Cluster Computing manuscript No.
(will be inserted by the editor)

Research Advances by Using Interoperable e-Science Infrastructures

The Infrastructure Interoperability Reference Model Applied in e-Science

Morris Riedel · Felix Wolf · Dieter Kranzlmüller · Achim Streit · Thomas Lippert

Received: 2009-02-01 / Accepted: date

Abstract Computational simulations and thus scientific computing is the third pillar alongside theory and experiment in todays science. The term e-science evolved as a new research field that focuses on collaboration in key areas of science using next generation computing infrastructures (i.e. co-called e-science infrastructures) to extend the potential of scientific computing. During the past years, significant international and broader interdisciplinary research is increasingly carried out by global collaborations that often share a single e-science infrastructure. More recently, increasing complexity of e-science applications that embrace multiple physical models (i.e. multi-physics) and consider a larger range of scales (i.e. multi-scale) is creating a steadily growing demand for world-wide interoperable infrastructures that allow for new innovative types of e-science by jointly using different kinds of e-science infrastructures. But interoperable infrastructures are still not seamlessly provided today and we argue that this is due to the absence of a realistically implementable infrastructure reference model. Therefore, the fundamental goal of this paper is to provide insights into our proposed infrastructure reference model that represents a trimmed down version of OGSA in terms of functionality and complexity, while on the other hand being more specific and thus easier to implement. The proposed reference model is underpinned with experiences gained from e-science applications that achieve research advances by using interoperable e-science infrastructures.

Morris Riedel
Co-Chair OGF Grid Interoperation Now Community Group
Jülich Supercomputing Centre, Instiute for Advanced Simulation
Forschungszentrum Jülich
D-52425, Jülich, Germany
Tel.: +49 2461 3651
E-mail: m.riedel@fz-juelich.de

Keywords e-Science Infrastructures · HPC · HTC · Interoperability · Reference Model · e-Health

1 Introduction

E-science applications take already advantage of various e-science infrastructures that evolved over the last couple of years to production environments. However, the Open Grid Services Architecture (OGSA) concept originally defined by Foster et al. in 2002 [7] as a common basis for such infrastructures is only slowly adopted. While OGSA represents a good architectural blueprint for e-science infrastructures in general, we argue that the scope of OGSA is too broad to be realistically implementable in today's production e-science infrastructures. We identified two reasons for this. First, the process of developing open standards that are conform to the whole OGSA ecosystem takes rather long, including the precise specification of all the required interfaces of these services and their adoption by the respective middleware providers. Second, the launch of OGSA-conform components within production e-science infrastructures consumes substantial time after being evaluated for production usage.

The absence of a realistically implementable reference model is diametral to the fundamental design principles of software engineering and has thus lead to numerous different architectures of production e-science infrastructures and their deployed technologies in the past. Some examples are the Enabling Grids for e-science (EGEE) infrastructure, which uses the gLite middleware [12], the TeraGrid infrastructure, which uses the Globus middleware [6], the Distributed European Infrastructure for Supercomputing Applications (DEISA) which uses the UNICORE middleware [29], the Open Science Grid (OSG), which uses the Virtual Data Toolkit (VDT), and NorduGrid, which uses the ARC middleware. Unfortunately, the most elements of these technologies and thus infrastructures are not interoperable at the time of writing because of limited adoption of open standards and OGSA concepts.

The lack of interoperability is a hindrance since we observe a growing interest in the coordinated use of more than one infrastructures from a single client that controls interoperable components in different e-science infrastructures. Recently, Riedel et al. [22] provided a classification of different approaches of how to use e-science infrastructures today. Among simple scripts with limited control functionality (i.e. loops), scientific application plug-ins, complex workflows, and interactive acces, there is also infrastructure interoperability mentioned as one approach to perform e-science. A growing number of e-scientists would like to benefit from interoperable e-science infrastructures in terms of having seamless access to a wide variety of different services or resources. For instance, many of them raise the demand to access both High Throughput Computing (HTC)-driven infrastructures (e.g. EGEE, OSG) and High Performance Computing (HPC)-driven infrastructures (e.g. DEISA, TeraGrid) from a single client or science portal. In this context, the fundamental difference between HPC and HTC is that HPC resources (i.e. supercomputers) provide a good interconnection of cpus/cores while HTC resources (i.e. pc-pools) do not.

Although one goal of OGSA is to facilitate the interoperability of different Grid technologies and infrastructures in e-science and e-business, we state that the requirements for interoperability in e-science infrastructures have to be specified much more precisely than it has been done within OGSA. In our earlier work [23] we have defined interoperability requirements based on lessons learned from interoperability work between production e-science infrastructures. Based on these requirements, the contribution of this paper is to define the necessary building blocks for an *Infrastructure Interoperability Reference Model (*IIRM*)* that is much closer oriented towards the interoperability of production e-science infrastructures than OGSA and already applied in practice to numerous interoperability use cases. It is important to note that this reference model should not replace OGSA but rather trim it down in functionality by dropping several parts of it and refining other parts that are mostly relevant to interoperability of e-science infrastructures today.

History of computer science shows that often complex architectures were less used than their trimmed down versions. For instance, the complex Structured Generalized Markup Language (SGML) was less used than its smaller version Extensible Markup Language (XML), which was less complex and therefore quickly became a de-facto standard in Web data processing. We argue that the same principles can be exploited with OGSA by defining a more limited, but more usable infrastructure reference model. This becomes also increasingly important in the context of economic contraints since the rather huge OGSA body requires massive amounts of maintenance while our reference model should significantly reduce these maintenance costs by providing only a small subset of functionality, but this in a more well-defined manner tuned for today's production usage.

This paper is structured as follows. Following the introduction, the scene is set in Section 2 where we survey the state-of-the art e-Science infrastructure that helped to identify specific requirements for interoperable e-Science infrastructures listed in Section 3. Section 4 defines a set of common interoperability approaches, while Section 5 presents our design approach in general and our infrastructure reference model in particular. The recent research advances in e-Science are presented in Section 6 and give insights into the implementation of our model and its evaluation by e-scientists. Finally, after surveying related work in Section 7, we present our conclusion in Section 8.

2 State-of-the-art e-Science Infrastructures

Today, scientists regard computational techniques as a third pillar alongside experiment and theory as shown in Figure 1. In this illustration, the first pillar stands for a certain theory or specific model in a research field.

One example of this pillar are scientists that use complex mathematical models to predict the diffusion of harmful materials in soil. The second pillar points to experiments performed in laboratories or probes of material, for instance, of harmful materials in soil. The computational techniques in the third pillar allows for computer-simulations based on efficient numerical methods and known phyisical laws. In our mentioned example, scientists can compute the flow of water underground and simulate the way in which various harmful substances react with potentially damaging consequences. We refer to these three pillars and its scientific results as *traditional scientific computing.*

As an addition, to the usual defined three pillars that are the foundations for science, the term enhanced science (e-science), sometimes also called electronic science evolved in the last couple of years. We base our considerations on the views of e-science given by John Taylor [30] that is defined as follows: *e-science is about global collaboration in key areas of science and the next*

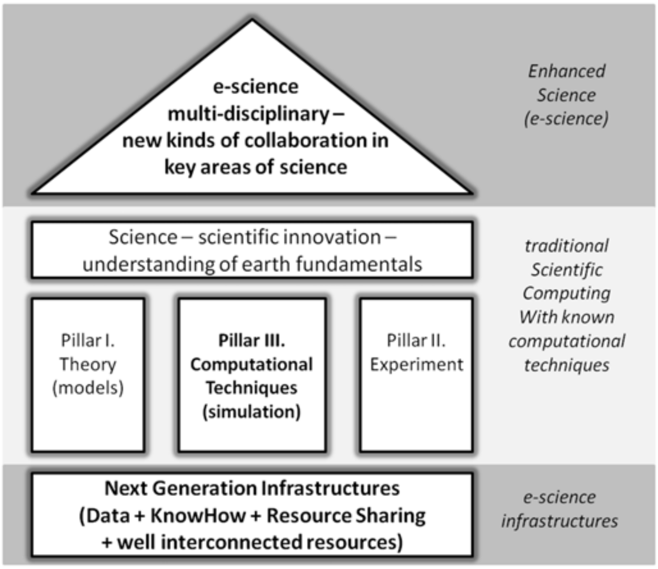

Fig. 1 The three fundamental pillars of traditional scientific computing make use of a solid e-science infrastructure basement to achieve e-ecience.

generation infrastructure that will enable it. Often, this definition has been extended in several ways to include a particular focus or a dedicated technology. For instance, recently, the dynamic deployment features achieved in using virtualization techniques of so-called clouds have been added as required features of next generation infrastructures. Nevertheless we keep this rather mature definition as a base for our discussions.

