

The Use of OGSA-DAI with IBM DB2 Content Manager for Multiplatforms in the eDiaMoND Project

Manfred Oevers, Brian M Collins, Alan Knox and John Williams

IBM United Kingdom Ltd, Hursley Park, Winchester, SO21 2JN, United Kingdom

February 2004

Introduction

IBM has joined with Oxford University, Mirada Solutions Ltd and a group of clinical partners to build a prototype Grid that will support diagnosis of breast cancer and provide medical professionals with additional information to help treat the disease. The clinical partners comprise four research hospitals, which have Breast Screening Clinics and associated universities. These are:

- Churchill Hospital and Oxford University
- Guy's and St Thomas' Hospital and King's College London
- St George's Hospital and University College London
- The Ardmillan and Edinburgh University

They will provide the requirements and will prove the prototype implementation.

The project has been named eDiaMoND [1,2] (Digital Mammography National Database) and will be one of the first Grids based on commercially available technology, including state-of-the-art software developed by Mirada Solutions to standardize new and existing mammograms. This paper will discuss some of the challenges faced in creating a Grid for mammography and focuses particularly on the experiences in using OGSA-DAI (Open Grid Services – Data Access and Integration) technology [3] to provide access to the images held in a virtualised data store. In particular we present our experience of making IBM DB2 Content Manager [4] data sources accessible through OGSA-DAI and what benefits resulted from the use of OGSA-DAI.

Background

Breast Screening in the UK is a major success [5,6]. It started in 1988, screened over 1.5 million women aged 50 to 64 and is estimated to save over 230 lives every year through early detection of cancer. It is viewed as a flagship programme within the National Health Service. However, the programme currently relies on manual patient files (paper and film) and the screening process requires much administrative support as a result.

The potential change from film to digital X-Ray equipment will necessitate a change to patient records. Mammograms require the highest resolution to enable radiologists to detect some of the early signs of cancer. In a digital world this means 50 micron resolution and digital images of around 32 Megabytes each. Though electronic patient records and digital mammograms promise additional benefits such as computer aided

detection they will also require specialised technologies to manage the capture, storage, retrieval, and movement of millions of large digital images. This is the key challenge which eDiaMoND is addressing.

Additionally there will be opportunities to exploit the technology developed by eDiaMoND in related areas. For example, other types of medical imaging, such as MRI, PET, Ultrasound and other digital X-Ray images, as well as non-Health related areas.

UK Breast Screening

Today

The UK Breast Screening Programme has almost 100 Clinics nationwide offering this free service. Today women aged between 50 and 64 are screened every 3 years and one mammogram view of each breast is taken. The mammograms are each “read” at a clinic by two separate full-time radiologists specifically trained in mammography (currently there are about 230 of these specialists).

Of the 1.5 million women who are screened annually approximately 80,000 are recalled because some anomaly is detected during the “reading”, and about 1 in 8 of those recalled are found to have cancer. These can be treated successfully in a high percentage of cases because Breast Screening allows for early detection. However, it is estimated that without the Breast Screening Programme over 230 women would have died because their cancers would not have been detected at an early enough stage.

Challenges

The incidence of breast cancer inexorably increases with a woman’s age. Because of the success of the current Breast Screening Programme, and based upon studies in the UK and other countries, it has been decided to increase the upper age limit from 64 to 70 years. In addition it has been found that detection is improved if two mammogram views, Cranio-Caudal and Medio-Lateral Oblique, are taken of each breast. Finally, over the next ten years there is a natural demographic increase in the population of women who will qualify for the Breast Screening Programme.

The combination of the above will increase the number of women screened to 2 million annually with 120,000 recalled and 15,000 cancers detected. This will result in nearly 50% increase in workload for the radiologists but promises a dramatic increase in the number of lives saved to around 1,250 annually.

Workflow

On average, for every 1000 women who attend screening, 960 are immediately given the “all-clear” and asked to attend again in three years time. Of the 40 women who are recalled for assessment only 5 are found to have cancer. Of the original 1000 one woman on average who was given an “all-clear” will develop breast cancer before she attends her next screening. These so-called “interval cancers” are carefully reviewed to see if there were symptoms that were missed at the previous screening. Thus improving the skills of radiologists in detecting very early signs.

