Health-e-Child: a grid platform for european paediatrics

Health-e-Child: A Grid Platform for European Paediatrics

K. Skaburskas[1], F. Estrella[1], J. Shade[1], D. Manset[2], J. Revillard[2], A. Rios[2], A. Anjum[3], A. Branson[3], P. Bloodsworth[3], T. Hauer[3], R. McClatchey[3], D. Rogulin[3] on behalf of the Health-e-Child consortium.

[1]CERN, Geneva, Switzerland,

[2]Maat-G Knowledge, Toledo, Spain

[3]Centre for Complex Cooperative Systems, UWE-Bristol, UK

Corresponding Author : Richard.McClatchey@uwe.ac.uk

Abstract. The Health-e-Child (HeC) project [1], [2] is an EC Framework Programme 6 Integrated Project that aims to develop a grid-based integrated healthcare platform for paediatrics. Using this platform biomedical informaticians will integrate heterogeneous data and perform epidemiological studies across Europe. The resulting Grid enabled biomedical information platform will be supported by robust search, optimization and matching techniques for information collected in hospitals across Europe. In particular, paediatricians will be provided with decision support, knowledge discovery and disease modelling applications that will access data in hospitals in the UK, Italy and France, integrated via the Grid. For economy of scale, reusability, extensibility, and maintainability, HeC is being developed on top of an EGEE/gLite [3] based infrastructure that provides all the common data and computation management services required by the applications. This paper discusses some of the major challenges in bio-medical data integration and indicates how these will be resolved in the HeC system. HeC is presented as an example of how computer science (and, in particular Grid infrastructures) originating from high energy physics can be adapted for use by biomedical informaticians to deliver tangible real-world benefits.

1. Introduction

The Health-e-Child (HeC) project aims to develop a prototype system which will demonstrate the integration of heterogeneous biomedical data sources over a grid linking multiple hospitals across Europe. In this integration, particular emphasis is put on distinguishing features such as universality of information, person-centricity of information and universality of application leading to the main tenet of the HeC effort: "the integration of information across biomedical abstractions, whereby all layers of biomedical information (i.e. genetic, cell, tissue, organ, individual and population layer) are vertically integrated to provide a unified view of a child's biomedical and clinical condition" [2]. One essential element required for the integration of data across the layers of biomedical information is the provision of suitable models for data and information.

The HeC system is providing a demonstrator of an integrated healthcare platform based on three main pillars: the modelling and integration of relevant biomedical sources, a service-oriented environment to manage distributed data and knowledge sources and the use of integrated decision support and knowledge discovery systems. The main of focus is the integration along the vertical domains (so-called "vertical integration"), which refers to the span across the medical spectrum of information: from molecular to cellular, tissue, organ, individual and population. In clinical diagnosis and treatment, the patient assessment is based on sensors related to one or more of these levels, and ultimately they all contribute to the overall picture of the patient's health.

One of the major cornerstones supporting the HeC project goals is the modelling of relevant biomedical data sources. The biomedical information that is managed by HeC spans multiple vertical ranges, comes from different data sources and is possibly distributed and heterogeneous with various levels of semantic content. HeC aims to create a set of models which facilitates the integration of all the available information that supports HeC system components, by providing access to the appropriate information between hospitals and that supports the integration across vertical levels of the medical domain.

To be able to combine all sources of data into the integrated view the model of the domain under consideration needs to be established. Such an integrated model must provide clinicians with a coherent view of patients' health and be adaptable to changes in the models of individual sources. Some of the criteria which HeC domain models should satisfy include:

- capturing information specified in clinical protocols;
- supporting high-level applications such as integrated disease modelling, decision support and knowledge discovery;
- forming the basis of data management in the HeC platform and supporting the clinical queries that are expected to appear in the Health-e-Child use-cases;
- being flexible, extendible and able to evolve.

The proposed solution to tackle the problems mentioned above is to build a comprehensive integrated data model based on the available clinical protocols, clinical workflows (e.g. patient journey), expert annotations (e.g. assessment of the overall status of the patient or the disease progression, interpretation of the raw data etc.) and external clinical knowledge (e.g. well-established associations between different kinds of data in biomedical domain). Such a model will provide the foundation for the flexible access to and sophisticated analysis of the integrated data (e.g. statistical, temporal analysis, similarity searches, knowledge discovery etc).