Today, such next generation infrastructures can be considered as a solid basement for the three pillars of traditional scientific computing and thus enable e-science that can be seen as a roof that basically stands for the collaboration in key areas of science as shown in Figure 1. We learned over the years that these next generation infrastructures can have many different names. In the US these infrastructures are basically named as cyber-infrastructures, while in Europe such a next generation infrastructure is called e-Infrastructure. At the time of writing, these next generation infrastructures are implemented as Grids, often named as e-science infrastructures in the rather scientific domain. However, all share the common goal of developing the future problem solving infrastructure that enables innovation through data, knowledge, and (dynamic) resource sharing via high performance interconnections.

Over the years, various types of e-science infrastructures evolved that can be basically classified into different categories according to their services in general and their offered resource-types in particular. As shown in Figure 2, the first category is represented by HPC-driven infrastructures since known Grids of this type integrate mostly large-scale cluster or supercomputers to enable massively parallel applications. This refers to applications that require an excellent interconnection between the cpus/cores and typically use parallel programming techniques such as the Message Passing Interface (MPI) [17] or OpenMP [5]. Two famous examples in this category are TeraGrid in the US and DEISA in Europe. In contrast, the second category is represented by HTC-driven infrastructures since Grids in this category are mostly tuned to support farming jobs (aka embarassingly parallel) jobs that does not require a good interconnection between cpus/cores. Known infrastructures of this type are EGEE in Europe, OSG in US, or NorduGrid in the nordic regions. There are also some infrastructures that we consider as hybrid infrastructures since they provide access to a limited set of large-scale facilities while still providing also access to pc pools and smaller clusters. Infrastructures of this category are, among others, the National Grid Service (NGS) of the UK, or the German national Grid D-Grid.

In theory it is basically hard to define the clear boundaries for this classification. In practice, however, the boundaries and scope of these categories are fundamentally different, especially when the resource type is considered as well as the overal usage policies. In terms of resource types we consider for our classification the major difference between large-scale high-end supercomputers compared to small or medium-sized as well as pc pools. Also important for this classification is the resource usage policies. In HPC-driven Grids the costly computational time is only provided to certain invididuals that passed a multi-step evaluation of their research proposal (i.e. DEISA Extreme Computing Initiative - DECI [34]). In contrast, HTC-driven Grids tend to provide access to resources based on known virtual organizations (VOs) and its members without the need for each e-scientist to apply for computational resoures. Nevertheless, a hybrid category is provided for those infrastructures that are rather hard to categorize without making wrong assumptions.

We also learned from this categorization that these different e-science infrastructure will remain since it breaks basically down to the physical hardware, its properties, or usage paradigms such as HPC and HTC concepts. As one example, in Europe, the HPC-driven Grids DEISA and HTC-driven EGEE are complementary and thus aim both for a long term strategy in terms of sustained funding. Since both e-science infrastructure types are relevant to research in Europe, the European Commission supports both with funding that can be seen as an evolution of these projects. While the HPC-driven DEISA infrastructure will work closely together with the PRACE infrastructure to include supercomputers in the petaflop/s performance range, the EGI will represent an evoluation of the EGEE infrastructure funded by the EU and EGI members that are the national Grid initiatives of many EU member states.

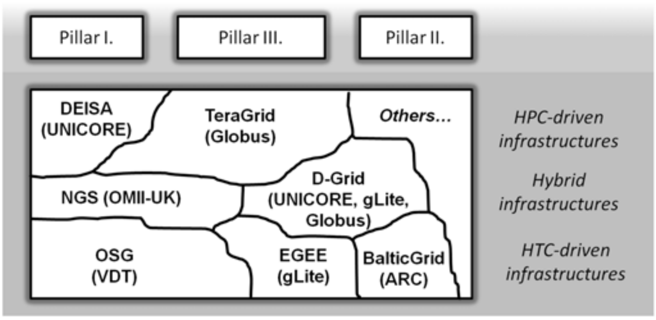

Fig. 2 Different classifications of production e-Science infrastructures.

The major problem of these different categories and their implementations is that the solid basement of Figure 1 basically breaks into a wide variety of different bricks as illustrated in Figure 2. In other words, this one next generation e-science infrastructure is basically represented by a well-known set of world-wide non-interoperable *Grid islands* majorly funded through public sources today. To provide an example, EGEE and DEISA are projects funded by the European Commission, while TeraGrid, OSG, D-Grid are national infrastructures based on different funding sources. As a side-remark, a classification according to the funding sources or nationality of the infrastructure that runs it does not make sense since we see scientific research in general and research innovation in particular as a world-wide challenge without any national boundaries.

In addition to the categorization of infrastructure types, Figure 2 also reveals that each e-science infrastructure runs its own technology. Even within one category, different middleware technologies exist. While TeraGrid uses Globus, DEISA is based on UNICORE. EGEE has deployed gLite, while OSG uses VDT and NorduGrid uses ARC. The hybrid category is a bit different, since NGS runs the OMII-UK software stack that basically also includes different software technologies and D-Grid became much known by its concept of deploying UNICORE, Globus and gLite in parallel.

Note that at the time of writing all these different Grid middleware systems are as a whole not interoperable with each other. We argue that this is due to the absence of a realistically implementable infrastructure reference model in Grids. As a result, the state-of-the art e-science infrastructures still struggle to provide the e-scientists with a stable basement (cp. Figure 1) to fully leverage the power of world-wide existing resources by one particular e-science project.

3 Requirements

Riedel et al. [20] published numerous experiences and lessons learned from numerous international interoperability projects and efforts within the Open Grid Forum (OGF) - Grid Interoperation Now (GIN) community group that in turn lead to several specific requirements for the interoperability between e-science infrastructures. In earlier work [23], we already provided a detailed list of requirements for infrastructure interoperability, therefore this section only highlights a few of them.

We already mentioned that infrastructure interoperability can be seen as one approach to perform e-science [22]. But the term 'infrastructure interoperability' is rather vague and thus we shortly provide here insights what we mean by this term and thus survey the advantages of having it. First and foremost, interoperability of infrastructures enables the access to a broader variety of resources. For instance by conveniently using different types of resources such as HTC and HPC resources together with a single client. This implies another advantage in the sense that interoperability is a smart way of extending the functionality of one infrastructure. Scientists want to leverage the unique key features of the different infrastructures, for instance by using the brokering mechanisms in HTC-driven infrastructures or using the explicit choose of resources within HPC-driven infrastructures to tune a sourcecode for large-scale simulation runs. As a consequence, one significant overall requirement of this approach is that the typical single-sign on feature has to be supported that enables e-scientists to use only one credential in order to access different infrastructures.

Also, interoperability of infrastructures allows for enhanced resource utilization as shown by Rodero et al. [25]. Implied is an advantage to have a better load-balancing between different kinds of computing infrastructures and their resources. Even some scientists consider to combine resources of different infrastructures in order to realize more realistic simulations (i.e. multi-physics using HTC and HPC resources). Other advantages gain more interest in the last couple of years and refer to a wide variety of economic considerations. They refer not only to the synergetic developments that avoid duplicate technology development in each of the e-science infrastructures. These considerations rather address the possibility of saving valuable computational time on rare and costly HPC resources in such infrastructures. With infrastructure interoperability in place, scientists could easily use HTC-driven infrastructure for smaller evaluation jobs before using the high-end supercomputers in HPC-driven infrastructures for 'full-blown' production runs. But in order to use this approach, the fundamental requirement is that e-scientists and resource providers agree on usage policies.