Today there is no easy access to screening records for epidemiological studies and teaching because they are on paper and film and located in filing systems across 100 clinics.

Project

Deliverables

eDiaMoND will deliver prototype implementations, in 2004 for evaluation by the clinical partners at the four hospitals. In addition a reference-architecture will be produced which will describe what would be needed to implement a production system.

The deliverables will cover the Grid infrastructure, Grid connected workstations, the database schema for the storage and retrieval of images and associated patient data, the ability to run computations for computer aided detection (CADe) and diagnosis (CADi), and finally the required hardware, software, and network required to provide several different qualities of service.

Scope

eDiaMoND is the replacement of mammograms (Film) and paper records with digital studies. A digital study will comprise metadata from the paper records and digital mammograms either scanned from film or directly from digital X-Ray machines. The metadata will be separate from but linked to the digital mammograms. Images are 32 Megabytes each and will result in approximately $\frac{1}{4}$ Petabyte of data being stored every year.

Compute

Mammograms have different appearances depending on the image settings and the characteristics of the acquisition system. This currently means that films taken three years apart are difficult to compare and when digitised need to be “standardised” in order for a valid temporal comparison to be made on computer. The standardisation algorithm, called SMF™ (Standard Mammographic Form), has been developed by and is a trademark of Mirada Solutions Ltd. The algorithm is numerically intensive and essentially calculates breast density from the mammogram, in terms of percentage fatty tissue and percentage glandular tissue in the whole volume of a compressed breast. It will allow computer aided detection algorithms to more easily identify certain signs, such as micro-calcifications as well as allowing a computer to reconstruct 3-Dimensional views of the breast from two views to aid accurate location of an object.

Non-Functional Requirements

The functional requirements of eDiaMoND dictate the logical architecture, which is that the database and compute resource are viewed as one logical resource. However, there are a number of key non-functional requirements, which place constraints on the physical architecture and hardware and software implementation.

The major non-functional requirements that affect eDiaMoND are as follows:

- Ethics – This will require both reversible and irreversible anonymisation of the data depending upon the application and the specific user access rights.
- Legal – Because some of the features in breast cancer are at the individual pixel level it is not acceptable to use lossy compression. If allowed mis-diagnosis could arise with consequent claims for malpractice.
- Security – All data will have to be encrypted when being transmitted over networks with the option of being stored in an encrypted form; even when the data is anonymised.
- Performance – Today a radiologist typically screens one patient every 30 seconds by looking at 8 mammograms (4 for the current and 4 for the previous screening). Current carousel lightboxes move to the next patient within 5 seconds. This means that the system needs to switch between patients within 5 seconds and load up 256 Megabytes for display every 30 seconds
- Scalability –The system should exhibit as near linear scalability as possible with up to 8 million images added per year across 100 Clinics.
- Manageability – It must be assumed that there are negligible IT skills within a hospital and hence remote system administration must be possible of any equipment located in the hospitals.
- Auditability – A complete audit trail of all additions and update to a patient's data must be available. This is particularly true to facilitate non-repudiation.

Prototype

eDiaMoND will comprise a Distributed Grid which supports the Scope but is implemented to meet the Non-Functional Requirements. Therefore there will be workstations and Grid nodes located at each of the four hospitals.

High Level Architecture

X-ray films are scanned and converted to DICOM file format [7] (Digital Imaging and COmmunications in Medicine), which combines the image data (pixels) with non-image data about the patient (e.g. name, birth date, etc.) and other meta-data (e.g. which equipment was used to capture the image) into a single file. The architecture of eDiaMoND handles these image data and non-image data in two different ways. The non-image data is stored in an IBM DB2 UDB Enterprise Server Edition V8.1 (DB2 for short) database to enable sophisticated searches (e.g. epidemiological studies). The individual DICOM files are stored within IBM DB2 Content Manager for Multiplatforms V8.1 (CM for short).

Both products are accessed through OGSA-DAI (Open Grid Services Architecture – Data Access and Integration) Grid Services [3]. This enables the sharing of a secure context to have consistent access control across both systems. This is depicted for a single breast care unit (BCU) in Figure 1.