2. From Requirements to Models

As a first step in requirements analysis we identified the relevant data and information sources that provide input for the modelling process. These included data acquisition protocols, focused interviews with the domain experts and external data and knowledge resources. The following major characteristics have been identified:

- The protocols provide a static view of the data being collected by clinicians in different departments representing atomic/distinct "facts" about a patient without any interpretations of this data (e.g. the "meaning" of a particular measurement) or semantic relationships to the data acquisition process or to the intended usage of the data (e.g. the usage of the data to produce a diagnosis etc.).
- Clinicians collect a considerable quantity of heterogeneous data, from highly structured (e.g. patient's baseline information, examinations etc.) to very unstructured (e.g. images, free-text etc.).
- Clinical protocols allow identifying the relevant modelling concepts in each medical sub-domain but do not capture the semantics of the relationships between these concepts explicitly (i.e. expert interpretation is needed).

The interviews with the clinicians have been conducted to focus on the clinical workflows (e.g. the patient 'journey'), and the data acquisition and usage with respect to the different tasks in defined clinical workflows (e.g. data collection and analysis, diagnosing, suggesting treatment and evaluating its outcome etc.). Not surprisingly, much implicit knowledge about the decision making in clinical practice is based on the personal experience of a clinician. A non-medical expert cannot usually decide about the relevance and importance of the selected data. At the same time, the clinicians are experts in a particular domain (e.g. radiology, genetics etc.) and thus have a narrow view of the patient's health, i.e. from a very particular point of view. Sometimes it is rather difficult for them to go through all reports of all investigations from different departments (e.g. imaging, blood tests etc.) – rather they would prefer to have an integrated summary/view of the relevant information. Also, clinicians cannot be expected to formulate sophisticated queries against a particular model (e.g. the model defined by the protocols) – they would rather formulate their analytical questions in terms of the vocabulary they are used to.

The complexity in modelling the Health-e-Child domain arises from diverse orthogonal aspects:

1. Different medical domains, i.e. rheumatology, cardiology, neuro-oncology, radiology, genomics etc. Data being collected in different departments are of different modalities, formats, structurally and semantically diverse. The models should therefore facilitate the horizontal (across different departments, hospitals, countries etc) as well as vertical (across different levels of granularity) integration. The overlapping domains have to be identified and seamlessly integrated, capturing all details from clinical protocols to establishing a common model of the patients' data.
2. Combining anatomy/physiology models (i.e. normal state and functioning of the body) with the disease (i.e. pathology) model. The semantics of different aspects of "normal" and "abnormal" behaviour and the correlations between different pieces of information at different levels need to be captured.
3. Capturing functional aspects of the sub-processes of the patient journey. The subtasks in the clinical workflows (e.g. diagnosis, treatment, follow-up etc.) may be similar in different departments sharing the same goals but the composition of these tasks as well as the strategy of realizing the tasks' goals might differ. The challenge is to identify reusable "process patterns" that can be formalized using task templates (e.g. as suggested by the CommonKADS [4] methodology) describing, for example, diagnosis or treatment procedures.
4. Temporal dimension. The clinical process usually follows a given time order (symptoms, study, diagnosis, treatment, follow-up, etc.), the time relates to some symptom/diagnosis and treatment concepts, time is a most important attribute for disease progression, and prognosis, and finally the temporal axis is apparent in the development of the individual.. Finally, the paediatrics domain adds an additional complexity due to the fact that the child is growing and the observations in time should be aligned with the anatomical changes as well as the knowledge about how a particular disease may affect these changes. Thus, the model must cater for different time representations and aspects.
5. Spatial (Horizontal) integration. In clinical practice, medical data about a patient is generated at different locations, such as different departments at a hospital or at different hospitals. The patient journey from the first symptoms through diagnosis to treatment and follow-up might span multiple locations which poses several problems in maintaining the entirety of patient data. In practice, doctors often are confined to information that is produced locally; comparing similar cases or reviewing rare, complex or interesting ones is usually based on local experience. Integration along the spatial (horizontal) axis means more uniform access to possibly remote data.
6. Vertical integration. The information from different levels of granularity needs to be cohesively integrated to provide an integrated view of the data to the clinicians. We distinguish six levels: molecular, cellular, tissue, organ, individual and population. For example, in rheumatology the following data available at different levels can be identified:

   Molecular: genomics and proteomics data
   Cellular: results of blood tests
   Tissue: results of blood, synovial fluid tests
   Organ: bones (deformation etc.) and joints (swelling, pain, limitation of motions etc.) assessment, bones and joints measures (shape, size, bone age etc.)
   Body: lifestyle, movement assessment, damage index, rheumatology examinations
   Population: epidemiological studies based on different criteria (country of origin, sex, etc.).

The semantic link between these levels is obvious: entities are in part-of hierarchies, but the complexity of the human body and processes usually does not allow for drawing straightforward conclusions from parts to the whole. The integrated model should provide quick access to the relevance-based, time-oriented and concept-oriented, configurable views of the data at different levels. In addition, vertically integrated data need to be made available not only for human interpretation but also for (semi-)automated processes such as the Health-e-Child enabling tools. To facilitate this kind of integration, knowledge representation techniques are being investigated and applied elsewhere in the project

7. The linkage and usage of the external medical knowledge. The main goal here is to establish a basis for the semantic coherence of the integrated data and provide mappings from clinical data to the external medical knowledge (e.g. biomedical ontologies and databases) to facilitate availability and accessibility of external information for querying and analysis by clinicians and applications (establishing the ground truth) as well as to make the knowledge acquired in the project available outside (sharing the knowledge).

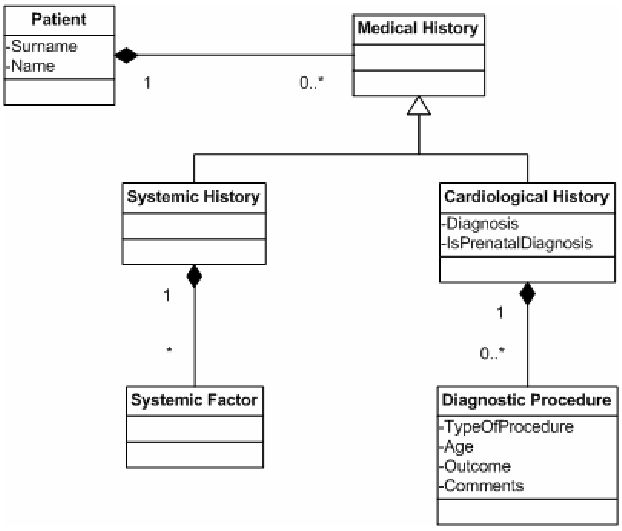

Figure 1. An extracted fragment of the HeC data model

As an example of the detail captured in the HeC data model Figure 1 shows a fragment of a patient's medical history as derived from a study of cardiology protocols.

3. Semantics and the Health-e-Child Data Model

One crucial factor in the creation of integrated heterogeneous systems dealing with changing requirements is the suitability of the underlying technology to allow the evolution of the system. A 'reflective' system utilizes an architecture where implicit system descriptions are instantiated to become explicit so-called "metadata objects". These implicit system aspects are often fundamental structures and their instantiation as metadata objects serves as the basis for handling changes and extensions to the system, making it somewhat self-describing. Metadata objects are the self-representations of the system describing how its internal elements can be accessed and manipulated. The ability to dynamically augment and re-define system specifications can result in a considerable improvement in flexibility. This leads to dynamically modifiable systems which can adapt and cope with evolving requirements. In this way we can separate the system description in terms of metadata from the particular physical representations of the data and thereby promote ease of integration and querying of the data whilst retaining the ability for the semantics of the system to evolve.

The complexity which arises from the use of diverse distributed data sources in Health-e-Child and the anticipated evolution of its medical information led us to the decision to adopt a modelling approach which heavily relies on metadata. In addition the model is enhanced with a semantic layer to facilitate the semantic coherence of the integrated data and to allow linking and reuse of the external medical knowledge.

