Data Aggregation and Analysis: A Grid-based approach for Medicine and Biology

Dimosthenis Kyriazis, Konstantinos Tserpes, George Kousiouris, Andreas Menychtas, Gregory Katsaros, Theodora Varvarigou
School of Electrical and Computer Engineering, National Technical University of Athens
9 Heroon Polytechneiou Str. 15773, Zografou, Athens, Greece
{dkyr, tserpes, gkousiou, a_menychtas, gkats, dora}@telecom.ntua.gr

Abstract

A constantly increasing number of applications from various scientific sectors are finding their way towards adopting Grid technologies in order to take advantage of their capabilities: the advent of Grid environments made feasible the solution of computational intensive problems in a reliable and cost-effective way. In this paper we present a Grid-based approach for aggregation of data that are obtained from various sources (e.g. cameras, sensors) and their analysis with the use of genetic algorithms. By also taking into consideration general historical data and patient-specific medical information, we present the realization of the proposed approach with an application scenario for personalized healthcare and medicine.

1. Introduction

Grid computing has emerged in the last years as a technology for large-scale, flexible and coordinated resource sharing and is increasingly considered as an infrastructure able to provide distributed and heterogeneous resources in order to deliver in a transparent way computational power to resource demanding applications [1]. Built on pervasive internet standards, Grids support the sharing, interconnection and use of diverse resources, integrated in the framework of a dynamic computing system, in a secure and highly efficient manner. Furthermore, a Grid-based environment enables the storage and distribution of data allowing access to various sources and analysis of them. Therefore, the information contained for example in medical records can be accessed and analyzed for various reasons (e.g. selection of the best treatment and prediction of its outcomes).

Exploitation of Grid technologies is imperative for medical applications due to a set of reasons such as the exponential increase of the required storage and computational resources, the heterogeneity of the required data (medical records, images, information obtained from sensors) with different preprocessing requirements and the large number of involved patients.

Medical-related applications generally belong to those collaborative environments that are based on input from networked sensors and aggregation of acquired data under real-time conditions. With the simultaneous advent of technologies to support heterogeneous sources of information and computing resources (through Service Oriented Architectures, Grids, etc), it is expected that in the following years to come, there will be a great blooming in the development of infrastructures comprising multiplicity in sensors both in number and nature. Ubiquity renders data management issues of major importance since the nature of data will be changed in order to include high quality input from sensors of all kinds, assuring privacy and aggregation associated and cross-checked with incomplete and inconsistent information. In this frame, any new achievements and directions in nanotechnologies, networks and biosensors are expected to achieve performance and interconnectedness.

To this end, a significant part of the value of Grid technology lies on the fact that Grids are in position to provide the fundamental management mechanisms for distributed data. This is one major reason that often many developed Grid-based systems were referenced as “data Grids”, since the integration of data, infrastructures, digital libraries and persistent archives was a challenge forcing continued evolution of Grid technology. This challenge remains valid for medical applications, the requirements of which range from the transition from data handling, sharing and aggregation to the provision of knowledge as utility.

Data virtualization and master data management impacts Grid-based environments in several ways, since it presents information as a service and promises to provide a single access point to manage and view data regardless of data source or physical location. Thus efficient and advanced data and information management techniques are one of the key factors associated with the agility of Grids. Taking advantage of this feature, Grids can be used for aggregating data formats coming from a variety of sources, such as sensors and medical archives into common public data formats that can be further analyzed by many numerous algorithms and mechanisms.

Besides the data aggregation, we also present an innovative approach for data analysis and simulation of possible therapeutic schemes (or as called "*Personalized Medicine Proposals*"). As the number of possible therapeutic schemes and consequently the number of simulations increases, the time required for evaluating and comparing the effects of the different schemes may become forbiddingly high. Exploiting Grid computing is a very attractive solution, as the resources provided in a Grid infrastructure may be efficiently used to reduce the overall required time for simulations.

