A Reference Framework for Requirements and Architecture in Biomedical Grid Systems

Chris A. Mattmann[1, 3], Vito Perrone[2], Sean Kelly[1], Daniel J. Crichton[1],
Anthony Finkelstein[2], Nenad Medvidovic[3]

[1]Jet Propulsion Laboratory
California Institute of
Technology
Pasadena, CA 91109, USA
{crichton, mattmann, kelly}
@jpl.nasa.gov

[2]Dept of Computer Science
Univ. College London
London WC1E 6BT, UK
{v.perrone,a.finkelstein}
@cs.ucl.ac.uk

[3] Computer Science Department
Univ. of Southern California
Los Angeles, CA 90089, USA
{mattmann,neno}@usc.edu

Abstract

In this paper we introduce the work done to define a framework for requirements and architectural understanding in biomedical grid computing systems. A set of core requirements for biomedical grids have been identified on the basis of our experience in the analysis and development of several biomedical and other grid systems including the National Cancer Institute's Early Detection Research Network (EDRN) in the US and the National Cancer Research Institute (NCRI) Platform in the UK. The requirements have been specified taking into account different points of view and are intended as a core set that can be extended on the basis of project specific aspects. These are also mapped to existing architectures of biomedical grid systems, and their constituent components. Such a framework is intended as a guide for equipping developers with conceptual tools to avoid costly mistakes when architecting biomedical grid systems.

1. Introduction

During the last ten years, biomedical research has grown rapidly fuelled in parts by the advances in computational bioinformatics, government support, and increase of interest for molecular-based medicine and in the use of Internet and information technologies (IT). Many data repositories and services have been developed and made accessible through the Internet to researchers who need to use them in their daily research activities. Often, however, these resources have been developed in independent projects and suffer of most of the known incompatibilities [1], e.g., heterogeneous data formats, and models. Such incompatibilities hamper the typical researcher's tasks where access and use of data available in different repositories is required. To address this problem requires recognizing that true advances in biomedicine can be achieved if data produced in countless distributed initiatives can be accessed in an integrated way, alleviating the heterogeneity of the data and systems involved. Several efforts have been established worldwide aimed at developing large-scale integrative systems that deal with the aforementioned problems. Most of these projects have similar objectives and propose similar solutions. For instance, many projects use the grid paradigm as reference for the system development, and reference implementations such as Globus [2] have been adopted to some extent. However, the research literature concerning existing projects mainly focuses on technical issues, whereas analysis and early architecture design aspects have not received, in our view, the attention they deserve. We believe that this course is extremely risky for the community since poor understanding of requirements can lead to system that is different from what users really need. Furthermore, wrong architectural choices (again related to poor understanding of requirements) can lead to the development of ineffective systems, high maintenance costs, and scalability problems.

Over the last years we have been involved in the analysis and development of several grid systems in the biomedical and in other e-science domains [1, 3-7]. We have also reviewed the documentation available for many other existing grid projects [8, 9]. We have observed that although projects may refer to (domain-)specific topics in e-science, all share a number of aspects and achievements upon which other similar systems can capitalize. This is even more evident if we restrict our focus on a particular domain such as biomedicine. Given our specific expertise on requirements analysis and architecture design, we have recently begun a project aimed at defining a reference framework where a core set of requirements for biomedical systems are related to architectural choices, including definition of a set of canonical components and styles for biomedical grid systems. Our work is intended to provide guidelines and reference requirements and architectural patterns to biomedical grid analysts and designers, allowing them to capitalize on the work we and other practitioners have done in similar projects.

2. Background and Related Work

There have been several high-level efforts to describe over-arching challenges for biomedical grid systems [6, 10, 11]. Where these efforts lack is in the formal, principled definition of requirements in the domain. Some of our recent projects have studied formal grid requirements specification in general [8, 9], however, these projects focus on general computational and data grid requirements, neglecting to consider the domain specific requirements for biomedical grids as a whole. Our current work on biomedical grids focuses on defining the general grid requirements, and additionally defining and capturing those requirements specific to the biomedical community, e.g., the system shall interface with picture archiving and communication services.