The metadata reveals the structure of the underlying heterogeneous medical data allowing consistent queries across populations of patients and disease types. The semantic layer adds knowledge to this metadata thereby facilitating the resolution of queries that bridge between related concepts. It is this combination of descriptive metadata with system semantics that provides the Health-e-Child data model with the ability to be both reactive in terms of the queries generated by user applications and to have the richness to enable integration across heterogeneous data sources. The data model overview is presented in Figure 2.

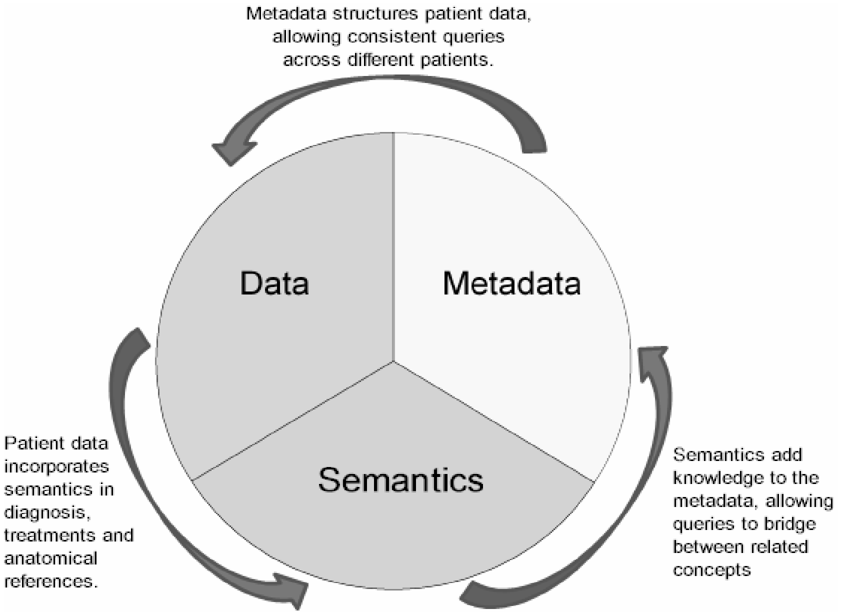
Metadata structures patient data, allowing consistent queries across different patients.
Data
Metadata
Semantics
Patient data incorporates semantics in diagnosis, treatments and anatomical references.
Semantics add knowledge to the metadata, allowing queries to bridge between related concepts

Figure 2. A high-level overview of the Health-e-Child data model

There are many noted advantages of the use of meta-data – it can improve system interaction and data quality, it can support system and domain integration, and it can enhance system maintenance, analysis and design. In HeC the metadata structures define and describe precisely what data can be stored and how it can be accessed and this allows the evolution of the domain-specific data model without the need for re-engineering the way in which the data is actually stored or accessed. In other words the metadata describes the data structures in the system but does not address how the stored information can be interpreted or how the meaning of (a subset of) the data can be extracted to allow inference of new knowledge, potentially hidden in the data. This interpretation and inference is carried out using the semantic layer which allows information to be integrated or aligned with external data sources or knowledge bases thus permitting knowledge reuse as well as making the knowledge available outside of the project [5]. Consequently, the semantics associated with the data needs to be captured to facilitate the use of integrated information.

The HeC project will develop procedures for semantic query enhancement and optimization using a HeC ontology [6]. The query answering process will combine these features to: enable semantic query reformulation, enhance answer sets and improve support for approximate queries. The addition of semantic structures will add flexibility and descriptive power to the integrated data model. This requires the creation of a suitable knowledge representation strategy and involves the investigation and use of established and emerging ontology engineering techniques such as ontology modularization, segmentation, specification and validation, and these and other methods may all prove useful in achieving this.

We shall demonstrate the power of knowledge representation in the Health-e-Child context through a set of targeted use-cases. The aforementioned semantic query enhancement will improve query answering with reasoning capabilities and similarity searches. Linkage to external knowledge sources will enrich the data presented about a patient and will enable classification of patients according to various flexible criteria. Finally, application-driven ontology engineering techniques will be the means to investigate how the semantic content can improve the results of high-level applications such as decision support systems.The full detail of the HeC integrated data model is beyond the scope of this short paper. Interested readers are directed to the HeC project web-site for a full description of the model, on how it is being implemented in the project and on how ontologies are being used to assist the clinician in semantically linking heterogeneous data elements of diverse information sources.