The remainder of the paper is as follows: Section 2 presents related work in the field of Grid computing for biomedical applications. The main topic of our study is described thereafter in Section 3 while Section 4 discusses an application scenario for the realization of the proposed approach. The aforementioned scenario refers to personalized healthcare and medicine. Finally, Section 5 concludes with a discussion on future research and potentials for the current study.

2. Related Work

In general, enabling applications execution on Grid environment has been a research topic since the distributed nature of a Grid-based infrastructure makes feasible the solution of computational intensive problems in a reliable and cost-effective way. To this direction, literatures [19], [20] and [21] present the work performed for various application domains (biocomputational, learning and medical). Moreover, Parameter Sweep Applications (PSAs) are a class of applications that deal with the analysis of a specific simulation for a range of parameter values. PSAs are a very common class of applications that are met in computational Grids. High performance parametric modeling has been identified as a killer application for Grids [3] and Grid-enabled PSAs have been recently developed in Bioinformatics [4]. Following this direction, there are many projects dealing with the Grid technology in the Biology and Medicine sectors. The BRIDGES project (Biomedical Research Informatics Delivered by Grid Enabled Services) [9] aimed at developing Grid-enabled bioinformatics tools to support biomedical research. WISDOM initiative (Wide In Silico Docking On Malaria) [10] aims to demonstrate the relevance and the impact of the Grid approach to address drug discovery for neglected and emergent diseases with the use of the EGEE infrastructure [2], while, the Akogrimo Project [11] specified a mobile Grid infrastructure and evaluated it through a heart monitoring and emergency management scenario with the use of mobile devices. The Grid Relational Catalog (GRelC) project is working towards ubiquitous, integrated, seamless and comprehensive data Grid management solutions to fully address application specific requirements. Such an environment was used in the bioinformatics sector as described in [12].

Furthermore, interesting works in the area of Grid computing with regard to biology are described in [13] and [14]. Authors in [13] discuss a Grid-based infrastructure that performs ingestion into a relational DBMS for data integration of biological data sources, which reduces the data redundancy of biological flat files. Literature [14] describes an approach for using Grid resources are used for access to medical image repositories, segmentation services, simulation of blood flow, and visualization in virtual environments of the simulated results.

Specific approaches for Grid services, such as an architecture for data management in Grids focused on the bioinformatics domain is presented in [16], while [15] includes a workflow management system description that supports software components for image import/export, caching, processing and notification. In that context, works on data aggregation are presented in [17] and [18].

The work presented in this paper advances the field of research in applying Grid technologies to biology and medicine since most of the current approaches refer to the use of Grid services for computation or data management purposes but do not address other pertinent issues such as data aggregation and analysis from various sources.

3. Grid-based Approach

3.1. Architecture

In the following figure (Figure 1) we present the component model of the proposed architecture. The main components are the *Data Aggregator* and the *Analyzer* that interact continuously in order to produce the *Personalized Medicine Proposal*.