Our work is grounded within the context of two real-world, biomedical grid projects. The first, the U.S. National Cancer Institute (NCI)'s Early Detection Research Network (EDRN) [4, 5, 7], is a large-scale research network comprising over 31 institutions and a large number of researchers supporting the early detection of cancer in the United States. The second project is the UK's National Cancer Research Institute (NCRI) [12], a network of 20 member organizations supporting common plans for cancer research and strategic initiative in the UK. Within the context of these two projects, we have delivered operational grid software, engineered formal requirements and specifications for biomedical grid software components, and participated in the development of ontologies and vocabulary allowing cancer researchers to discover and annotate their data. Our work can be compared with Brenton et al. [10] which describes the European Data Grid project in the context of supporting biomedical informatics. The authors describe three key genomics challenges that biomedical grids must address: (1) data acquisition and storage, (2) access to external bioinformatics web resources, and (3) local data management. The authors explain several core components for biomedical grids, including science processing algorithms, and grid middleware. These coarse-grained components map to our fine-grained biomedical components described in Section 4. In contrast to Brenton et al., our work identifies similar core components, but also shows their mapping to the core requirements, and a set of core architectural principles. Brenton et al.'s work is similar to that of Pohjonen et al. [13], who describe the use of biomedical grids in the pervasive computing domain.

3. Defining Core Requirements for Biomedical Grid Systems

Biomedical grids are generally large-scale systems addressing the key issues of data and services sharing in an open and changing community. In such a context a clear understanding of the key goals and requirements is crucial to avoid developing a system that does not fit the stakeholders' expectations. The complexities of the biomedical domain are two-fold: the heterogeneity of initiatives and stakeholders and the high level of specialization of the various sub-fields that can contribute to the grid system. Both of these complexities pose several challenges to requirements engineers and architecture designers. It is crucial to adopt systematic requirements engineering methods to master the complexity and remove early any ambiguity potentially leading to late failure. On the other hand, it is known [14] that in many cases such techniques are not widely adopted in the practice even in the case of complex systems development.

In our analysis we have adopted rigorous requirements engineering techniques such as the *ViewPoints* framework [15] and goal oriented methods [16] to analyze the requirements of our systems. By leveraging basic system engineering principles such as *separation of concerns, decentralization, layering, etc.,* these techniques have allowed us to achieve two main results: (1) Isolate a core set of requirements which can be considered typical of this category of systems; (2) Take into account explicitly the different perspectives that coexist in such a category of systems in order to analyze early potential conflicts and synchronies [17].

The requirements we have identified can be considered a core set which should apply, in variable measure, to any biomedical grid project but that can be extended and adapted on the basis of each project peculiarities. These are not intended as all the possible requirements of any biomedical system but as a guide analysts and designers involved in the development of biomedical systems can use to drive the analysis activities and the architecture definition. To address this objective, requirements have been specified in a way that is general enough to concern a category of systems rather than to a specific system, but detailed enough to relate them to specific architectural choices (as we detail in Section 4). Both functional and non-functional requirements are been covered. Their description reflects the multi-perspective analysis we have carried out. Given their purpose, only three high level perspectives have been considered: the *grid users*, the *resource providers* and the *grid* (i.e. the system being developed and its proponents). ***Grid Users*** are those stakeholders who need to find and use biomedical data for their daily work. The grid will provide them with the needed functionalities to discover, access and process the needed data. Different types of users should be