4. The Health-e-Child Grid Platform Requirements

Realization of the project goals (Biomedical vertical integration of data, information sharing, query processing, knowledge discovery, decision support etc.), requires an infrastructure that is highly dependable and reliable. Physicians may require guarantees that the system is always available and that the processes that integrate data on patients are reliable, even in the case of failures. Moreover, the infrastructure may have to allow for the transparent access to distributed data, to provide a high degree of scalability, and to efficiently schedule access to computationally intensive services by applying sophisticated load-balancing strategies. Physicians may need information immediately in order to make vital decisions. Hence, long response times due to a high system load are often not well-tolerated. Consider a case where a similarity search across a potentially large set of documents is needed. In order to support this search, feature extraction has to take place for all documents/images, nearest neighbours have to be determined, etc. All these steps require significant computing power and should not be limited to the organization where the images are stored. Rather, additional feature extraction services should be installed automatically at hosts that currently feature a low load.

The Health-e-Child project has compiled a set of user requirements after talking to the clinicians and hospitals. These requirements were kept in mind while designing and creating the Health-e-Child Platform, described elsewhere [7]. According to these requirements, the system shall provide a distributed computing infrastructure for sharing and exchanging biomedical information and knowledge which will serve as a backbone for the Health-e-Child enabling applications. The system will use gLite for virtualizing distributed computing resources and enabling secure access to sensitive medical data. The Grid middleware shall provide secure, coordinated and controlled access to the distributed computing resources. The Grid middleware shall facilitate the creation/removal/modification of Virtual Organizations (VOs) to allow co-working between clinicians. It shall be possible to manage Grid infrastructure allowing addition, removal and modification of nodes on the grid. Furthermore, according to the requirements, a "HeC Gateway" will be created that will provide a suitable access strategy to the grid. The HeC Gateway shall provide access to the functionalities of the underlying grid middleware for the higher-level components of HeC, and will provide a general API for access to the grid. The API shall expose grid functionalities for the HeC services and allow managing and monitoring processes running on the grid; the API shall also allow the execution of jobs on the grid, and the monitoring and analysis of the results of the jobs.

The main Grid infrastructures objectives of HeC are:

- to provide a gLite-based infrastructure for data integration development and testing.
- to develop appropriate grid interface(s) to gLite services for the HeC Gateway components and client applications.
- to provide training for the HeC Gateway and client application developers in the use of gLite services and
- to ensure that updates and releases of gLite are available and applied on the HeC infrastructure for the duration of the project.

To support the HeC software development cycles, trial deployments, and evaluation and experimentation with the HeC-specific gLite middleware configurations, three sub-infrastructures PDI (Platform Development and Integration), PPROD (Pre-Production) and PROD (Production) were identified (as shown in figure 3). The main goal of the gLite deployment was to expose the grid middleware functionalities within the HeC platform. We decided to build our own PDI infrastructure to better understand how all the gLite components interact and also to train those responsible for the actual deployment of the software in the participating institutions and hospitals - the HeC PPROD and PROD infrastructures. It also allowed us to deploy and experiment with the gLite releases and other software components we needed to support the HeC Gateway development with the project-specific functionality. We decided to mimic as much as possible real PPROD and PROD sites on our testbed.

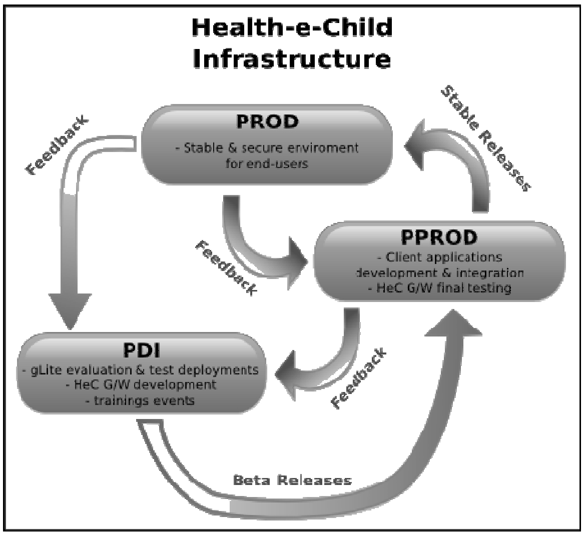

Figure 3. An overview of the Health-e-Child Grid infrastructures

A set of APIs acting as a grid interface to gLite services for the HeC Gateway components [7] and client applications was developed. The development of the API was carried out keeping in mind the idea of its exposure directly as a web service if required in the future.