An important aspect of interoperability that is often misunderstood is that the technologies have to be basically linked together even if they are considered to be on different levels. In [23], we have stated that interoperability affects basically different layers that are the network layer, the infrastructure or resource layer, the job management layer, and the data management layer. Of course, on each layer Grid middleware technologies provide different components and its important that they can interact possibly via open standard protocols, but the most significant foundation is what we refer to as *'plumbings'*. These plumbings can be seen as orthogonal links into these layers and are concerned with common communication and resource models as well as the common security and information setup. They are essential to achieve infrastructure interoperability beyond the scope of interoperability between one open standard verified in isolation within one layer.

In addition to the foundations from the previous paragraph, the interoperability between e-science infrastructures in terms of job management is achieved by considering the different job types, job descriptions, and even execution environments. Of course all these have to be usable in conjunction with an interoperable job management interface and protocol. The different job types refer to the difference of massively parallel jobs and farming jobs, while jobs that may steer large-instruments such as telescopes are out of scope of this contribution. The requirement of having interoperable job descriptions refers to the description of which applications have to be executed on which resource, possibly using a certain amount of input data. Finally, interoperable execution environments are applied on the resource level in order to get access to certain libraries that have to be used such as Message Passing Interface (MPI) or visualization libraries.

The requirements of the data management layer can be summarized as data transfer, data access, and access to digital repositories and databases. The common data transfer requirement refers to specific protocols that are adopted in each technology to transfer data physically to the resource that computes the job. Data access, in contrast, refers to access data typically identified via logical file names while their actual transport to the compute resource again uses common data transfer mechanisms underneath. Finally, the access to digital repositories and databases includes common protocols to obtain datasets from large database or repositories mostly using meta-data for identification of datasets.

4 Interoperability Approaches

Our work is also influenced from the wide variety of different approaches that enable interoperability of technologies. We organized the International Grid Interoperability and Interoperation Workshop (IGIIW) [39] 2007 and 2008 at the well-known e-science conference series and thus gained a broad understanding of worldwide interoperability efforts and approaches. We shortly outline the known different approaches in this section and give corresponding references to examples that have taken the corresponding approach.

4.1 Additional Layer Concepts

The most famous and thus most common approach is the additional layer concept, which enables interoperability by having a layer with transformation logic on top of different Grid technologies (i.e. Grid middleware). This trasnsformation logic is responsible to change the job description formats and protocols to the corresponding ones supported by the respective middleware. This concept is implemented in Grid portals like GridSphere [38] or APIs like JavaGAT [24] or GPE [19] and thus this additional layer is often located on the client-side. This approach is illustrated in Figure 3.

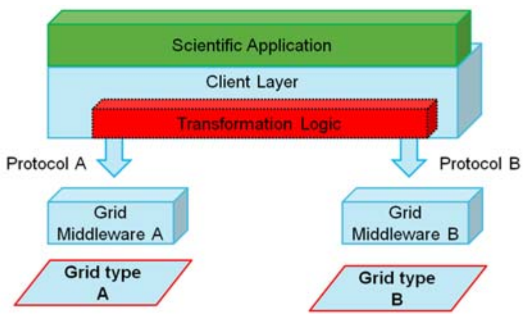

Fig. 3 Additional layer approach.

4.2 Neutral Bridge Approach

The fundamental idea of the bridge approach is to introduce a neutral protocol that can be always used by clients since it is not affected to changes in the Grid middlewares. This neutral protocol is used to contact the neutral bridge implementation, which in turn uses its transformation logic to change the neutral protocol in the different proprietary formats for each of the corresponding Grid middlewares. This approach as shown in Figure 4 is taken to achieve the interoperability between the CORBA-based Integrade middleware and Globus Toolkits as described by Stone et al. in [28].

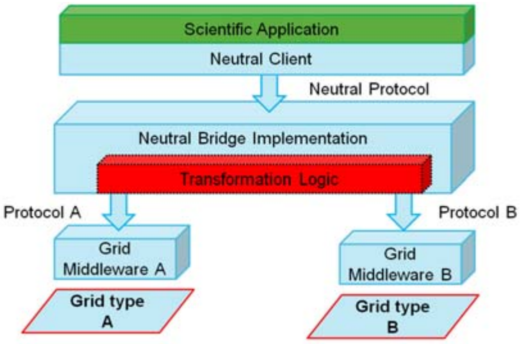

Fig. 4 Neutral bridge approach.

4.3 Gateway Approach

The gateway approach shown in Figure 5 refers to one central entity that is able to translates any middleware protocol into any other middleware protocol using its transformation logic. It is used, for instance, to realize the interoperability between the European infrastructure EGEE and VEGA, which is the Grid Operating System (GOS) for the CNGrid infrastructure in China. Kryza et al. describes in [11] that the interoperability is achieved via a universal interoperability layer named as Grid Abstraction Layer (GAL) that can be seen as one instance of a gateway. By implementing the gateway approach, the GAL not only enables the interoperability between EGEE and VEGA, but also allows for the integration of any other Grid environments.

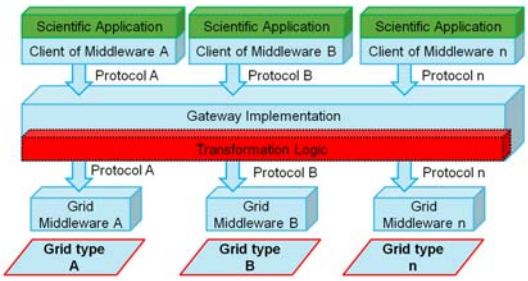

Fig. 5 Gateway approach.

4.4 Mediator Approach

The mediator approach is similiar to the neutral bridge approach, but instead of using a neutral protocol the respective client technology sticks to one specific protocol A and is thus one specialization of the Gateway approach. Protocol A can be used to access all Grid middlewares that natively support this protocol A, but also it can be used to access known mediators. These central mediators are always used via one specific protocol, but are in turn able to translate it into any other protocols with their implied transformation logic. This approach is adopted in the technologies that make EGEE interoperable with BOINC-based infrastructures as described by Kaczuk et al. in [10]. This approach is depicted in Figure 6.

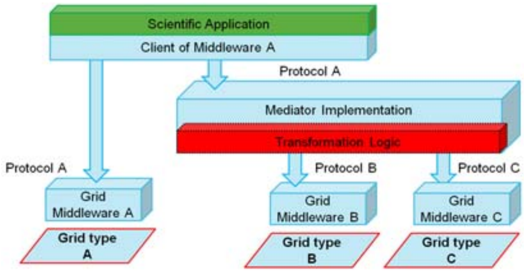

Fig. 6 Mediator approach.

4.5 Adapter Concepts

Another often applied approach is the adapter approach. This means a typical Grid middleware client submits with Protocol A its job to the respective Grid middleware, which in turn, after processing the job description, executes the job or forwards it to a dedicated adapter. This adapter in turn provides the transformation logic that transforms the job into the format of the corresponding other Grid middleware. Hence, the difference to other approaches such as mediator is that the Grid job is actually processed in one middleware stack before being forwarded to another middleware stack B for execution. This approach as shown in Figure 7 is adopted to achieve the interoperability between UNICORE 5 and gLite as described by Riedel et al. in [20].

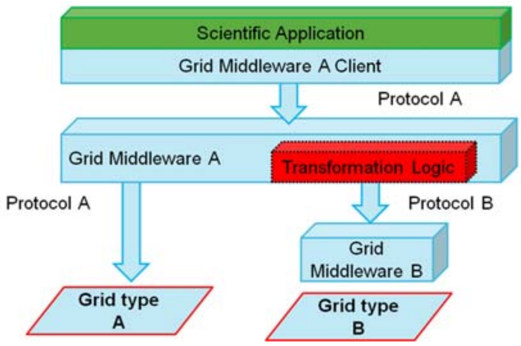

Fig. 7 Adapter concepts.