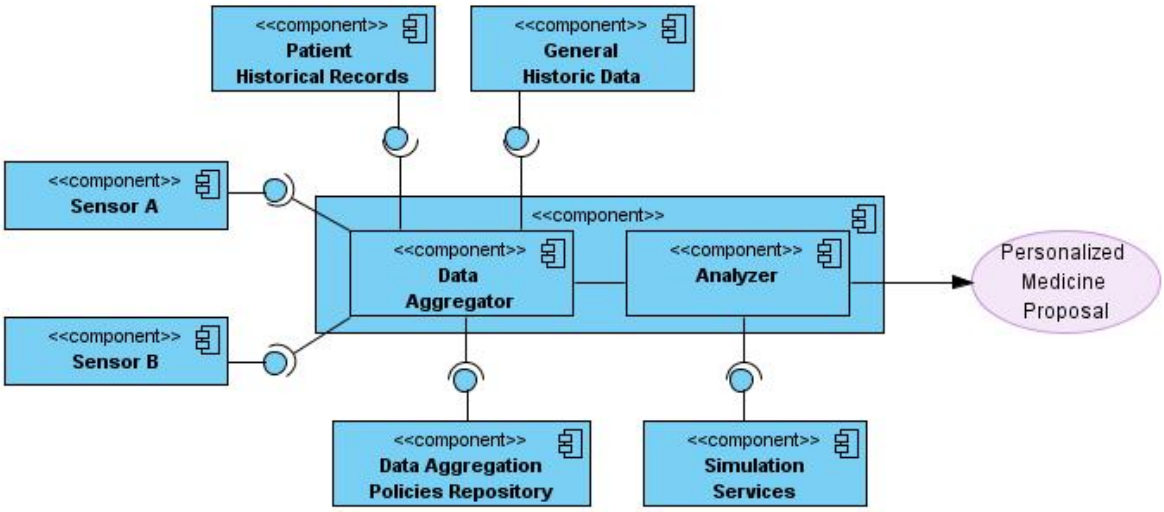

Figure 1: Component Model

The *Data Aggregator* is a component that dynamically changes its functionality based on a set of policies that are acquired by the *Data Aggregation Policies Repository*. These policies, which are associated with specific medical treatments or events from sensors, alter the data stream sources for the Data Aggregator and at the same time apply the algorithms that will be used on the streams so as to generate a set of patient specific information and relay it to the Analyzer. The data streams are separated in two categories, the real-time data from the sensors and the historical data either the patient and or for the similar medical conditions and statistics.

The *Analyzer* component exploits this set of information and performs an analysis in order to produce the personalized healthcare and medicine proposal as described in Section 4. The analysis process is also customized based on the real-time and historical data but in all cases includes a simulation of the available treatments and its parameters. For the simulation process, dedicated services are used that combine the mass data and computational capabilities of Grid environments. During the analysis process, the simulation results are evaluated and if they are not satisfying in terms of conformity with the historical data and statistical models, additional information are requested from the Data Aggregator and the simulation process for the particular treatment starts again.

In the following paragraphs, we describe in detail the aforementioned core components of the proposed approach.

3.2. Data Aggregator

The Data Aggregator component is able to support the aggregation of data deriving from heterogeneous resources in a time sensitive manner. Such resources include biosensors, actuators and digital inventories and knowledge libraries (e.g. pharmaceutical inventories or diagnosis databases). The challenging part in this design, is not the collection of the data as a bytestream but the parameterization of the Data Aggregator so as to "comprehend" the information and make sure the critical information is transmitted to the end nodes interested in the data (e.g. Data Analyzer). Roughly speaking, this implies that in order to resolve a query, the combinatory use of various pieces of information is needed. These pieces are defined by the application developer along with the appropriate way to present the results (e.g. store it, post it on the web, push it to another component).

For the purposes of the design we adopted the concept of an event-driven Service Oriented Architecture. Messages that are propagated by input resources, e.g. sensors, cameras, etc, are received by dedicated Information Management Services (IMS) that process them per category. This is illustrated in Figure 2 where the information flow is depicted between the input resource categories and the Data Aggregator with the mediation of the respective IMSs. The result of this process is a report that contains all the necessary information that needs to be relayed: the sender's ID, the receiver's ID, lifecycle information (e.g. a timestamp) and the message itself. These reports are generated and "wrapped" in the form of an event message, following the WS-Eventing specification of the W3C standardization body. Once the Data Aggregator receives an Event, the respective policy is triggered (see Figure 1) and the information contained in the report is treated accordingly. This means that the information is stored to an intermediary repository and

consequently combined with other pieces of information generally defined by the policy.