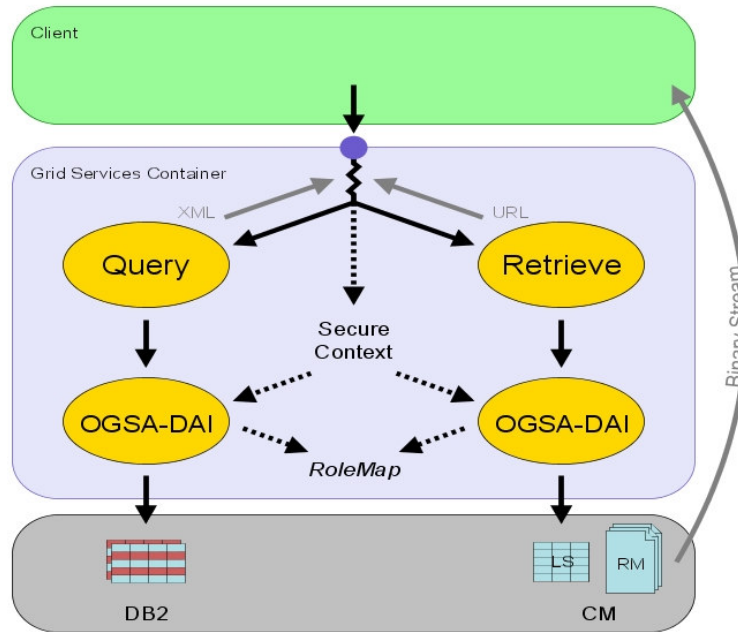


Figure 1 - eDiaMoND data flows for a single BCU

The eDiaMoND Grid consists of multiple such BCUs and the task is to create a virtual data store for the image and non-image data across all those BCUs and shield the user from the underlying implementation. The two basic strategies for creating a virtual database in the eDiaMoND Grid are replication of data across all image server nodes or distribution of a query across all image server nodes. These strategies can be implemented either at the OGSA-DAI level or the product level, giving a total of four possibilities, as shown below in Figure 2. The same strategies exist for CM as well as DB2 data source. The four possibilities are as follows (DB2):

1. Keep the data in physically separate locations and use DB2 Information Integrator (federation) to access the data. The federated server is then accessed on the Grid via OGSA-DAI.
2. Use OGSA-DAI distributed query processing (DQP) to distribute the query, this implies exposing each data source as a Grid service over which DQP is executed, which is itself a Grid service.
3. Use DB2 replication to construct a single data source, which will in turn be exposed as an OGSA-DAI Grid service.
4. Use Grid replication to construct a single data source.