4.6 Open Standards Approach

The best, but also very ambitous approach to interoperability is the approach to use common open standards. In the past years, many standardization bodies such as the Open Grid Forum (OGF) have worked on open standards, but today we only see a limited number of them deployed on production e-science infrastructures such as EGEE, DEISA, NorduGrid, or TeraGrid. Hence, the interoperability problem would be solved by having all the standards in all key areas of middleware technologies adopted like outlined in the OGSA roadmap [9]. The approach is similiar to the neutral bridge approach, but instead of using a neutral protocol, it uses open standards adopted in the Grid middlewares itself. Needless to say, this approach does not require transformation logic at all and is illustrated in Figure 8.

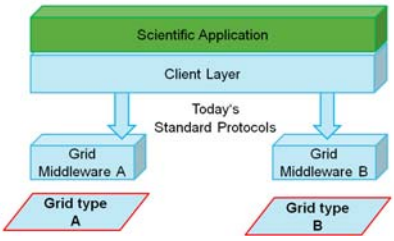

Fig. 8 Open standards approach.

4.7 Middleware co-existence

Finally, although being not exactly a direct solution for the interoperability problem, the middleware co-existence approach basically circumvents this problem by the provisioning of each desired different Grid middlewares with the respective Grid clients for each resource. Hence, this implies a major amount of overhead and is thus rather rarely used, but used in the German national Grid D-Grid that actually deploy UNICORE, gLite, and Globus middleware in parallel. Due to the parallel deployments, a transformation logic is not required. This approach was listed for the sake of completeness in terms of approaches to the interoperability problem and is shown in Figure 9.

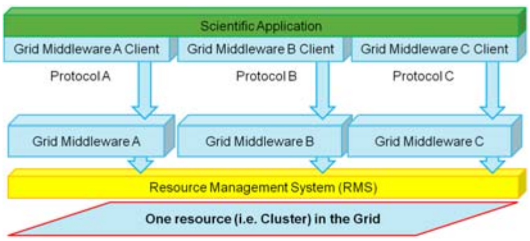

Fig. 9 Middleware co-existence approach.

5 The Interoperability Reference Model

In this section, we present our *infrastructure interoperability reference model (IIRM)* that satisfies the requirements mentioned above. Our design is based on the open standards approach mentioned in the previous chapter, but while plain standards according to the OGSA roadmap are rather top-down defined, we follow an bottom-up approach taking the production experience of a limited amount of technologies and standards as the fundamental drivers. We thus define a simplified reference model including data and job management aspects as well as orthogonal standards in terms of communication, information and security.

It is important to notice that we do not intend to provide the same full functionality of the OGSA and consider the IIRM rather as an economy version of OGSA. Of course, this means we have to initially drop several open standards that are not directly in the focus of the IIRM such as accounting standards (i.e Usage Record Format [13]) or Service Level Agreements (SLA) related specifications (i.e. WS-Agreement [2]). However, we think that the IIRM can be considered as a short-term goal reaching a first broadly available production-level interoperability of infrastructures while the full OGSA conformance remains as a strategic long-term goal. Hence, the IIRM can be seen as an important milestone towards OGSA and does not intend to replace OGSA in its full functionality.

In other words, our approach is clearly driven by the pareto principle, which means that 80% what happens comes out of 20% of our actions. We consider the 80% as the required functionality in Grids that we cover in the IIRM (i.e. job and data management as well as security and information aspects) while we argue that standards for it have been taken 20% of the time for their standardization. In contrast, we argue that the remaining 20% of the required functionality in Grids are rather unique requirements of some communities (i.e. SLAs, Workflow-language standards, deployment standards, etc.) and thus will take the remaining 80% of time in standardization efforts.

The fundamental idea of this section is to provide insights into the core building blocks of the IIRM and its design layout addressing certain essential functionality that basically aims at filling the missing links between numerous open standard specifications. Nevertheless, in the scope of this contribution we can not provide the deep technical details and thus we refer for details to the Production Grid Infrastructure (PGI) working group of OGF that has taken the IIRM as an input in order to standardize its core building blocks that are described in the following paragraph. The PGI working group has

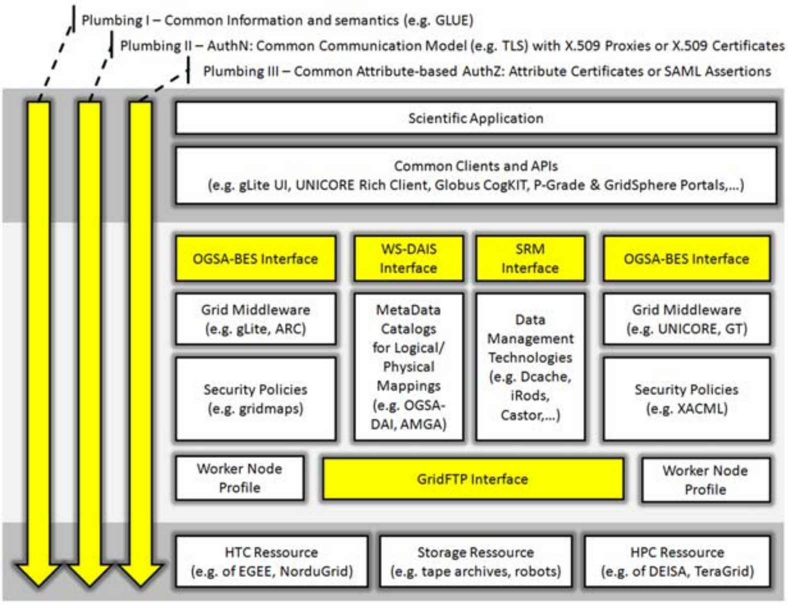

Fig. 10 Core Building Blocks of the Infrastructure Interoperability Reference Model that enables seamless access to different kinds of e-science infrastructures. In this particular example the access to both HTC- and HPC-driven infrastructures.

been launched as a spin-off standardization activity by the GIN community group. Both groups will complement each other and thus GIN proceeds to enable real e-science applications that require resources in more than one production infrastructure demonstrating the technical feasibility of interoperability using open standards wherever possible and thus provides feedback to the PGI group which standards are mature enough to be integrated in subsequent IIRM design versions.

5.1 Core Building Blocks

Figure 10 illustrates the core building blocks of the IIRM design while the lowest layer (i.e. HTC resources, storage resources, and HPC resources) can be considered as the given environment for each IIRM implementation setup. The fundamental idea behind the IIRM is to formulate a well-defined set of standards and refinements of specifications for job and data management that are aligned with a Grid security and information model in order to address the needs of production e-science infrastructures based on the experiences gained in GIN and in other world-wide interoperability efforts. The Web service interfaces and schemas defined in the IIRM are thus basically a set of profiles of open standards and specifications such as OGSA-Basic Execution Services (BES) [8] and thus the implied Job Submission Description Language (JSDL) [3], GridFTP [14], Storage Resource Manager (SRM) [26], and efforts of the GLUE2 working group [37].

The significant contribution of this IIRM is not only the well-defined set of standards as shown in Figure 10, but also the work beyond the current specifications and extensions to focus on the missing links between specifications, which are typically standardized in isolation from each other. Thus, in contrast to the work within the standardization group, we see the illustrated subset of the standard landscape as a whole ecosystem to be adopted by the production e-science infrastructures. We even argue that it is absolutely necessary to take reasonable authentication/authorization models into account that we refer to as *plumbings* within this contribution. These plumbings are a fundamental requirement to promote interoperability in production e-science infrastructures. Another important plumbing is related to the common information provisioning and semantics. All in all, the whole setup is tuned to the use cases found in production e-science infrastructures.