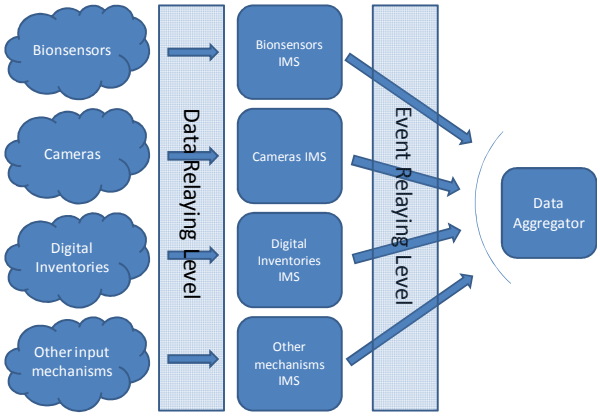

Figure 2: Overview of the information flow between groups of input mechanisms and their respected Information Management Services to the Data Integrator component

Then the Data Aggregator sends the information extracted by the aforementioned process to the components that it is configured to. In our paradigm, the component is the Data Analyzer. Again, the information is wrapped up in a WS-Event message so as to trigger the Data Analyzer's policies in order to reach to a final meaningful conclusion regarding the patient. In an implementation level, this architecture is based on Service Oriented Computing concepts, inheriting the OGSA [22] and Web Services Architecture principles through standards and specifications such as WS-Eventing (each component identified in the overall design is exposed as a Web Service).

In conclusion, a similar architecture for a service oriented data aggregator for a sensor network is described in [8]. There are two main differences between the proposed model and the one described in [8]. The first one is conceptual and lies to the fact that Kang is using a single notification mechanism as a Service Data Element broker. The second one is related to the design which is based on the Grid Web Services Description Language (GWSDL) that allows the specification of the properties of a Grid service's data. In our case, the properties of the service data are irrelevant to the system and are incorporated in the report. Therefore, it falls to the application developer's responsibility to properly configure the Data Aggregator, allowing more flexibility in a business level.

3.3. Data Analyzer

The Data Analyzer component exploits the information gathered from the Aggregator component and proceeds with a set of simulations in order to produce the personalized healthcare and medicine proposal. In the case that there are models of the patient's reaction to the cure that are very difficult to solve with traditional mathematical methodologies, genetic algorithms is a promising solution. With the emergence of Grids, computational power is a cost-effective solution, especially in the case where the job is divided into small, almost parallel tasks. By using Genetic Algorithms we can harvest this power and simulate a number of possible treatments inside the range of acceptable solutions until the performance criteria are met.

The Data Analyzer component implements genetic algorithms (GAs) as a methodology used in many areas of technology for optimization and simulation purposes. Standard GA process include the definition of an initial population of candidate solutions (chromosomes) and the transformation of this population according to the success measured by the according function (in this case the results of the simulation run). The algorithm starts to execute and performs runs for each of the candidates. Then it creates a new generation based on the best chromosomes (elitism) of the previous one according to Darwin's law about the survival of the fittest while implementing a set of functions in order to find better results. These functions include mutation (the random alteration of parts of the candidate) and crossover (the concatenation of two candidates to create a new one by copying e.g. the first half from one and the second half from another) according to nature's evolution methodology. This means that while we search in what appears to be a random fashion inside the solution space, there are instinctive rules that direct the search towards more optimal cases, like the assumption that near a good candidate it is more possible to find a better one or that by combining two good chromosomes their offspring will combine the best characteristics from each one. The algorithm ends when the performance function succeeds in exceeding the desired limit or when the predefined number of generations is executed.

With the emergence of Grids, parallel GAs can be realized as a set of solutions for improving the performance of the search algorithm. For medical simulations, the execution time to run a test on a single processor means that we have to wait:

$$T = G * T_{chromo} + N * G * T_{simavg}$$

where T_{chromo} is the time needed to create the new generation, N the population size for each generation, G the number of generations and T_{simavg} the average simulation time for each candidate.