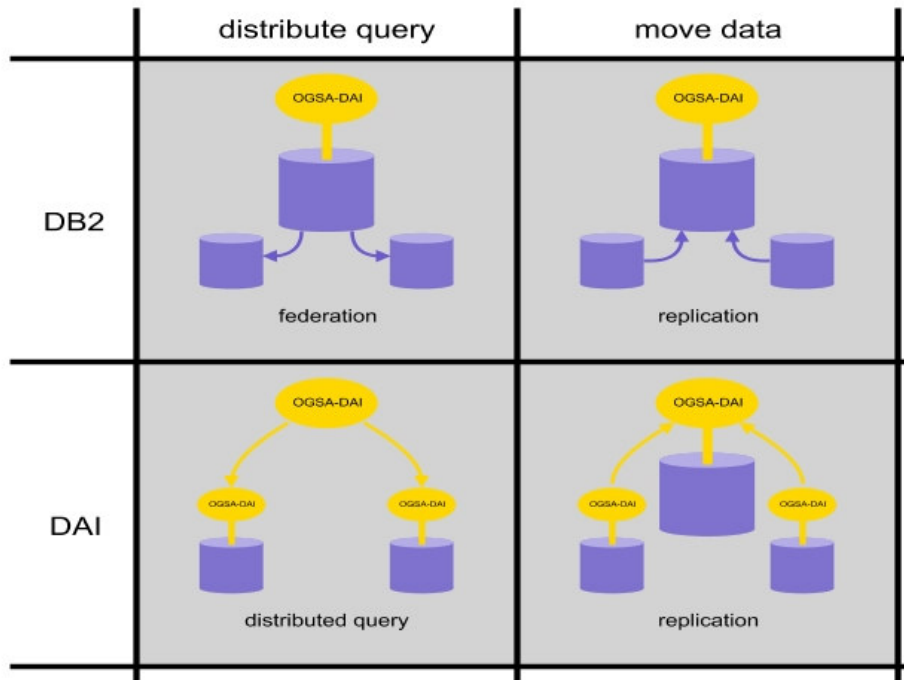


Figure 2 - eDiaMoND data flows

The project has chosen to implement the first possibility in the prototype and to research the use of DQP. This design is also used to virtualise the image store, which is based on CM. DB2 Information Integrator for Content (II4C) is used to enable federated searches over CM systems at the different clinical evaluation sites.

Content Manager Grid Enablement

Architecturally CM consists of two components, the Library Server (LS) and the Resource Manager (RM). The LS acts like the catalogue of a Library and the RM is the bookshelf. Just like in a real library there is more than one bookshelf, CM can support more than one RM, which makes the architecture scalable. LS and RM collaborate to provide secure, transactional access to the content stored in CM. For more information see [8]

OGSA-DAI already supports relational and XML data sources and provides a flexible framework into which to plug other data sources. Since the query language for Content Manager is XPath it seemed quite natural to expose Content Manager as another XML data source. OGSA-DAI is a reference implementation of the Data Access and Integration Services specification of the Global Grid Forum [9] and is therefore a good starting point for the Grid-enablement of Content Manager. However the model of interaction with a data source in Content Manager differs from that of ordinary databases. Content Manager provides an implementation of CORBA dynamic data objects (DDO) and their extensions, namely extended data objects (XDO).

These are accessible through an object oriented API with Java™ and C++ language bindings. We show how the internals of the OGSA-DAI software made it easy to enable CM and II4C as a data source and provided considerable developer productivity.

Short Summary of OGSA-DAI

OGSA-DAI is logically a collection of collaborating Grid Services that act as proxies for the systems that actually hold the data. The systems that can currently be accessed from the Grid via OGSA-DAI are relational databases and XML databases. Grid Data Service Factories (GDSF1, GDSF2) are created by a Grid Service Container as persistent Services that represent a data source, a database inside a RDBMS or a collection inside an XMLDB. The factories register themselves with a DAI Service Group Registry (DAISGR1), which is also persistently created and only allows services that are Grid Data Service Factories to register with it. An analyst (A1) searches the registry for a Factory that can create a Grid Data Service against the required data resource.