It is important to note that the plumbings are essential in our design process even if this limits the use of the IIRM to only a limited number of scientifically-driven infrastructures, but it provides stability and satisfy today's requirements of e-scientists that would like to leverage the interoperability between e-science infrastructures. All plumbings at once are not necessarily required to achieve interoperability in each IIRM implementation setup. For instance, plumbing I refers to the use of the GLUE information model within the Grid clients as well as within the Grid middlewares. In the context of our work, the GLUE2 elements on the middleware side have to be part of the properties of the

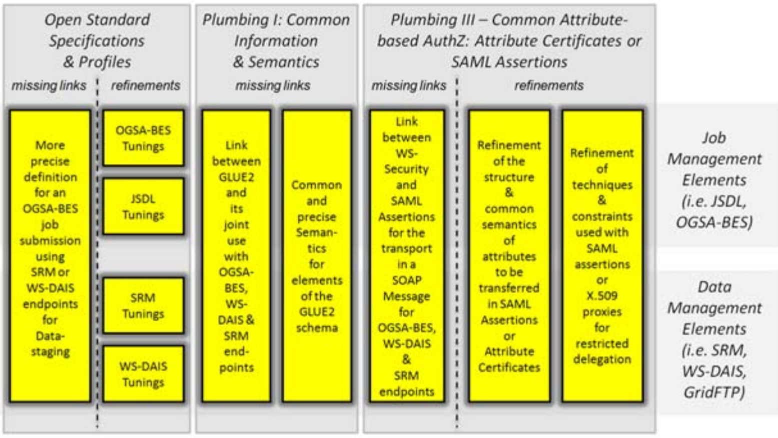

Fig. 11 Overview of the refinements of specifications and missing links we provide as key elements in the IIRM design. Plumbing II has been left out for simplicity since the standards of this plumbing (X.509 or X.509 proxies) do not require any changes. While the standards in terms of job management (i.e. JSDL, OGSA-BES) and data management (i.e. SRM, WS-DAIS, GridFTP) can be considered as 'horizontal' implementations, the plumbings can be seen as vertically integrated standards.

so-called BESFactory Web service portType, which in turn allows Grid clients to obtain the up-to-date information about the corresponding Grid resource.

As stated in the requirements, scientific application from all kinds of scientific domains typically would like to use a common client technology to access e-science infrastructures as shown in Figure 10 using single-sign on even across e-science infrastructures. They are typically using elements of plumbing II (cp. Figure 10) to achieve that. We refer to plumbing II as common communication models, which means the use of full X.509 credentials or X.509 proxies. A wide variety of common clients exist, but for our design it is important that they implement Web services clients to access the job management standard OGSA-BES, which implies the use of JSDL. Hence, OGSA-BES represents an important element of our IIRM and thus we raise the demand that IIRM-compliant Grid middleware have to adopt this standard. At the time of writing numerous Grid middleware systems have already adopted this standard such as gLite (i.e. CREAM-BES), UNICORE 6, and ARC (i.e. A-REX).

The most Grid middleware systems define their own security policy functionalities. While some Grid middleware systems rely on the well known gridmap file, others use open standards such as the eXtensible Access Control Markup Language (XACML) [16]. The realization of these policies is not important since this can be considered as an implementation detail. But the policies are an essential part of the profile since they interact with the defined plumbings III in achieveing a attribute-based authorization.

In more detail, either attribute certificates are used within proxy extensions or signed Security Assertion Markup Language (SAML) assertions [4] are used to convey attributes to the Grid middleware that typically state information such as specific roles, project or membership in a particular virtual organization. Typically such attributes are relased from a dedicated attribute authority such as the Virtual Organization Membership Service (VOMS) [1], or more recently, Shibboleth. That means during an OGSA-BES call (i.e. within the SOAP body of a Web service call) we transfer attributes along the call to enable attribute-based authorization in the Grid middlewares. But where exactly the attributes are transported depends on the use of the different plumbings. They are either encoded as attribute-certificates [31] within the extensions of X.509 proxies on the transport level if plumbing II is used with X.509 proxies, or we transfer attributes as signed SAML assertions in the SOAP header using WS-Security extensions. The latter is possible for both options of plu.mbing II. What matters most is that these attributes are checked at the Grid middleware level with the security policies in order to grant or deny access according to the conveyed attributes of end-users.

Once access is granted the worker node profile is essential since it specifies known environment variables that are helpful during the computational job submission. Information encoded in these variables are for instance, the requested amount of memory for a computational job or the requested amount of CPUs/cores. Finally, the job is executed on the corresponding infrastructure. In the specific deployment of Figure 10 this means that the client can use the IIRM and its well-defined plumbings with a single client in order to access HTC resources in one e-science infrastructure and HPC resources in another infrastructure.

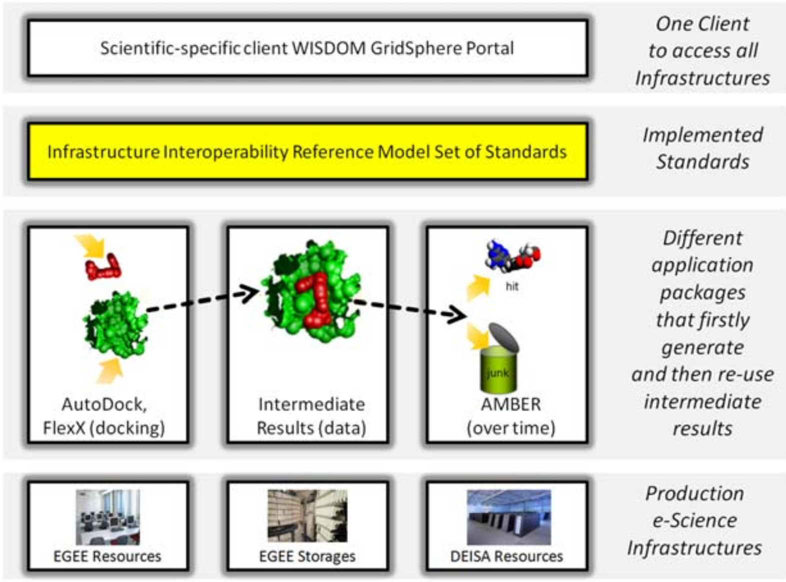

Fig. 12 Accelerate Drug Discovery by using an HTC-based and HPC-driven e-science infrastructures.

5.2 Design Layout and Essential Functionality

Figure 11 shows an overview of the key elements of the IIRM that partly refer to refinements of already existing standards or even tunings when certain standards such as OGSA-BES or SRM have to be changed slightly on the portType level (i.e. add/remove of certain operations) or in schemas like within the JSDL tunings.

The tunings of the OGSA-BES support manual data-staging, which require to obtain the location of the remote job directory and thus also requires to start the submitted activity manually after the manual staging is performed. This require slight changes to the state model provided by the specification as well as changes to some operations (i.e. retrieve locations, manual startup after creation, etc.). Other examples of the OGSA-BES tunings include the support of different computing shares with one OGSA-BES service instance, or a description of endpoints using the information schema GLUE2 instead of the less rich description provided by the specification itself.

The JSDL tunings of our IIRM refer to several changes to the schema. First and foremost, we use GLUE2 schema elements to precisely describe required resources instead of using the less rich resource requirements description of JSDL itself. Other examples of JSDL tunigs refer to the support of several missing features in describing state-of-the-art supercomputer and cluster setups such as multi-threading, network topology, libraries and such like that are all not supported by any kind of JSDL extension specifications.

Similiar to the OGSA-BES tunings, the SRM tunings define a specific subset of supported operations (i.e. SRM functions) that are guaranteed to work across the various different SRM implementations that in turn provide access to different kinds of storage systems (e.g. tapes, disks, etc.). The plumbings refer to the plumbings in Figure 10 and thus are important to fill the missing links as shown in Figure 11. Since this paper focus on research advanced in using the IIRM, a complete description is out of scope, but please refer to the PGI website [40] for more information.