5. The Health-e-Child Grid Infrastructure

To support HeC software development cycles, experimentation with HeC specific gLite middleware deployments, developers' and infrastructure administrators' trainings, three sub-infrastructures were identified (see Figure 3):

1. PDI (Platform Development and Integration)
   1.1. evaluation of new gLite releases
   1.2. experimentation with gLite deployment strategies
   1.3. development and early testing of HeC Gateway
   1.4. training of HeC developers and infrastructure administrators
2. PPROD (Pre-Production)
   2.1. medical client applications development, integration with HeC Gateway and possible gridification
   2.2. HeC Gateway final testing, before deployment in Production
3.PROD (Production) – a stable and secure environment for end-users
   3.1. HeC Platform deployed and operational among hospitals participating in the project

PDI, PPROD and PROD are three distinct non-overlapping sub-infrastructures forming the HeC Infrastructure. Figure 3 schematically represents them and also shows Grid middleware evaluation, validation, deployment and software development cycles on the HeC Infrastructure. The first stage deployment on the HeC grid infrastructure follows a centralized deployment. It was decided that this deployment must be tried out on PDI and then implemented on PPROD and PROD. In such a deployment scenario there is one site hosting the gLite core services (top-level BDII, WMS/LB, central LFC, AMGA) and it must be the responsibility of one of the HeC partners. Other sites are simply hosting SEs, CEs and site-level ISes, as is standard practice in gLite middleware deployment.