In [5] two solutions are recognized. The one incorporates a master-worker paradigm in which the master node is responsible for creating the population and evolving it while gathering results from the evaluation functions which are sent for calculation in the worker nodes. The delay that can be expected for this process for each generation is:

$$T = T_{chromo} + T_{comstart} + \max_{k}[T_{simk} + T_q] + T_{comres}$$

and for all generations:

$$T = G * T_{chromo} + G * T_{comstart} + G * T_{comres} + \sum_{i=1}^{G}\{\max[T_{simk} + T_q]\}$$

where T_{chromo} is the time needed to create the new generation in the master node, $T_{comstart}$ the time to send assignment to the worker node, T_{comres} the time to send the result of the maximum time node back to the master, T_q the wait time for execution in the worker node, T_{simk} the simulation time for each of the chromosomes k in the generation and G the number of generations.

The second solution comprises of the division of the population in subparts that evolve independently. Each node is responsible for its own part and only at specific times there is an exchange of the fittest solutions in order to enrich every node's chromosome base. The expected delay for this case for one generation is:

$$T = T_{chromo} + \sum_{k=1}^{N} T_{simk}$$

and for all generations:

$$T_{all} = G * T_{chromo} + \sum_{i=1}^{G}\{\ \sum_{k=1}^{N} T_{simk}\ \}_i + \frac{G}{C} * T_{com}$$

where T_{chromo} is the time needed to create the new generation, N is the population size for each generation, G is the number of generations, T_{simk} is the simulation time for each of the chromosomes k in the generation, C is the size of generation interval for communication of subpopulations and T_{com} is the communication time for exchange of subpopulations/

What will be the algorithm of choice is up to the implementers and the application they wish to optimize. For the case that simulation time is large the best solution is the master-worker implementation, due to the fact that the maximum time will be defined by:

$$\sum_{i=1}^{G}\{\ \max_{k}[T_{simk} + T_q]\ \}_i$$

if the number of available CPUs is equal to the population size. This means that we can decide about N based on the expected availability of Grid resources. In the subpopulations case it is defined mainly by:

$$\sum_{i=1}^{G}\{\ \sum_{k=1}^{N} T_{simk}\ \}_i$$

which for average values of T_{simk} can be transformed into:

$$N * G * T_{simkavg}$$

Of course the convergence will be faster due to the fact that space is searched in parallel so there will be less generations in order to achieve the same result. A thorough survey on the division of search space and the effect in the number of generations can be found in [6].

So it is actually dependent on the type of simulation running. If the execution time for each simulation run is high then the best solution is the master-worker method. If it is relatively low in computational cost and the search space is large then the subpopulation model will perform more efficiently. Furthermore, the doctor's involvement is required only in the beginning in order to determine the initial population and insert a number on constraints about the input variables that can be derived from the patient's medical records (for example the total dosage taken for a medicine) or the medicine's records (for example, dosage above this limit is lethal). This limits the search space and also helps in finding better solutions by excluding from the beginning the unfeasible ones.

Implementation of a Grid enabled GA service can be found in [7]. The detection and assignment of resources is done automatically and they follow both of the previously mentioned cases, dividing the population in subcomponents which are sent to different clusters and inside each cluster the subpopulation is run following the master-worker paradigm. The speed-up acquired is about 2-4 times faster than the case of running n subpopulations in one cluster.

Based on the above, the Data Analyzer implements a Genetic Algorithm that significantly decreases the time needed to perform simulations the medical simulations.

4. Application Scenario: Personalized Healthcare & Medicine

Aiming at providing healthcare and medicine proposals, all relevant data needs to be collected and analyzed for each patient. Moreover, prior to the

medicine proposal, a simulation has to be performed so as to conclude which therapeutic method fits better to the specific patient. The historical data of the patient are available through various sources (usually distributed across medical institutions) as well as other relevant medical data (such as clinical, demographic, etc), which are obtained for a specific medical case using pattern-matching algorithms in order to identify patients that are similar.

Our proposed approach can be used in two different ways, which are presented in the following sequence diagrams. The main difference between them is the actor that triggers the whole process.

