World Academy of Science, Engineering and Technology
International Journal of Bioengineering and Life Sciences
Vol:2, No:6, 2008

Grid Computing in Physics and Life Sciences

Heinz Stockinger
(Invited Paper)

Abstract**—** Certain sciences such as physics, chemistry or biology, have a strong computational aspect and use computing infrastructures to advance their scientific goals. Often, high performance and/or high throughput computing infrastructures such as clusters and computational Grids are applied to satisfy computational needs. In addition, these sciences are sometimes characterised by scientific collaborations requiring resource sharing which is typically provided by Grid approaches. In this article, I discuss Grid computing approaches in High Energy Physics as well as in bioinformatics and highlight some of my experience in both scientific domains.

Keywords**—** Grid computing, Web services, physics, bioinformatics

PREFACE

This article is "a written version" of an invited plenary talk at the Third International Conference on Communication Technology ICCT'06, 16 December 2006, Vienna, Austria, *organised by the World Enformatika Society. The conference has a rather broad scope in several fields of communication technology ranging from communication networks over wireless communication technologies to communication software. However, Grid and distributed computing are specific domains in computer science. Therefore, this article is intended for a broad audience that does not necessarily have a background in Grid computing. I will provide a general background on Grid computing and my experience with applying it to High Energy Physics as well as bioinformatics.*

I. INTRODUCTION

THE terms "Grid" and "Grid computing" have been coined by Ian and Carl Kesselman in their book "The Grid, Blueprint for a new computing infrastructure" [10]. Although the concept of meta-computing and distributed computing that spans several sites and administrative domains has been around for much longer (for instance the Condor project had early ideas and implementations already dated back to the late 1980s [15]), the era of Grid computing with a specific name and its own dedicated research and user community started in the late 1990s. This is also when many Grid research projects have been initiated and early foundations for Grid computing have been made.

In the late 1990s the World Wide Web was already very well established and heavily used not only in academia and industry but also in many households. Some early Grid visionaries were expecting and foreseeing a similar success for the Grid as we have seen for the Web. In late 2006, about a decade after the initial steps, the main euphoria has already passed, and we are now more in a phase of consolidation and of turning original research visions into solid, science-driven applications. This is particularly important for application scientists in physics or biology that want or need to use Grid technologies for advancing their scientific goals. In this article I will discuss some of the requirements of such applications, what has been achieved so far and what next steps are still required.

Heinz Stockinger is a Researcher at the Swiss Institute of Bioinformatics (Vital-IT Group), Quartier Sorge, Bâtiment Génopode, CH-1015 Lausanne, Switzerland, e-mail: Heinz.Stockinger@isb-sib.ch

The article focuses on my personal experience with Grids in High Energy Physics (HEP) at CERN as well as bioinformatics in the context of several multi-national and national projects. Both scientific domains are very representative application areas due to their requirements on data and storage. In fact, the HEP use case is often considered to be *the* example Grid use case. This is also visible in the number of current Grid projects that are set up around HEP. Bioinformatics and biomedical applications have typically more stringent requirements on data security and privacy (sensitive data) than HEP applications, and are therefore important to evaluate if current Grid solutions satisfy concerns about data and resource sensitivity.

II. GRID COMPUTING

In a recent survey I invited many Grid scientists to define the Grid and what they think the current status was [21]. Whereas the Grid research community has a rather homogeneous view on what Grid computing is and where the boarders are with respect to Internet and distributed computing, this view is not always shared in the general computer science community or in the business world. The main differences are observed in hardware and middleware. For instance, for some people a cluster on a local area network is already considered a Grid whereas others think there need to be several administrative domains involved rather than only a single site.

Despite these "subtle" differences, there are a few aspects that make Grids "special", i.e. distinctive and a research domain on its own. Among the most important factors are resources sharing, virtualisation, abstraction to resource access and standardisation in order to make different Grid implementations interoperable. The last item is directly connected to the fact that we are now in a state of research where we are turning original research visions into *working solutions*. This is particularly important to make the Grid vision palpable to a large user community.