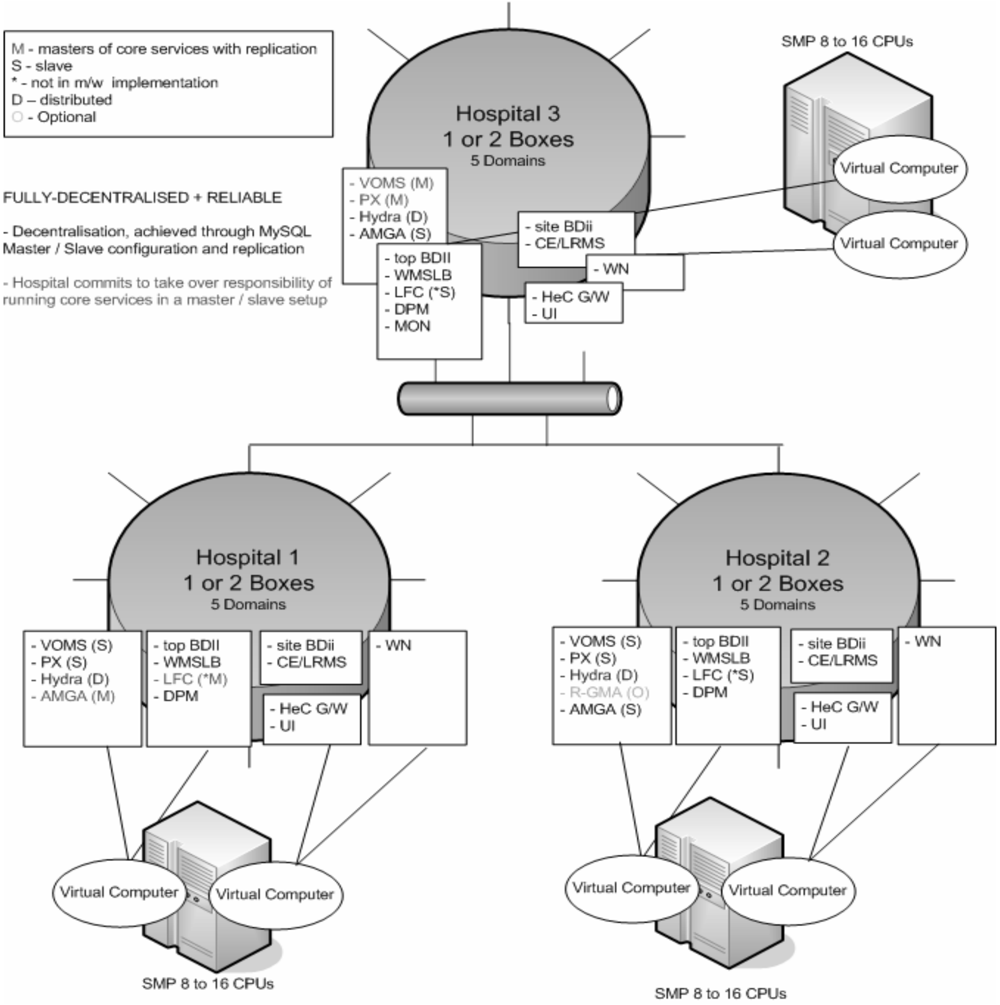

Figure 4 .The HeC architecture for decentralized system deployment.

The second stage deployment on the HeC grid infrastructure follows a decentralized deployment. Before deployment the system must be very well tested on PDI and then implemented on PPROD and PROD. With this deployment scheme we would like to achieve a certain level of decentralization and autonomy for participating sites with respect to the deployment of grid core services where we eliminate the need for a single site hosting the services in a high availability setup for the whole infrastructure. Evaluation of the gLite middleware services stack shows that this approach is feasible, however not trivial and may require certain development. The main mechanisms to achieve the goal would be the deployment of the services in a master/slave setup (with the possibility of takeover) using replications of gLite services' backends and the redundant deployment of services with proper configurations of clients. Figure 4 schematically shows the anticipated deployment architecture to support autonomy of sites.

6. The gLite Integration Scheme

6.1. Integration Objectives

The main goal of the gLite integration was to expose the grid middleware functionalities within the HeC platform. To do that, different possibilities were studied and in order to select the best one (for the HeC community), three main objectives were identified:

- To hide as much as possible the complexity of the grid middleware. This complexity is mainly due to the power of the middleware itself in term of functionalities.
- To deliver as quickly as possible a first version so that the HeC community can start to integrate grid functionalities in their applications/services.
- To make the grid functionalities easy to use: this solution must not increase the amount of work that has to be done by the users of the grid middleware.

The two first points required that only basic functionality was to be exposed initially. As a consequence, this integration has to be done in an extensible way in order to be able to easily extend the basic exposed functionality with others more complex, if required.

6.2. Integration Scheme Alternatives

Three different integration schemes have been identified. The first integration scheme consists in compiling and integrating the gLite UI or gLite APIs at the client level (i.e. the desktop application interacting with the HeC platform). The main advantage of this scheme is performance since there is no layer between the client application and the grid functionalities - but it also induces several problems. For instance, this approach does not satisfy the HeC security requirements [8]. Indeed, in [8], it is explicitly stated that a user must have generated a VOMS (Virtual Organization Management System) extended proxy certificate to connect to the HeC platform [9]. It is currently possible to use some gLite services without the VOMS extensions in the proxy so that the gLite grid middleware itself does not address this requirement. To address it in this configuration, each client application developer would have to verify it in the client application source code itself and this can induce some non-functional constraints which are not desirable. Moreover, this integration scheme exposes all the complexities of the grid which is not the best approach when encouraging client developers to use the grid functionality. Finally, the use of this architecture obliges the client application developers to use specific languages such as C/C++ or python as gLite does not provide APIs for all the languages that can be used by the clients.

The second integration scheme consists of wrapping-up gLite APIs in services (where these services are components of the HeC platform). This approach gives the possibility of fine tuning the gLite functionalities in the sense that the platform developers can expose only the subset of functionalities they want the client application developers to use. Another advantage is that most of the gLite complexity is hidden to the client application developers. Despite all these positive points, this integration scheme also has some disadvantages. Firstly, it induces more developments server side. Secondly, an overhead is introduced as grid functionalities are wrapped into services. Finally the gLite community does not currently provide all the required APIs. In fact, the services are written in Java and a few parts of the grid middleware have Java APIs. The SWIG software development tool [10] was tested to encapsulate some of the C/C++ native gLite APIs in some Java source code, but there was a notable decrease in performance.