In this case (depicted in Figure 3), the process is initiated by the end-user / doctor that requests an analysis for a particular patient. Thereafter the Analyzer component interacts with the Data Aggregator in order to acquire information for the analysis and more specifically about the simulation. The aggregator loads the policies that are associated with the specific analysis and then establishes the data transfer connections with the sensors and the Historic Data Services. The patient specific information is forwarded to the analyzer that performs simulations using the corresponding simulation services and based on the simulation results; a patient-specific medicine proposal is produced.

On the other hand, the process may be triggered by a real-time event from a sensor that is used to monitor some parameters of a patient (as depicted in Figure 4). The Data Aggregator loads the policies that are associated with this event and initializes the data streams that are indicated within the policies. These streams include either real-time data from other sensors or historical information. The Analyzer gathers the information and starts the analysis similar to the previous approach.

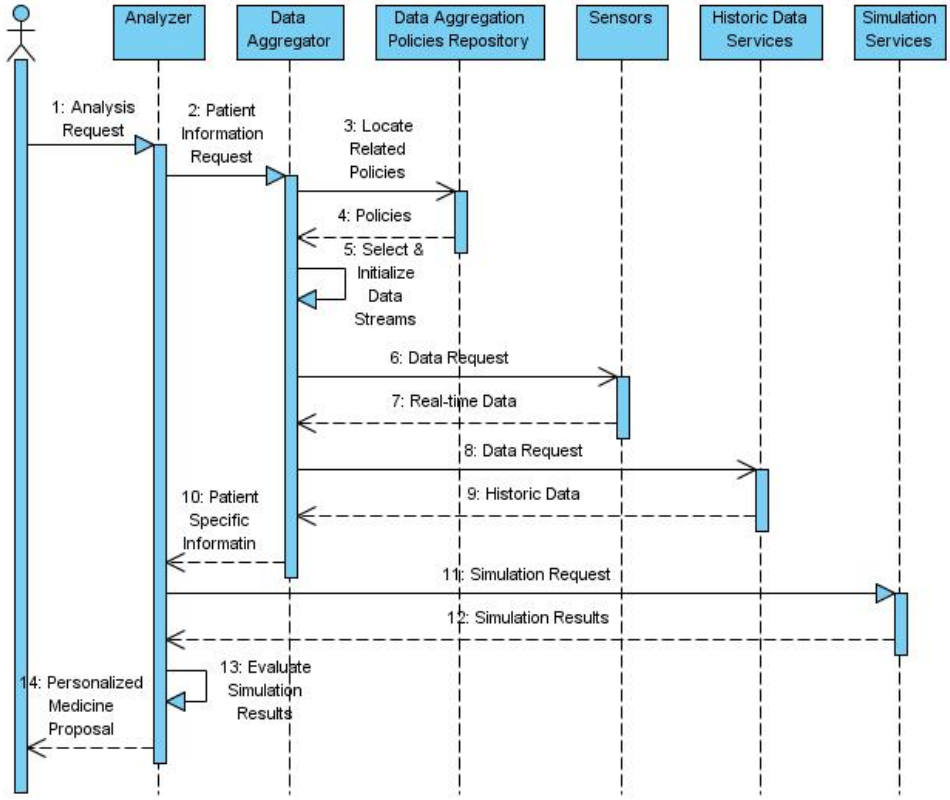

Figure 3. Sequence Diagram with the end-user as the main actor

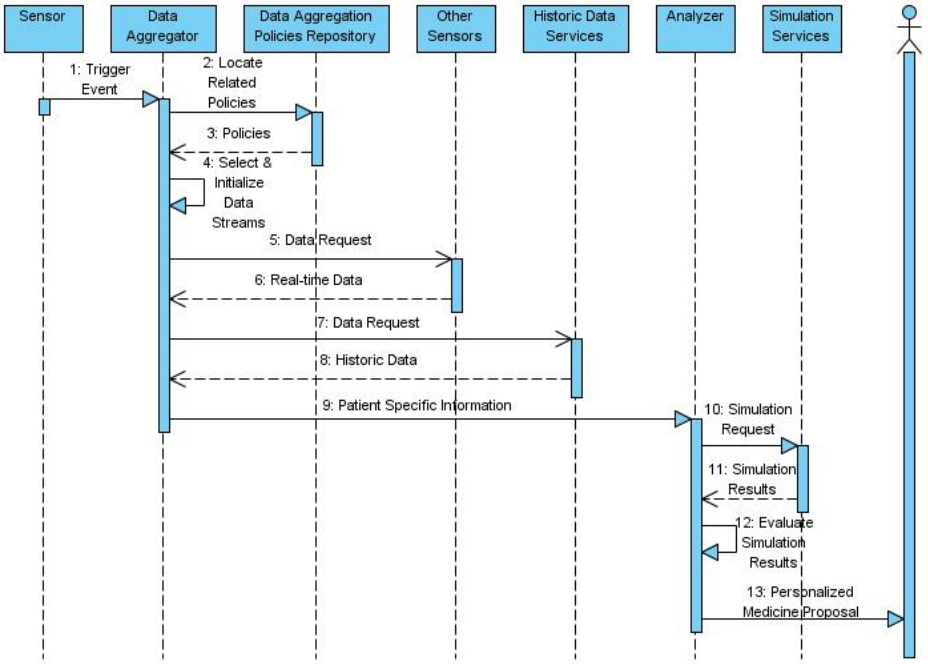

Figure 4. Sequence Diagram with a sensor as the main actor

5. Conclusions and Future Plans

Personalized healthcare and medicine poses the requirement of a complex infrastructure that will allow for aggregation of data from different sources and significant speed up of the data analysis and simulation process. Due to the exponential increase in the complexity of the simulation and the heterogeneity of required data and their preprocessing needs, as well as the large number of potentially involved patients, the large scale execution capabilities and vast computational capacities offered by Grids may prove exceptionally beneficial. Execution times prove that a considerable speedup may be achieved by using the Grid and that the Grid can also provide solutions in case that comparative results for therapeutic schemas are needed in real time. Moreover, the proposed set of components along with their interconnections / interfaces can be implemented to any service-based Grid middleware (such as GRIA [23], [24]) since they can be deployed as web services.