Some people might simply say that "the hype about Grid computing has gone" which is a normal situation of any new technology or paradigm shift. If we look at the "hype cycle" [11] of Grid computing (depicted in Figure 1) we can see that the main attention/attraction has passed but we are still in a phase where technologies advance and become

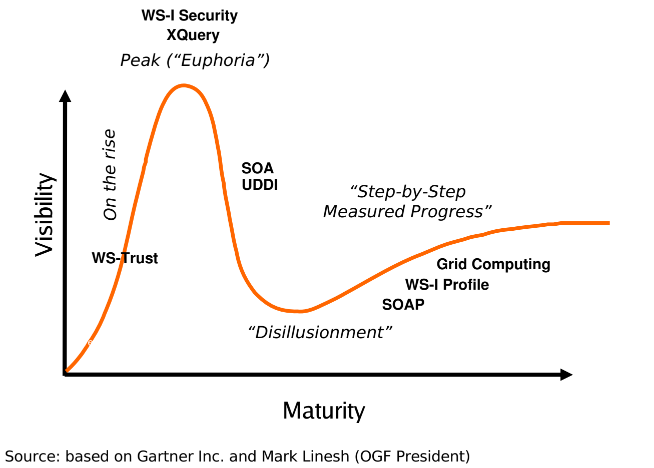

Fig. 1. Hype cycle for Grid computing and some related standards according to Gartner Inc. [11].

more mature. For instance, if we look at the number of multi-national Grid projects funded by the European Commission in the 6th Framework Programme, this tendency is also visible there.

Another interesting aspect is also the way how funding agencies are currently promoting Grid technologies. Whereas in the 6th Framework Programme Grid computing was mentioned in many places and funded via several sources, in the 7th Framework Programme the term Grid does not explicitly appear so often but is implicitly mentioned. Again, a point in the direction that Grids are more and more considered to be a technology provider for existing applications. A similar trend is also visible in North America where the term **cyberinfrastructure** is now used much more often to refer to Grid technologies whereas in Europe the term **e-Science** is still more dominant.

For several years the Grid community has been working on standardisation and architectural work. In parallel, Web services and service oriented architectures have been adopted and are common building blocks of modern Grid middleware solutions. In this way, the Grid community follows a general main stream that is currently visible in computer science and IT. This should allow for more interoperability and acceptance of produced software solutions on the one hand, but it also becomes more difficult to clearly distinguish the boarders of Grid computing on the other hand.

The latest Grid standardisation effort resulted in the Web Services Resource Framework (WSRF) [24] that enhances general Web services by several architecturally important points such as transactional state, service notification, service lifetime management etc. The overall goal is to have a common architecture that follows the Open Grid Services Architecture (OGSA) [9]. This standardisation effort is constantly evolving as more industry partners join the Open Grid Forum (OGF) [17]. This results in frequent updates and modifications to the draft specification of the standard. Just recently another Web services based standardisation effort has started by a small group of leading IT companies: the ideas are rather similar to the WSRF specification but should get more agreement from major IT providers. In the end, standardisation is all about agreement with the aim to make services interoperable and supported by both, the academic world as well as the commercial IT industry.

In the early days of Grid computing there was often the discussion about *the* Grid, referring to a single, world-wide Grid. Over the years we have seen that this vision has not been reached due to several reasons. The most dominant one is that several projects and organisations create their *own* Grids (using different implementations, standards and protocols) and give access only to a certain, registered user community that works in a certain application domain such as High Energy Physics, engineering etc. Although at the first thought this looks like the creation of several "virtual private Grids" rather than a single Grid, it seems to be rather natural that organisations do not want to share their resources with "everyone else" otherwise the resources would soon get sparse. For example, imagine a computer centre with a cluster of 512 processors: if free access would be given to everyone, a single user group would have only very limited CPU capacity. As a counter argument one can say that people should *pay* for that service if they want to use it. This is a very common aspect in the context of national supercomputing centres that typically charge for CPU usage. Furthermore, certain IT providers sell CPU time and storage resources on demand. In summary, it is very likely that Grid implementations will include a similar charging model for resource utilisation. The first steps into this direction have already been proved successful.

