Web Services for the DDSM and Digital Mammography Research

Chris Rose, Daniele Turi, Alan Williams,
Katy Wolstencroft, and Chris Taylor

Imaging Science and Biomedical Engineering,
The University of Manchester,
Manchester, M13 9PT, United Kingdom
http://www.medicine.manchester.ac.uk/isbe/

Abstract. The Digital Database for Screening Mammography (DDSM) is an invaluable resource for digital mammography research. However, there are two particular shortcomings that can pose a significant barrier to many of those who may want to use the resource: 1) the actual mammographic image data is encoded using a non-standard lossless variant of the JPEG image format; 2) although detailed metadata is provided, it is not in a form that permits it to be searched, manipulated or reasoned over by standard tools. This paper describes web services that will allow both humans and computers to query for, and obtain, mammograms from the DDSM in a standard and well-supported image file format. Further, this paper describes how these and other services can be used within grid-based workflows, allowing digital mammography researchers to make use of distributed computing facilities.

1 Background

The DDSM [6] provides high-resolution digitised mammograms, expert ground-truth and metadata (including the date of study and digitisation, the Breast Imaging Reporting and Data System (BI-RADS) [1] breast density and assessment categories, a subtlety rating, the type of pathology and detailed categorisation of the nature of the perceived abnormality using the BI-RADS lexicon). The DDSM is available free of charge by File Transfer Protocol (FTP).

While the DDSM does provide software to decode their mammograms[1], the default distribution of this software does not build under modern compilers without modification, a step that may prove difficult to those with insufficient experience of C/C++ software development for UNIX-like operating systems. Furthermore, even when properly compiled, the DDSM software outputs the image data as a stream of raw bytes; one then has to normalise these according to the model of digitiser used to image the original films and then create an image file that is readable by one's image analysis software environment. An introduction to web services is given in Section 1.1. Section 2.1 describes a web service that allows digital mammograms from the DDSM to be obtained

[1] In particular the DDSM's `jpeg` program.

Susan M. Astley et al. (Eds.): IWDM 2006, LNCS 4046, pp. 376–383, 2006.

in a standardised and well-supported lossless file format. Section 2.2 describes a service that allows groundtruth images to be obtained in the same file format.

While a web-based query tool is provided by the DDSM, it is useful only to human users or automated tools that have been specifically designed with the DDSM in mind. If the metadata were in a more useful format, it could easily be exposed for both human and computer use. Section 2.3 describes a formal ontology that has been developed to describe the DDSM resource and a web-based user interface to allow users to query the ontology.

Section 2.4 describes how web services can be used together within *workflows* to run full experiments and how a full record of how such experiments were performed can be recorded by capturing provenance events. Section 2.5 details a supporting website for the work described in this paper.

1.1 An Introduction to Web Services

The concept of web services may best be explained with a simple example of a hypothetical scientist named Bob who lives in Manchester, UK. Bob has a CAD algorithm that other scientists want to use. Traditionally, Bob would package his CAD algorithm into some form that is easily installed and run by other scientists. He would then deliver it to those scientists via Internet download or on physical media (e.g. CD-ROM). However, Bob might not be able to let other scientists run his software on their computers because:

- Bob may not have planned to share his software and may have made assumptions in its design that limits its portability;
- users might need an expensive license to use a required proprietary library;
- the software may need access to a resource (e.g. a large database) that resides at Bob's lab;
- Bob might frequently update his software, so making each update available to all his users might be troublesome;
- packaging the software for easy installation might be too time-consuming for a busy research scientist.

Bob might decide that it is easier to allow other scientists to run his software on his computer, accessing it via the Internet. This can be achieved by exposing his software as a web service. Using an appropriate piece of client software that implements the Simple Object Access Protocol (SOAP) standard [5], other scientists can run Bob's CAD algorithm on their data. In this way, Bob's CAD algorithm can be used by remote scientists as if it were installed on their computers, or integrated into software as if it is a library containing the required functionality.

The "interface" to a web service—the location of the computer that provides the service and a specification of its inputs and output and their data types—is described using the Web Service Description Language (WSDL) [2]. The URL of a service's WSDL file is all that is needed for a SOAP client to be able to use the service[2].

[2] The WSDL files for the services described in this paper can be obtained from the website described in Section 2.5.