Notwithstanding, it is within our future plans to include a workflow management service to the proposed architecture, which based on a patient's record, will obtain information from specific providers since for example different hospitals - following their expertise - hold different kind of records for the same patient (e.g. heart records). Furthermore, future work on the data analyzer component will be focused on identifying the simulation with the longest execution time within a generation in order to avoid cases of low resources' utilization (occurring when the rest of the generation's simulations are completed).

Concluding, Grids form networks of resources along with monitoring and diagnosis facilities around the patient, which in combination with historical medical records, diagnosis and analysis services, allow for the realization of therapeutic schemes. A prerequisite to produce the aforementioned schemes is a component able to obtain information from different sources (e.g. cameras, sensors, etc) and aggregate this kind of data along with historical data in order to be consumable by data analysis and simulation services. The outcome of these simulations will enable medicine to become more personalized and patient-oriented, targeting the optimal individual treatment. To this direction, the data aggregation and analysis services presented in this paper can serve as a means to support this endeavor.

6. References

[1]. I. Foster, C. Kesselman, S. Tuecke. "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", International Journal of Supercomputer Applications, 2001

[2]. The EGEE Project, http://www.eu-egee.org/

[3]. Abramson, D., Giddy, J., and Kotler, L.," High Performance Parametric Modeling with Nimrod/G: Killer Application for the Global Grid?" IPDPS'2000, Mexico, USA, 2000.

[4]. G. Aloisio, M. Cafaro, S. Fiore, M. Mirto, "ProGenGrid: A Workflow Service Infrastructure for Composing and Executing Bioinformatics Grid Services", 2005.

[5]. H. Morishita, R. Ono, I. Ono, N. Okamoto, M.," A framework of Grid-oriented genetic algorithms for large-scale optimization in bioinformatics", Congress on Evolutionary Computation, 2003.