In conclusion, a single, free Grid infrastructure (such as the Web) might not be available at all due to the constraints on resource ownership. However, the Grid community is moving towards the direction of a few standards that will allow to build a Grid infrastructure that connects many of the currently non-interoperable "private Grids".

III. Use Case: High Energy Physics

After some background and theoretical aspects in Grid computing let us now look at actual implementations. We start with a High Energy Physics use case since this is one of the most demanding one in both computational as well as storage management aspects.

A. *CERN and the Physics Challenges*

CERN, the European Organization for Nuclear Research, has a long tradition in dealing with computing challenges in the past. The overall goal of CERN and its High Energy Physics experiments is to study the origin of matter, i.e. fundamental research in physics. In order to do so, large particle detectors with a corresponding particle accelerator are used to create conditions similar to the Big Bang. In the detector, particle collisions, tracks and decays are studied. An example of a particle detector that is used by one of CERN's experiments is given in Figure 2.

There are several challenges in establishing and operating High Energy Physics experiments. Since the physics apparatus that are created and used are very expensive, a single institute

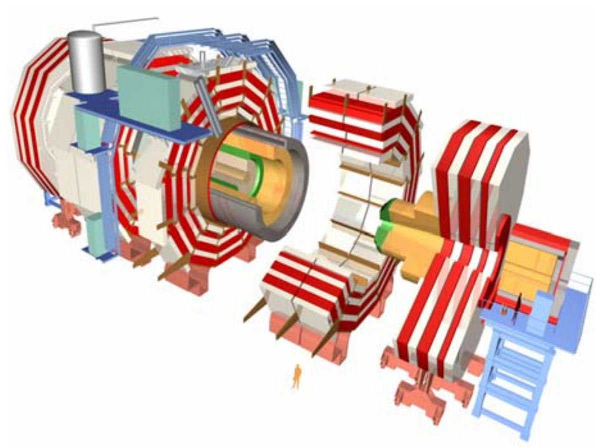

Fig. 2. The particle detector of the Compact Muon Solenoid (CMS) experiment at CERN. Particle collisions take place in the centre of the detector. After some filtering of physics events, data needs to be written to storage devices with a speed of 100 - 200 MB/s for a sustained time of more than 100 days a year which results in several Petabytes of data per year.

can typically not afford (neither financially nor scientifically) to build nor operate such devices on its own. Therefore, physicists create big collaborations of many institutes and universities all over the world in order to share knowledge, hardware, software and also financial costs. In Grid terminology each physics experiment that forms a collaboration corresponds to a virtual organisation and almost ”perfectly” fulfils the basic Grid concepts of resource sharing, virtualisation and transparent access. Additionally, the participating institutions are distributed worldwide and complement the computing and storage resources of CERN. Therefore, a distributed, multi-tier computing model has been built that takes this distributed sites into account. Grid middleware and services are used to orchestrate the job submission to the remote sites as well as the data exchange and transfer involved to analyse physics data.

B. The Web and the Grid

In the computer science community CERN is mainly known for the Web which was inspired by previous HEP experiments and similar computing challenges in the late 1980s. Tim Berners-Lee’s work was an important spin-off rather than a mainstream physics effort when he was starting the “revolution” of the Web by leading the first HTTP standard and the implementation of the first Web servers. This invention was driven by the need to create a standard way to display information on heterogeneous hardware and software environments that were used by physicists, computer scientists and engineers.

Today (i.e. in the year 2006) there is not such a diversity of hardware and software platforms anymore since main stream processor architectures are rather standardised. The same is true for operating systems: only a rather small number of UNIX operation systems is available, and there is an effort in the scientific community to create a common Linux distribution called “Scientific Linux” [20]. In fact, most of the regional centres in the various tiers in CERN experiments use the same