6 Research Advances in e-Science

Today, *e-Health* can be considered as one of the most important research fields that refers to the use of information and communication tools as well as using computational methods to support these tools behind the scene or to support the understanding health fundamentals. It thus plays a significant role in improving the health of world-wide citizens.

In order to proof the feasibility of our proposed IIRM design, we have applied its components to two different real world use cases in the field of e-Health that both raise the demand for interoperability of e-science infrastructures. The first is related to drug discovery and uses the IIRM implementation to conveniently access both the HTC-driven infrastructure EGEE and the HPC-driven infrastructure DEISA. In contrast, the second use case related to bloodflow simulations actually use only hybrid (i.e. NGS) and HPC-driven infrastructures (i.e. DEISA and TeraGrid) in order to get access to a broader variety of HPC systems for running an ap-

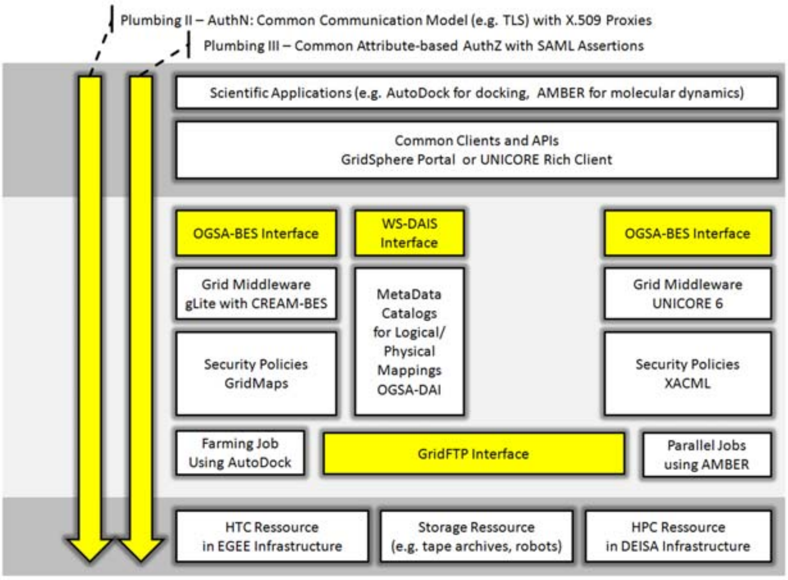

Fig. 13 Accelerate Drug Discovery by using an implementation of the IIRM to seamlessly access DEISA and EGEE.

plication with different scales on different HPC architectures.

Although both examples are basically related to the e-Health research area, the implementation of the IIRM can be also applied to other use cases to enable research advantages in other fields such as within the fusion community. The aim of the fusion community is to research and to establish a long-term safe and environmentally friendly energy source. This is attractive because it uses abundant and delocalized fuel in terms of water and lithium. Hence, without going into too much detail, the fundamental idea is to obtain energy from the union (aka fusion) of light and nuclei. However, this community also raises the demand for a seamless access to HTC-based and HPC-driven infrastructures and we have started to work with members of the EU Fusions for Iter Applications (EUFORIA) project [35] members in order to support their efforts. In short, they would like to use their scientific-specific client named as Kepler workflow tool to access both the HTC-based EGEE infrastructure and the HPC-driven DEISA infrastructure. The project works on porting of several farming codes towards a massively parallel version of it and vice versa. However, the results in the context of this particular projects are still to vague to be included in this paper, but it is surely also an interesting use case of the IIRM design in order to support research advances in other non e-Health related research fields.

6.1 Accelerate Drug Discovery

This section provides insights in latest research advances in using the IIRM to achieve interoperability between two European computing infrastructures. The field of e-Health we address in this context is the drug discovery process. In more details, our approach is based on the so-called in-silico drug discovery process, which uses computational simulations to speed up the identification of potential new drugs. In this context, the SHARE project [41] indicates that pooling knowledge and computer technology together to do in-silico drug discovery can correspond to savings of about 300 million US Dollar and more significantly reduce the development time of a new drug approximately by two years per drug. Hence, the pharmaceutical industry is constantly looking for ways of reducing the time and costs involved in drug development. The interoperability of e-science infrastructures breaks institutional boundaries to help achieve these goals.

We thus can conclude that interoperable e-science infrastructure provide a lot of potential to perform a cheaper and faster drug discovery using in-silico methods. Mentioned research advances in this field have been achieved in collaboration with the WISDOM initiative [21]. WISDOM aims at developing new drugs for Malaria and so far WISDOM scientists have used the EGEE e-science infrastructure for large-scale in-silico docking methods and their scientific workflow. This workflow includes basically two fundamental steps as shown in Figure 12. The first step uses the scientific applications FlexX [36] and AutoDock [33] that are both typically provided on several resources within the EGEE infrastructure and thus accessible via gLite.

But the output of step 1 is only an intermediate result. It is a list of best chemical compounds that are

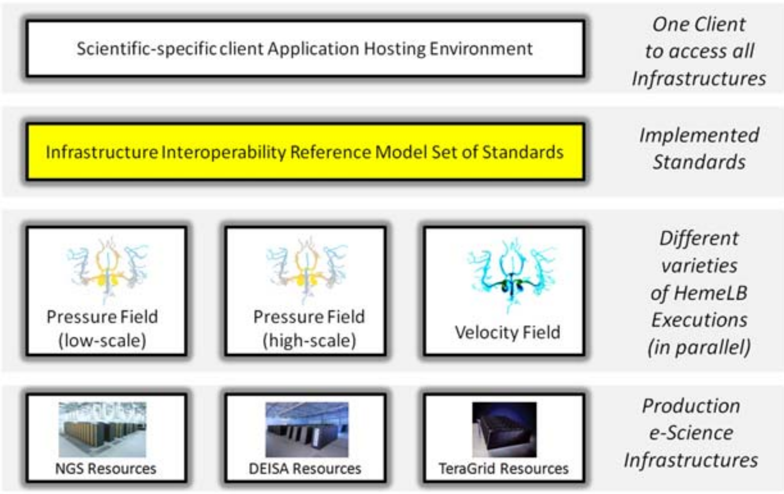

Fig. 14 Different computational simulations of a brain bloodflow using HemeLB.

maybe potential drugs and thus not the final solution to perform the in vitro (i.e. real laboratory tests) and subsequent in vivo (i.e. living organism tests) steps. Therefore, a scientific method developed by Rastelli et al. [18] is to use molecular dynamics (MD) computations to refine thisbest compound list. While this step was so far done on the EGEE infrastructure, there is a lof of potential to use the scalable Assisted Model Building with Energy Refinement (AMBER) [32] MD package within the DEISA e-science infrastructure with highly scalable supercomputers. For more insights into both computational steps please refer to Riedel et al. [21].

Therefore, the fundamental goal of this adoption of the IIRM is to improve the e-science methods in the WISDOM initiative and thus signficantly accelerate the drug discovery process by seamlessly using EGEE in conjunction with DEISA as shown in Figure 13. The illustration also provides insights in the used components of the IIRM. We have shown the implementation of this particular IIRM use case at the Supercomputing 2007 conference in a pre-production setup. In more detail, the OGSA-BES interface of gLite (i.e. CREAM-BES) is used to execute FlexX or AutoDock on the HTC-driven infrastructure EGEE and the results are put on normal storage while its metadata and the link to the storage locations (i.e. GridFTP URIs) are put into a relational database. This database is accessible via an WS-DAIS-compliant specification implementation (tuned). This access is used to obtain in turn the correct GridFTP URIs for transfer into the DEISA infrastructure. Then, on one particular HPC resource within DEISA, the AMBER application is used the previously transferred data (i.e. intermediate results) in order to compute the final compound list, which is considered to go in the in vitro and in vivo steps.

In the meanwhile, we collaborate with the e-scientists to apply for a DECI project [34] proposal in order to get again computational time on the HPC-driven Grid infrastructure DEISA while they are already organized as one VO in EGEE. Hence, we achieved the technical interoperability with the IIRM in this context while the usage policies of actually getting computational time more conveniently on the rarely available HPC resources is still subject to be solved by others.