Table 1. Reference Requirements for Biomedical Grids

Requirement		Short Description
R1	Data and Data Management	Manage different data formats, handle geographically dispersed data
R2	Data Access	Provide uniform access to heterogeneous data repositories
R3	Metadata	Use metadata as a means for discovering data, and explicitly query metadata registries for metadata about data within the system.
R4	Data Publishing	Provide basic means by which distributed data can be accessed and retrieved
R5	System Information, Monitoring and Tracking	Allow users and system administrators to know information about the biomedical grid system itself (e.g., what resources are currently available)
R6	User Interface and User Functions	Allow custom and standard user interfaces to "plug in" to the backend biomedical grid infrastructure, and query and retrieve metadata and data.
R7	Applications and Tools	Allow existing biomedical applications (e.g., PACS services) to plug in to the grid.
R8	NFR - Security	Provide authorization and authentication for biomedical grid users, across organizations.
R9	NFR - Compatibility	Biomedical grid infrastructures should be interoperable with one another.
R10	NFR - Load, Capacity, and Scalability	Allow for petabyte scale data volumes, and efficient data transfer.
R11	NFR - Performance	Optimum service levels should be maintained as system load and system state change.
R12	NFR - Fault Tolerance and Robustness	Security services should not have any possible single point of failure, data access services should show some degree of fault tolerance
R13	NFR - Extensibility and Modifiability	It must be possible to add new services and resources to the system once deployed.
R14	NFR - Integrability	The system must integrate heterogeneous components whether project specific or legacy

considered, however it is often important to distinguish between ***naive user*** and the ***expert user***, depending on their skills and their attitude as to dealing with informatics resources. ***Resource providers*** are organizations that curate either data repositories or services registries [18]. Generally resource providers provide the data or services that may be used by the grid users in their daily activities. The grid generally enables access and integration of data across the various resource providers and may support data publishing from researchers to resources. Finally, the ***grid*** point of view considers the organization(s) involved in the deployment of the grid system. Requirements have been grouped into a number of categories that can be considered high level functionalities or qualities the system has to provide, including 7 categories of functional requirements and 7 non-functional ones. Lack of space prevents us to describe in full all the requirements we have identified. As an example, Table 1 shows some excerpts from the specification of a functional category, *R2-Data Access*. As for any requirement, generic considerations along with the perspectives of the three types of stakeholders are discussed and specific sub requirements identified. The multi-perspective view and the categorization should also ease adaptation and extension of this core set to encompass the specific characteristics of biomedical grid projects.

4. Relating Requirements to Architectural Choices

There are many architectural styles regularly utilized in grid systems as we have noted in our previous work [8]. Architectural styles are key design idioms that guide the componentization, and assembly of software architectures in a given family of software systems. There have been numerous identified architectural styles over the years (see [19-21]), including pipe-and-filter, and peer-to-peer styles. The chosen architectural style for a software system can greatly affect its design and requirements. Thus, the architectural style chosen for a biomedical grid should reify the design constraints and requirements presented in the prior section. The two pervasive styles used in biomedical grid systems are: (1) the client/server style, and (2) the peer-to-peer style. For example, client/server style components regularly involve *synchronous* interactions, which are highly reliable. Client/server components are well suited to deal with *data access*. On the other hand, peer-to-peer components are highly scalable, involving asynchronous interactions, and fault tolerance. Such components are more easily suited to deal with *load*, *capacity* and *scalability* requirements.

We have found there to be four main architectural principles, that, coupled with the above two architectural styles, guide the set of canonical components for a

Table 2. Partial mapping of requirements to architectural principles and core components

	AP1	AP2	AP3	AP4	Data Repositories	Metadata Registries	Workflow Components	Data Distribution
Data Management				X	X			X
Data Access				X	X			X
Metadata	X		X			X		
Data Publishing							X	X
User Interface							X	
Applications and Tools		X					X	
Compatibility	X	X						
Load, Capacity and Scalability	X		X		X	X		X