[6]. S. Tsutsui, "Forking GAs: GAs with Search Space Division Schemes", Evolutionary Computation, MIT Press.

[7]. Lim, D., Ong, Y., Jin, Y., Sendhoff, B., and Lee, B. "Efficient Hierarchical Parallel Genetic Algorithms using Grid computing". Future Generation Computer Systems, 2007.

[8]. YunHee Kang, "An Extended OGSA Based Service Data Aggregator by Using Notification Mechanism", Grid and Cooperative Computing, Springer, 2004.

[9]. The BRIDGES Project, http://www.brc.dcs.gla.ac.uk/projects/bridges/

[10]. The WISDOM Initiative, http://wisdom.eu-egee.fr/

[11]. The AKOGRIMO Project, http://www.mobileGrids.org/

[12]. S.Fiore, M. Mirto, M. Cafaro, S. Vadacca, A. Negro, G. Aloisio, "A GRelC based Data Grid Management Environment," 21st IEEE International Symposium on Computer-Based Medical Systems, 2008.

[13]. M. Mirto, S. Fiore, M. Cafaro, M. Passante, G. Aloisio, "A Grid-Based Bioinformatics Wrapper for Biological Databases,", 21st IEEE International Symposium on Computer-Based Medical Systems, 2008.

[14]. A. Tirado-Ramos, P.M.A. Sloot, A.G. Hoekstra, M. Bubak, "An Integrative Approach to High-Performance Biomedical Problem Solving Environments on the Grid", Parallel Computing, Special issue on High-Performance Parallel Bio-computing, 2004.

[15]. J. Snel, S. Olabarriaga, J. Alkemade, H. Andel, A. Nederveen, C. Majoie, G. den Heeten, M. van Straten, R. Belleman, "A Distributed Workflow Management System for Automated Medical Image Analysis and Logistics", 19th IEEE International Symposium on Computer-Based Medical Systems, 2006.

[16]. G. Aloisio, M. Cafaro, S. Fiore, M. Mirto, "A Split & Merge Data Management Architecture for a Grid Environment", 19th IEEE Symposium on Computer-Based Medical Systems, 2006.

[17]. S. Reynaud, G. Mathieu, P. Girard, F. Hernandez, "LAVOISIER: A Data Aggregation and Unification Service", Proceedings of Computing in High Energy and Nuclear Physics (CHEP06), Mumbai, India, February 2006.

[18]. Tzung-Shi Chen, Yi-Shiang Chang, Hua-Wen Tsai, Chih-Ping Chu, "Data Aggregation for Range Query in Wireless Sensor Networks", IEEE Wireless Communications & Networking Conference (WCNC 2007), Hong Kong, 2007.

[19]. E. Katsaloulis, A. Floros, Y. Provata, T. Cotronis, "Gridification of the SHMap Biocomputational Algorithm", International Special Topic Conference on Information Technology in Biomedicine, 2006.

[20]. L.Ardaiz, L. Diaz de Cerio, A. Gallardo, R. Messeguer, K. Sanjeevan, "ULabGrid Framework for Computationally Intensive Remote and Collaborative Learning Laboratories", IEEE International Symposium on Cluster Computing and the Grid, 2004.

[21]. T. Glatard, J. Montagnat, X. Pennec, "Grid-enabled workflows for data intensive medical applications", Computer Based Medical Systems, Special Track on Grids for Biomedicine and Bioinformatics, 2005.

[22]. Open Grid Services Architecture (OGSA), www.ggf.org/documents/GFD.30.pdf

[23]. M. Surridge, S. Taylor, D. De Roure, and E. Zaluska, "Experiences with GRIA-Industrial Applications on a Web Services Grid", in Proceedings of the First International Conference on e-Science and Grid Computing, pp. 98-105. IEEE Press, 2005

[24]. GRIA, Grid Resources for Industrial Applications, www.gria.org