6.2 Towards The Virtual Phyiological Human

While e-Health is a large scientific research field, we focus in this section on recent work towards the *Virtual Physiological Human (VPH)* [42]. In this context, e-Scientists take already advantage of single e-science infrastructures such as Grids to perform computationally-intensive investigations of the human body that tend to consider each of the constituent parts separately without taking into account the multiple important interactions between them. These subdivisions make it impossible to investigate the systemic nature in which the body functions, however, many e-science applications in this area are limited by the computational power provided in there respective e-science infrastructures while sharing it with numerous other applications of science and engineering.

In contrast, the VPH vision is a methodological and technological framework that will enable collaborative investigations of the human body as a unique complex system. This initiative is part of the larger international Physiome Project that *clearly raise the demand for aligning world-wide interoperable e-science infras-*

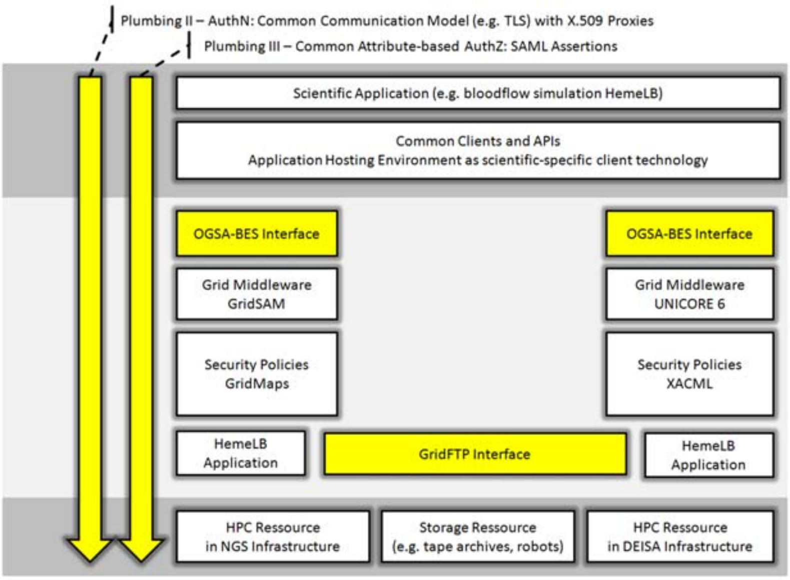

Fig. 15 IIRM implementation among the NGS and DEISA used with the scientific-specific client Application Hosting Environment and the HemeLB bloodflow application.

tructures for collaborative research within its roadmap [27] in order to tackle the computational-intensive challenges that the the simulation of the VPH implies. The VPH community seeks to serve the development and integration of multi-scale models, which have different computational requirements ranging from single processor desktop machines to the largest supercomputers available in *different kinds of e-science infrastructures.*

The particular pre-production setup in this section is using an application in the research field of cardiovascular diseases that are the cause of a large number of deaths in the developed world. The problems of patients are often due to anomalous blood flow behavior in the neighborhood of bifurcations and aneurysms within the brain. In this context, cerebral blood flow behaviour plays a crucial role in the understanding, diagnosis, and treatment of this disease. Thus, the central goal of our application is to simulate this blood flow behaviour using the computer power available on production e-science infrastructures today. The application thus raises a demand for a large amount of computing resources offering different scales, because simulating a whole brain flow is computational-intensive.

In more detail, the GENIUS project is working in this particular field and is mainly concerned with performing neurovascular blood flow simulations in support of clinical neurosurgery. It uses a lattice-Boltzmann code named HemeLB [15] designed to simulate fluid flow in the sparse topologies of the patient brains. Notice that the simulation models are typically derived from patient-specific brain x-ray angiography scans are in turn used as input to the simulator. As a consequence, our infrastructure setup actually requires the possibility for large file transfers that are able to transport the x-ray-based data to the computing resources across boundaries of existing e-science infrastructures. The latter is specifically important to circumvent a duplicate storaecentlyge of these large datasets in each different e-science infrastructure that is used.

Figure 14 indicates another key requirement of our application that is related to the seamless access of different infrastructures. The e-scientists typically use their own specific clients that in our case is named as the Application Hosting Environment (AHE). The fundamental goal of this tool is to allow clinicians to seamlessly interact with a large amount of computational power available in different e-science infrastructures even from within their operation theatre. This specific use case implies the interactive access to resources of these infastructures as well in order to perform computational steering in real-time. In this context, computational steering refers to the change of application parameters on the fly during the application execution on one of the resources within an e-science infrastructure. Hence, the goal of these real time visualization and computational steering is to allow clinicians to interact with the simulations as they run in order to review the possible effects of various surgical investigations.

Figure 15 illustrates the pre-production setup of the use case enabling the AHE to submit jobs, in this particular example, to the NGS and DEISA.

7 Related Work

As already mentioned in the introduction of this paper, the rather high-level OGSA initially defined by Foster et al. [7] defines an architecture model taking many requirements from e-science and e-business into account and develops rather top-down standards to satisfy them. In contrast, our work is motivated by lessons learned from e-science infrastructure efforts that raise the demand for a reference model that is trimmed down in functionality compared to OGSA, is more specific but less complex than OGSA, and thus more realistically to implement. Hence, the IIRM, in contrast to OGSA, is rather a bottom-up approach to standardization.

A more detailed comparison between OGSA (including its roadmap [9]) and the IIRM identifies basically four major differences. First, the IIRM is less complex in order to address the rather long standardization processes in standards bodies. To provide an example, the OGSA-BES standard took 36 iterations to be completed as a full standard over a duration of approximately two years. We only take a subset of currently available standards. Second, OGSA is not specific enough with respect to e-science infrastructures since it takes a huge requirement set into account. Most notably, it does not address the standard refinements or missing links between several specifications that have been provided by the IIRM for a specific subset of open standards. We argue that the standardization process according to the OGSA roadmap is difficult since the job standards do not address data issues while the data standards do not address job issues, and additionally, both job and data standards declare that security is out of scope. We argue that this produces rather isolated standards, instead of well-linked standards that have been defined by the IIRM design by also specifiying the missing links between important standards (i.e. SRM data staging with OGSA-BES secured via the specified security plumbings).

As a third difference, we argue that the many components that are basically part of the OGSA roadmap raise economic issues. At the time of writing, many e-science infrastructures struggle to have a sustainable strategy in terms of maintenance cost and thus commercial options are considered as one way. In other words, we argue that a full grown OGSA conform technology with all dedicated services might be actually too expensive to maintain over a long period. IIRM, in contrast, only focusses on a small subset and can be thus considered to be an economic version of OGSA. Finally, the fourth difference relies in the bottom-up approach of the IIRM compared to OGSA. This means we know already from the production experience that the standards defined by the IIRM are working while a full grown OGSA conform technology has still to perform a lot of test runs before many open standards will be deployed on production e-science infrastructures.

8 Conclusions

In this paper, we raised the demand for an infrastructure reference model to promote interoperability between todays production e-science infrastructures. We have shown that the elements of our proposed reference model are based on experiences and lessons learned from many world-wide interoperability projects. We can conclude that the IIRM represents a trimmed down version of OGSA in terms of functionality and complexity, but still providing the most significant features used in e-science infrastructures over the last couple of years. We have shown the core building blocks of the IIRM and many of them are already deployed on the infrastructures and only minor changes (i.e. missing links and refinements) have to be done in order to achieve interoperability in production e-science infrastructures.

In order to evaluate our IIRM design, we also gave insights into two e-Health related use case applications. We have presented pre-production setups of these IIRM implementations as life demonstrations at the Supercomputing 2007 and 2008 respectively. We can conclude that these IIRM implementations satisfy the demand of the scientists by providing access to multiple interoperable infrastructures and thus provides assurances that an IIRM-based client can access their required multiple infrastructures by still using single sign-on and the same security credentials. This in turn allows for new research advances in their particular research field fundamentally supported by the wide variety of e-science infrastructures that exist today.