biomedical grid. The first principle, *division of labor (AP1)*, ensures that no one component in the system is responsible for providing all of its capabilities. This principle allows biomedical grid components to support "plug and play" architectures, and load, capacity and scaling. The second principle, *technology independence (AP2)*, mandates that the underlying implementation substrate not dictate the architecture of the system, and vice versa. This principle directly affords biomedical grid components extensibility, and modularity, supporting the large amount of the aforementioned heterogeneity of the domain. The third principle, *metadata as a first class citizen (AP3)*, recognizes the need for explicit software components to manage *metadata*, or data about data. Metadata describes the scientific data in a biomedical grid, allowing complex search, and discovery of images, specimens, through components such as metadata registries. The fourth principle is *separation of the software and data models (AP4)*, allowing both to evolve independently. Because data, software, and users are highly heterogeneous in biomedical grid systems, the software that is used to realize the grid must not be directly tied to the data that it manages. Put simply, a change in the data model (e.g., the addition of an attribute to describe an image) *should not* mandate a change in the software implementation.

Using the aforementioned styles, and architectural principles, we have deduced the core classes of components required in any biomedical grid. This set is not meant to be exhaustive, but represents a tangible milestone as we move forward with our understanding of such systems. The first class of biomedical grid component is a *data repository*. Data repositories manage science data information, such as locations of files, their identifiers, and sizes. Data repositories are client/server components that address data access and management requirements, ensuring that there exists due division of labor within the system. The second class of components is *metadata registries* that are client/server components that catalog and manage metadata. Metadata stored by the registry generally falls into three categories: housekeeping information (e.g., object id, collection time), resource information (e.g., author, creator, location), and domain-specific information (e.g., specimen code, patient id). Metadata registries directly address the separation of software and data model principle, as well as the metadata as a first class citizen principle. The third class of components is *workflow components*. Workflow components are responsible for managing scientific data "pipelines": essentially pipe-and-filter style [22] compositions of processing algorithms that allow scientists to generate derived data and value added science from original raw samples. Workflow components fulfill the architectural principles of division of labor, and technology independence. This class of components fulfills data publishing, user interface and application-level requirements. The final class of components is *data distribution and retrieval* components. Data distribution and retrieval components allow for users in the biomedical grid to access and retrieve data from data repositories. Data distribution also occurs between components within the biomedical grid, such as between two data repositories, or between a workflow component and a data repository. These components directly support technology independence, and software and data model independence. Additionally, these components address requirements *data management, data access, data publishing, and performance*. Table 2 summarizes a sampling of the requirements discussed in the prior section, identifying the relationships with architectural principles (AP) and components described above.

5. Discussion and Future Work

Clear understanding of requirements and their relationships with architectural choices is a critical success factor for any large-scale project. However, the complexity of the bioinformatics poses challenges to the teams involved in the analysis and architecture design for grid systems. State-of-art requirements and software engineering techniques can help to master the complexity but often the teams involved are not sufficiently equipped or the project(s) lack funding to perform a thorough analysis.

Drawing on our experience, we have developed a framework where a set of core requirements distilled from the biomedical domain has been related to typical architectural choices of grid systems. Although we do not claim that our work is a ready to use as requirements and architecture specification yet we believe it can provide significant guidance to teams involved in the development of grid systems in the biomedical domain. The framework can be considered either a starting point for analysis activities or a reference against which the requirements and architecture specification of a new grid system can be compared. The framework has been conceived to enable its extension to encompass the characteristics of a specific system. Requirements are extended either by specializing the requirements within one of the proposed stakeholders [15] or by specializing the stakeholders.

Of course this approach does not prevent introducing new independent stakeholders and/or requirements. We note that introducing new requirements could entail new conflicts that will need to be resolved. Although given its abstract nature it is impossible to evaluate the coverage of requirements and architectural components with respect to a generic biomedical grid system, we have validated it against our case study projects and a number of other projects such as [10, 11, 13, 23] (using the publicly available documentation) obtaining encouraging results.

References

[1] A. M. Ouksel and A. Sheth, "Semantic Interoperability in Global Information Systems," *ACM SIGMOD Record*, 1999.