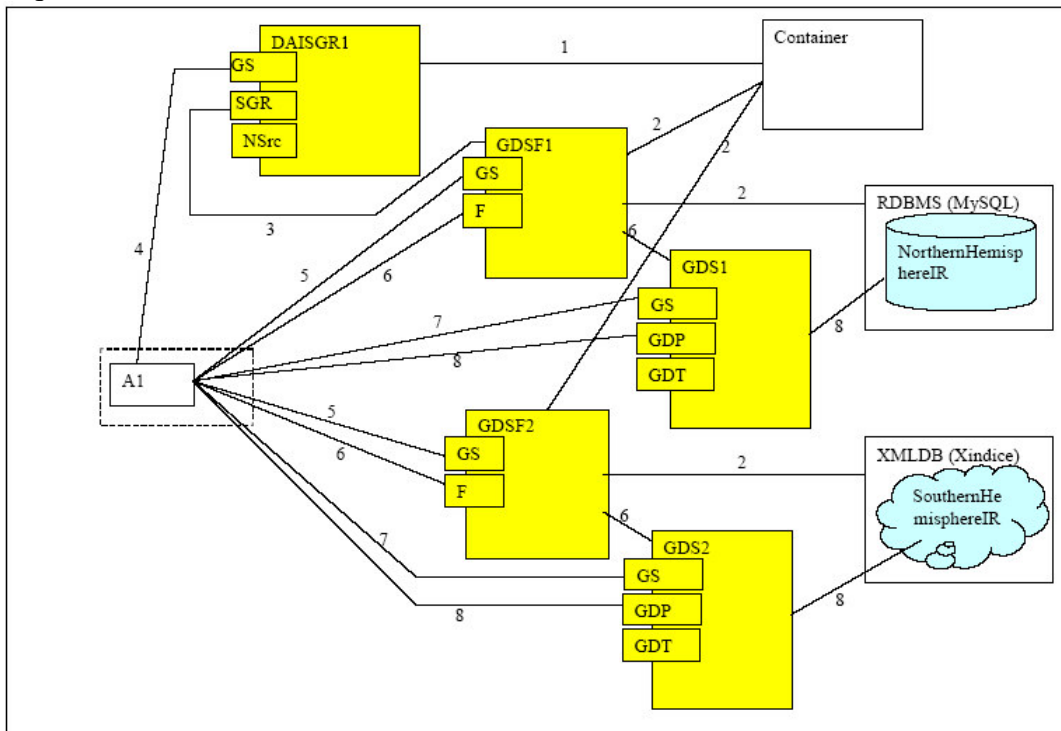


Figure 3 - Collaborating OGSA-DAI Services (taken from [5])

The Grid Data Service Factory can be introspected by the analyst to establish its precise capabilities. Subsequently the analyst requests the creation of a Grid Data Service. This GDS corresponds to a connection to the underlying data source and the analyst interacts with the GDS by submitting an XML *perform document* that specifies some request. The result of the request is returned to the analyst as an XML document. For more information see [10].

Perform Documents and Activities

Activities are the operations that a GDS will support for a given data source. Activities are the smallest unit of work that a client can request from a data source. OGSA-DAI comes with a set of predefined activities, e.g. `sqlQueryStatement` activity that performs an SQL query. Each Grid Data Service can be configured as to which activities it supports, which gives control as to which functionality is exposed to the

Grid. Activities are also an extension point of OGSA-DAI, i.e. new activities can be written and their functionality be added to the system. Furthermore activities can have inputs and outputs by which they communicate with each other. This enables the construction of sophisticated processing sequences where e.g. an SQL query is performed, the output transformed into a report document, which is ZIP compressed and delivered to a URL via http. Each activity has an associated XML schema that describes how a request for this activity is to be formulated within the perform-document send by a client. The order of activity elements within a GDS Perform Document is not relevant. The order is implicitly determined by one activity requesting by name as input the output of another activity. If an activity has neither input nor output it can be executed asynchronously and only a status is send back to the client. For more information see [11, 12].

Content Manager as an OGSA-DAI Data Resource

To enable Content Manager for OGSA-DAI means to have a Grid Data Service Factory that can create a Grid Data Service that can connect to a Content Management System. It was possible to map Content Manager concepts to database concepts in such a way that the infrastructure provided by OGSA-DAI could be used without change, in particular this means that the configuration points available for a Factory are sufficient to configure a CM Factory.

OGSA-DAI concept	Corresponding CM concept
JDBC Driver, e.g. com.ibm.db2.jcc.DB2Driver	Data store object, e.g. com.ibm.mm.sdk.server.DKDatastoreICM
Driver URI, e.g. jdbc:db2://localhost:50000/SAMPLE	Data store name, e.g. ICMNLSDB

In order to get access to a data resource one has to also specify a *driverManagerImplementation* class that manages the driver. This class can e.g. support connection pooling. For JDBC the *getConnection()* method on the *DriverManager* returns a *Connection* Object. For Content Manager this is different in that the data store has a *connect()* method that puts the data store into a connected state. A data store after a call to *connect()* therefore corresponds to a *Connection* object in JDBC. The *driverManagerImplementation* class is one of the extension points of OGSA-DAI and is specified in the configuration file for the Grid Service Factory.

OGSA-DAI also provides for the mapping of Grid credentials (i.e. X509 Certificates) to credentials that are known to the data source. Because CM uses DB2, the role mapping already in OGSA-DAI can be used without change. The class the implements the mapping of Grid credentials to data source credentials is a further extension point of OGSA-DAI. Through this mapping cross -organisational access can be configured.

OGSA-DAI also makes information about the schema of a data resource available. This information is published as Service Data Elements and the configuration of a Grid Data Service Factory requires that a callback class to retrieve this information be specified. This has not been investigated in any detail yet, but the idea is, to treat a CM data store as an XML Database. This makes sense as the query language for CM is XPath. It needs to be investigated to what extent this makes overall sense, after all

once II4C is taken into account a variety of data sources can be accessed from Content Manager and it might be sensible to define a Content Management specific way to expose the structure of a data store.

