Indexing using WS in a distributed environment and annotating documents from an index. From our experiments we have found that ProxyWS is able to facilitate data transports where normal SOAP messages would have failed. We have also demonstrated that with the use of the ProxyWS legacy WS can scale further, by avoiding data delivery via SOAP and by delivering data directly from the producing to the consuming WS.

### Impact

Many different approaches have been introduced in an attempt to address the problems mentioned earlier. Examples of these include Styx Grid Services, Data Proxy Web services for Taverna and Flex-SwA. Some noteworthy features of these approaches are: Direct streaming between WS, Usage of alternative protocols for data transports, and larger data delivery to legacy WS. However, each of these examples only addresses one part of the problem and, furthermore, do not include any means of allowing access to remote data resources. Leveraging these existing proposals and combining them with the VRS we implemented a ProxyWS. To validate it,

we have tested its performance using 2 data-intensive WF. The first is a distributed indexing application that uses a set of WS to speedup the indexing of a large set of documents, while the second relies on the creation of that index for retrieving and recognizing protein names contained in results coming from a query. With the use of the ProxyWS we are able to retrieve data from remote locations (8.4 GB of documents for indexing), as well as to obtain more results relative to a query (8300 documents using the ProxyWS versus 1100 using SOAP).

## Keywords

web services, data transfers

## URL for further information

<http://staff.science.uva.nl/~skoulouz/pmwiki/index.php/ProxyWS/ProxyWS>

**Author:** Mr KOULOZIS, Spiros (Informatics Institute, University of Amsterdam, the Netherlands)

**Co-authors:** Dr BELLOUM, Adam (Informatics Institute, University of Amsterdam, the Netherlands); Mr MEIJ, Edgar (Informatics Institute, University of Amsterdam, the Netherlands)

**Presenter:** Mr KOULOZIS, Spiros (Informatics Institute, University of Amsterdam, the Netherlands)

**Session Classification:** Data Management

**Track Classification:** Software services exploiting and/or extending grid middleware (gLite, ARC, UNICORE etc)