[2] C. Kesselman, et al., "The Anatomy of the Grid: Enabling Scalable Virtual Organizations," *Intl' Journal of Supercomputing Applications*, pp. 1-25, 2001.

[3] R. D. Bentley, et al., "EGSO - The European Grid of Solar Observations," In Proc. *10th European Solar Physics Meeting "Solar Variability: From the Core to Outer Frontiers"*, 2002.

[4] D. Crichton, et al., "A Distributed Information Services Architecture to Support Biomarker Discovery in Early Detection of Cancer," In Proc. *2nd IEEE e-Science and Grid Computing Conference*, Amsterdam, the Netherlands, 2006.

[5] D. J. Crichton, et al., "An Interoperable Data Architecture for Data Exchange in a Biomedical Research Network," In Proc. *CBMS*, 2001.

[6] A. Finkelstein, et al., "Computational Challenges of Systems Biology," *Computer*, vol. 37, pp. 26-33, 2004.

[7] H. Kincaid, et al., "A National Virtual Specimen Database for Early Cancer Detection," In Proc. *CBMS*, 2003.

[8] A. Finkelstein, et al., "Relating Requirements and Architectures: A Study of Data Grids," *J. Grid Computing*, vol. 2, pp. 207-222, 2004.

[9] C. Mattmann, et al., "Unlocking the Grid," In Proc. *Component-based Software Engineering (CBSE)*, St. Louis, MO, 2005.

[10] V. Brenton, et al., "DataGrid, Prototype of a Biomedical Grid," *Methods of Information in Medicine*, vol. 42, pp. 143-147, 2003.

[11] K. H. Buetow, "Cyberinfrastructure: Empowering a "Third Way" in Biomedical Research," *Science*, vol. 308, pp. 821-824, 2005.

[12] V. Perrone, et al., "Developing an Integrative Platform for Cancer Research: a Requirements Engineering Perspective," In Proc. *5th e-Science All Hands Meeting*, 2006.

[13] H. Pohjonen, et al., "Pervasive Access to Images and Data -- The Use of Computing Grids and Mobile/Wireless Devices Across Healthcare Enterprises," *IEEE Trans. Information Technology in Biomedicine*, vol. 11, pp. 81-86, 2007.

[14] H. Kaindl, et al., "Requirements Engineering and Technology Transfer: Obstacles, Incentives and Improvement Agenda," *Requirements Engineering*, vol. 7, pp. 113-123.

[15] A. Finkelstein, et al., "Viewpoints: A Framework for Integrating Multiple Perspectives in Systems Development," *Intl' Journal of Software Engineering and Knowledge Engineering*, vol. 2, 1992.

[16] A. Dardenne, et al., "Goal-Directed Requirements Acquisition," *Science of Computer Programming*, vol. 20, 1993.

[17] A. v. Lamsweerde, et al., "Managing Conflicts in Goal-Driven Requirements Engineering," *IEEE TSE*, vol. 24, pp. 908-926, 1998.

[18] "Information Architecture Reference Model," CCSDS, Draft Informational Report, Green Book CCSDS-312.0-G-0, 2006.

[19] R. Fielding, "Architectural Styles and the Design of Network-based Software Architectures", Ph.D., University of California, Irvine, 2000.

[20] R. Khare and R. N. Taylor, "Extending the Representational State Transfer (REST) Architectural Style for Decentralized Systems," In Proc. *ICSE*, Edinburgh, Scotland, 2004.

[21] R. N. Taylor, et al., "A Component-and-Message-Based Architectural Style for GUI Software," *IEEE Transactions on Software Engineering*, vol. 22, pp. 390-406, 1996.

[22] M. Shaw and D. Garlan, *Software architecture : perspectives on an emerging discipline*. Upper Saddle River, N.J.: Prentice Hall, 1996.

[23] caBIG, http://cabig.cancer.gov/, 2007