Activities for CM

Once a factory has been configured, suitable activities have to be determined and implemented. Activities have access to the *driverManagerImplementation* class mentioned above and can thus gain access to the data resource. This is achieved by the Engine calling a *setContext()* method on the activity before executing the *processBlock()* method repeatedly until no more output is produced and the activity closes its output object. The following UML diagram illustrates this.

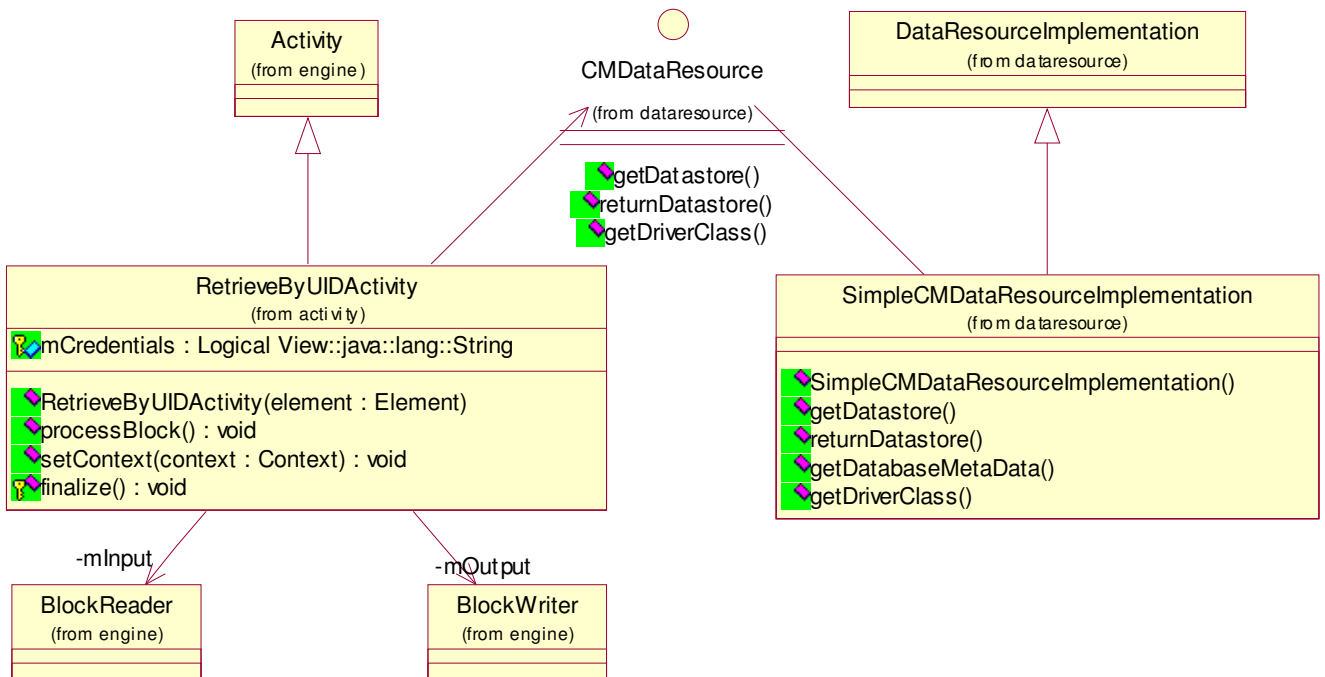


Figure 4 - UML for the RetrieveByUID Activity

A call to *getDastore()* returns an already connected data store, which after it has been used must be returned to the *DataResourceImplementation* via a call to *returnDastore()*. This design allows for data store pooling to be provided by the *DataResourceImplementation* class.

Four activities have been implemented so far to gain experience with OGSA-DAI. One is very specific to the eDiaMoND project and takes a query parametrised by a globally unique ID and searches the data store for Items that have a matching attribute. A URL to the associated content is created by CM for each match and returned as the response to the request. This activity (as defined by the perform document schema) also has a second implementation, which searches a federated data store. In that case the data store object with which the GDSF has to be configured has to be *DKDastoreFed* and the query expression passed to the federated data store has to be adapted, but the infrastructure already in place to support a CM data source is

sufficient to also support II4C. This activity can be made more generic by allowing an XPath expression to be passed in.