Since our evaluation use cases have been very successful, we have given the IIRM as an input to OGF by creating a GIN spin-off activity named as the Production Grid Infrastructure (PGI) working group. By chairing this group, our goal is to standardize the IIRM elements and plumbings to assure that also numerous other Grids can benefit from our proposed IIRM design. With having participants from many important e-science infrastructures such as DEISA, EGEE, NGS, NorduGrid and their middleware providers UNICORE, gLite, OMII-UK, and ARC, we are looking forward to get the core building blocks of our proposed IIRM design standardized very soon. This will significantly contribute to the vision of having an *interoperable united federation of world-wide e-science infrastructures* in the future offering standardized PGI-compliant access.

Acknowledgements The work presented here is related to efforts of many talented computer scientists and e-scientists that are funded by various public funding organizations that we thank and acknowledge, because this work would not have been possible without their continous and sustainable support. We also want to express our gratitudes to the members of the OGF Grid Interoperation Now community group and the OGF Production Grid Infrastructure working group. The work and discussions in these groups significantly supported the progress and adoption of the IIRM in many different ways. We also especially thank the e-scientists of the WISDOM team that include J. Salzemann, A. Da Costa, V. Bloch, V. Breton, M. Hofmann-Apitius, and, most notably, V. Kasam. In the context of the scientific use case of Grid-enabled neurosurgical imaging we are deeply thankful to P. Coveney, S. Manos, and S. Zasada. This work is partly funded via the European project DEISA-II that is funded by the European Commission in FP7 under grant agreement RI-222919. Also, we thank the application support team in Jülich in supporting our efforts in demonstrating the IIRM use cases at the Supercomputing 2007 and 2008. Our final thanks go to the Forschungszentrum Jülich of the Helmholtz association in general and the Jülich Supercomputing Centre in particular.

References

1. Alfieri, R., et al.: From gridmap-file to voms: managing authorization in a grid environment. In: Future Generation Comp. Syst., 21(4):549-558 (2005)
2. Andrieux, A., et al.: Web Services Agreement Specification (WS-Agreement). OGF Grid Final Document Nr. 107 (2007)
3. Anjomshoaa, A., et al.: Job Submission Description Language (JSDL) Specification, Version 1.0. OGF Grid Final Document Nr. 136 (2008)
4. Cantor, S., Kemp, J., Philpott, R., Maler, E.: Assertions and Protocols for the OASIS Security Assertion Markup Language. OASIS Standard (2005). Http://docs.oasis-open.org/security/saml/v2.0/
5. Chandra, R., et al.: Parallel Programming in OpenMP. Morgan Kaufmann (2001). ISBN 1-55860-671-8
6. Foster, I.: Globus Toolkit version 4: Software for Service-Oriented Science. In: Proceedings of IFIP International Conference on Network and Parallel Computing, LNCS 3779, pp. 213–223. Springer-Verlag (2005)
7. Foster, I., et al.: Open Grid Services Architecture (OGSA), Version 1.5. OGF Grid Final Document Nr. 80 (2006)
8. Foster, I., et al.: OGSA Basic Execution Service Version 1.0. OGF Grid Final Document Nr. 108 (2007)
9. Jordan, C., et al.: Defining the Grid: A Roadmap for OGSA Standards v.1.1. OGF Grid Final Document Nr. 123 (2008)
10. Kacsuk, P., et al.: Towards making BOINC and EGEE Interoperable. In: Proc. of the IGGIW Workshop, e-Science Conference 2008, Indianapolis, USA (2008)
11. Kryza, B., Skital, L., Kitowski, J., Li, M., Itagaki, T.: Analysis of Interoperability Issues Between EGEE and VEGA Grid Infrastructures. Springer-Verlag (2006)
12. Laure, E., et al.: Programming the Grid with gLite. In: Computational Methods in Science and Technology, pp. 33–46 (2006)
13. Mach, R., et al.: Usage Record - Format Recommendation. OGF Grid Final Document Nr. 98 (2007)
14. Mandrichenko, I., et al.: GridFTP v2 Protocol Description. OGF Grid Final Document Nr. 47 (2005)
15. Manos, S., et al.: Patient specific whole cerebral blood flow simulation: A future role in surgical treatment for neurovascular pathologies. In: Proc. of the TeraGrid 2008 Conference, Virginia, USA (2008)
16. Moses, T., et al.: eXtensible Access Control Markup Language. OASIS Standard (2005)
17. Pacheco, P.: Parallel Programming with MPI. Morgan Kaufmann (1996). ISBN 1558603395
18. Rastelli, G., et al.: Validation of an automated procedure for the prediction of relative free energies of binding on a set of aldose reductase inhibitors. In: Bioorg Med Chem 15(24) (2007)
19. Ratering, R., et al.: GridBeans: Supporting e-Science and Grid Applications. In: 2nd IEEE International Conference on e-Science and Grid Computing (E-Science 2006), Amsterdam, The Netherlands (2006)
20. Riedel, M., Laure, E., et al.: Interoperation of World-Wide Production e-Science Infrastructures. In: Journal on Concurrency and Computation: Practice and Experience (2008)
21. Riedel, M., et al.: Improving e-Science with Interoperability of the e-Infrastructures EGEE and DEISA. In: Proceedings of the MIPRO (2007)
22. Riedel, M., et al.: Classification of Different Approaches for e-Science Applications in Next Generation Computing Infrastructures. In: Proceedings of the Int. e-Science Conference, Indianapolis, USA (2008)
23. Riedel, M., et al.: Experiences and Requirements for Interoperability between HTC- and HPC-driven e-science Infrastructures. In: Proceedings of the Workshop HPC for e-Science, Korea e-Science All Hands Meeting (2008)
24. Rob, V., et al.: User-friendly and Reliable Grid Computing Based on Imperfect Middleware. In: Proceedings of the International Supercomputing Conference 200, Reno, USA (2007)
25. Rodero, I., et al.: Modeling and Evaluating Interoperable Grid Systems. In: Proc. of the IGGIW Workshop, e-Science Conference 2008, Indianapolis, USA (2008)
26. Sim, A., et al.: The Storage Resource Manager Interface Specification Version 2.2. OGF Grid Final Document Nr. 129 (2008)
27. STEP-Consortium: Seeding the EuroPhysiome: A Roadmap to the Virtual Physiological Human (VPH). Http://www.europhysiome.org/roadmap
28. Stone, D., et al.: A Model for Transparent Grid Interoperability. In: Proc. of the CCGrid 2007 Conference (2007)
29. Streit, A., et al.: UNICORE - From Project Results to Production Grids. In: L. Grandinetti (ed.) Grid Computing: The New Frontiers of High Performance Processing, Advances in Parallel Computing 14, pp. 357–376. Elsevier
30. Taylor, J.: enhanced-Science (e-Science) Definition. Http://www.e-science.clrc.ac.uk
31. V. Ciachini V. Venturi, A.C.: The VOMS attribute certificate format. Technical Report, OGSA Authorization Working Group (2005)
32. Website: AMBER. Http://amber.scripps.edu
33. Website: AUTODOCK. Http://autodock.scripps.edu
34. Website: DECI. Http://www.deisa.org/grid/initiative.php
35. Website: EUFORIA - Project. Http://www.euforia-project.org
36. Website: FLEXX. Http://www.biosolveit.de/FlexX
37. Website: GLUE2 Open Grid Forum Grid Forge Page. Https://forge.gridforum.org/sf/projects/glue-wg
38. Website: GridSphere. Http://www.gridsphere.org/
39. Website: International Grid Interoperability and Interoperation Workshop (igiiw). Http://www.fz-juelich.de/jsc/igiiw
40. Website: OGF - Production Grid Infrastructure Working Group (PGI-WG). Http://forge.ogf.org/sf/projects/pgi-wg
41. Website: SHARE - Project. Http://www.share-project.org
42. Website: Virtual Physiological Human (VPH). Http://www.europhysiome.org