The last integration scheme consists of creating a Java API which wraps the gLite UI CLIs. This integration scheme is not perfect since it, like the previous one, induces an overhead compared to the first solution. Nevertheless, the we have decided to go for this solution as it has a lot of advantages:

- This approach is highly modular and extensible: only the required CLIs can be wrapped-up in the first version of the API to deliver the basic grid functionalities. Then, if the platform developers ask for more functionality, it can be extended very easily.
- This solution implies that only the middleware developers have to know in detail of he complexity of the grid middleware. The created API will act as a first abstraction layer for the platform developers. Well documented, they will be able to use it like a Java API.
- This API can be designed thinking of the possibility to expose it as a web-service in the future if needed. This approach is completely in-line with the platform design (following a Service Oriented Architecture, SOA, approach) [11].

In parallel to the above activities, much work has been carried out to acquire knowledge about the deployment of gLite. As the CERN HeC representative is part of EGEE Operations Activity, there have been many technical interactions between HeC project and operations, and the certification and middleware re-engineering activities of EGEE. This facilitated direct gLite grid middleware knowledge transfer from EGEE to HeC. Also, feedback from HeC to EGEE can be identified by the following two activities: i) the feedback to EGEE project in terms of extensive usage and testing of AMGA [12] (The HeC project developers are in close contact with AMGA developers and actively participating in testing of AMGA (decentralized deployment and feature testing)) and ii) some very precise bug reports were also submitted by the HeC middleware developers to EGEE on several occasions as we started to deploy our HeC testbed mainly with pre-production gLite releases. Now we deploy production gLite releases on the HeC testbed and in case of found problems immediately provide feedback to the EGEE/gLite team.

7. Conclusions

The HeC project has been running since January 2006 and will conclude its study until December 2009. The project involves paediatric hospitals in the UK, France and Italy working closely with academics (e.g. UWE, the University of Athens, DISI-Genoa, INRIA and CERN) and with commercial partners (e.g. Siemens, Maat-G) to achieve common goals in the EC FP6 Integrated Project. Already the project has been able to develop complex data models to describe biomedical data from Juvenile Idiopathic Arthritis, Brain Tumours and Cardiology domains. It is working to provide semantic integration of biomedical data to enable clinicians to seamlessly access data across a gLite-based Grid infrastructure based in London, Paris and Genoa. Other hospitals in Italy, Saudi Arabia and the US have expressed interest in extending this international infrastructure.

Work is continuing in the areas of data access and management, in the use of ontologies, in the development of suitable decision support, disease modelling and knowledge discovery applications and in the area of data security and privacy. The project is developing a set of definitive paediatric clinical use-cases for the validation and testing of the applications and infrastructure during the final months of the project and later papers will report on the outcome of those tests.

References

[1] The Information Societies Technology Project: Health-e-Child EU Contract IST-2004-027749 Description of Work.

[2] J. Freund, et al. “Health-e-Child: An Integrated Biomedical Platform for Grid-Based Pediatrics”, Studies in Health Technology & Informatics #120, pp 259-270 IOS Press 2006

[3] EGEE website - http://www.eu-egee.org/ and gLite Grid middleware website - http://www.glite.org/

[4] G. Schreiber et al., “Knowledge Engineering and Management: The CommonKADS Methodology”, MIT Press, Cambridge, Mass., 1999.

[5] N. F. Noy. “Semantic Integration: A Survey of Ontology Based Approaches” ACM SIGMOD Record, Special Issue on Semantic integration, 2004.

[6] D. Fensel., "Ontologies: A Silver Bullet for Knowledge Management & Electronic Commerce”. Springer-Verlag publishers 2000.

[7] Health-e-Child deliverable D7.1 “The Health-e-Child Gateway Design specifications and developers guide”

[8] Health-e-Child WP4 deliverable D4.2 (Internal)

[9] I. Foster, C. Kesselman and S. Tuecke, The Anatomy of the Grid, International Journal of High performance Computing Applications, 15, 3, 2001

[10] SWIG home page - http://www.swig.org/

[11] C. M. MacKenzie et al. Reference Model for Service Oriented Architecture 1.0, OASIS Committee Specification 1, 2 August 2006

[12] AMGA – the ARDA Metadata Catalogue Project http://amga.web.cern.ch/amga/