The next two activities are more generic and involve the creation of Attribute and ItemType definitions. These use CM tools to create CM Attributes and ItemTypes described by an XML document. The grammar of these documents are described by DTDs, however OGSA-DAI uses XML Schema to describe the grammar of conforming perform documents, hence the DTDs had to be transformed into XSDs in order to be usable inside a perform document.

Finally we have an activity that loads a DICOM file into CM. This activity utilizes the *datastore* and *deliverFromURL* activities that come as part of the OGSA-DAI distribution. The *datastore* activity is used to provide an XML file that contains meta-data about the file to be loaded and the *deliverFromURL* activity provides the file to be loaded as a ByteStream. The *loadDICOM* activity only contains the CM specific code to load data from this ByteStream into CM.

Experience

The experience so far shows that OGSA-DAI provided a great deal of flexibility in the granularity of activities that can be written. For a general purpose Grid enablement of Content Manager one would have to design a suitable set of activities that is fine grained enough to allow for flexibility, but also composable enough such that a series of interrelated actions can be specified in a single PerformDocument. Once the infrastructure for connecting to a CM data store was in place, the developers could concentrate on writing activities. Here the reuse of existing activities cut down development time and showed the flexibility of OGSA-DAI.

Future plans

There are a number of directions that we would like to investigate. So far the only data store object that we have used are DKDataStoreICM and DKDataStoreFed. II4C can access other data stores including relational databases. Which would lead to another way to expose a database on the Grid.

We would like to further investigate the different federation options. Because of the OGSA-DAI enablement, federation can be done at the Grid layer using Distributed Query Processing software also developed as part of the OGSA-DAI project. This is of particular interest to the eScience community in the UK. On the other hand II4C can also federate different data source and it will be interesting to compare and contrast the two approaches.

Acknowledgements

The authors wish to thank other members of the eDiaMoND consortium for their input into, and comments on, the work described in this paper, in particular the role of the clinical collaborators in determining the requirements for this project.

The authors wish to acknowledge the support provided by the funders of the eDiaMoND project: The UK Department of Trade and Industry, the EPSRC (ref no: GR/S20956/01) and IBM.

IBM and DB2 are trademarks of IBM Corp. Java is a trademark of Sun Microsystems Inc.

References

- [1] The eDiaMoND Project: www.ediamond.ox.ac.uk
- [2] J.M.Brady, D.J. Gavaghan, A.C.Simpson, M.M. Parada, and Ralph Highnam. EDiaMoND: A grid enabled federated database of annotated mammograms. In F. Berman, G.C. Fox, and A.J. Hey, editors, Grid Computing: Making the Global Infrastructure a Reality, pages 923-943. Wiley Series, 2003.
- [3] OGSA-DAI: www.ogsadai.org.uk/
- [4] Content Manager – <http://www.ibm.com/software/data/cm>
- [5] NHS Breast Screening Programme: www.cancerscreening.nhs.uk/breastscreen
- [6] P.B.Dean. Overview of breast cancer screening. In M.Do, M.L.Giger, R.M.Nishikawa, and R.A.Schmidt, editors, 3rd International Workshop on Digital Mammography, Volume 1119 of Excerpta Medica International Congress Series, pages 19-26. Elsevier Science, 1996.
- [7] DICOM standard – medical.nema.org/
- [8] CM Library: www.ibm.com/software/data/cm/cmgr/mp/library.html
- [9] GGF-Data Access and Integration Working Group: www.cs.man.ac.uk/grid-db
- [10] OGSA-DAI product overview: www.ogsadai.org.uk/docs/current/OGSA-DAI-USER-UG-PRODUCT-OVERVIEW.pdf
- [11] OGSA-DAI activity guide: www.ogsadai.org.uk/docs/current/OGSA-DAI-USER-UG-ACTIVITY.pdf
- [12] OGSA-DAI under the hood www.ibm.com/developerworks/library/gr-ogsadai