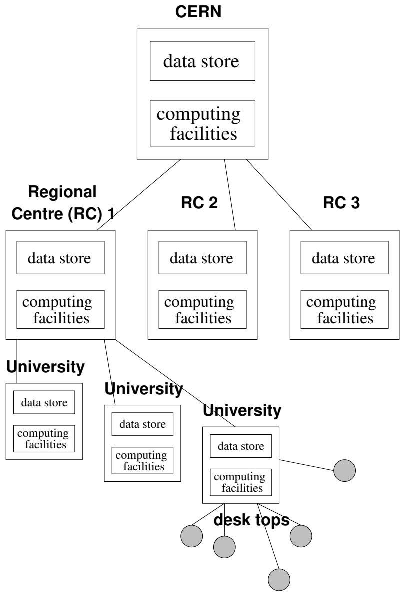

Fig. 3. The typical multi-tier computing architecture of a High Energy Physics experiment. CERN is the Tier 0 centre where the main physics raw data is created and stored. Raw data needs to be reprocessed using computing intensive algorithms that typically run in Tier 1 sites (Regional Centres) which also have large storage resources. End user physics analysis is done at each tier in the model, and data needs to transferred between the sites.

operating system in order to allow for a simplified and more controlled computing environment that needs to scale to a large user community. Therefore, standardisation at the hardware as well as the operating system level provided one more way in the direction of ”global computing”.

However, the amount of data to be processed and stored has increased tremendously since the early Web days, and there is now the need to standardise on resource access and data management which is the aim of a Grid computing model. Therefore, the Grid standardisation effort can be directly compared with the Web standardisation effort with the main difference that the Web was originally created to share *information* whereas the Grid has been created to share all kinds of *resources* from computing cycles, to storage as well as other scientific devices and apparatus.

C. High Energy Physics and the Grid

Up to now I mainly focused on High Energy Physics experiments at CERN which have collaborators all over the world. There are also other important HEP centres such as Fermilab near Chicago (USA), the Stanford Linear Accelerator Center (SLAC) in California, KEK in Japan, DESY in Germany etc. Although they are smaller in size, they have comparable computing requirements and hence similar computing models with similar Grid approaches. However, in the remainder of this article we concentrate on CERN experiments since they are typically slightly more challenging in terms of computing and storage capacity although the real start of these experiments is only foreseen for end of 2007/beginning of 2008.

The High Energy Physics community and in particular CERN has inspired a number of multi-national Grid projects that have been created in order to support the multi-tier computing model. One of the biggest Grid projects (operating the world's biggest Grid infrastructure) is the **EGEE** [5] project: it connects more than 150 sites, offering more than 20,000 CPUs and more than 10 Petabytes of storage. EGEE currently provides two versions of the Grid middleware that can co-exists and are interoperable (the **gLite** [12] and the **LCG** middleware [14]). Such a big infrastructure is made possible by connecting sites with partner projects from Europe (**NorduGrid** [16] with the **ARC** [6] middleware) as well as from North America via the **Open Science Grid (OSG)** [18] which uses a middleware system based on the **Virtual Data Toolkit (VDT)** [23].

In this article, we do not intend to go into the details of the different middleware flavours but only point out the main building blocks that they have in common to realise a distributed Grid infrastructure.

D. Basic Grid Components

The main building blocks or services are categorised below. The concepts are used in all of the three Grid middleware systems introduced in the previous subsection.

- **Computing Element**. The term Computing Element is an abstraction for any kind of computing resource such as a cluster, a supercomputer or just a single machine. Typically, a Computing Element provides a standard Grid interface to hide the complexities of the underlying computing hardware and software.
- **Storage Element**. A Storage Element is the equivalent concept for a storage system. It provides an abstraction for several different storage systems such as file systems, mass storage systems etc. Currently, Storage Elements often provide a file based interface in contrast to data stored in relational databases or object databases.
- **Replica Catalogue**. Data is distributed and often replicated to several sites for fault tolerance and availability reasons. A replica catalogue is a special purpose metadata service that is used to locate (file) replicas in the Grid.
- **Information System**. The Information System keeps track of all services in the Grid. It is used for discovery of services such as Computing Elements, Storage Elements, and other higher level services and applications. Additionally, the Information System is sometimes used to monitor the status of services as well as of other higher level Grid applications.
- **Resource Broker**. End users or high-level services typically do not directly interact with all Grid services. It is therefore the task of a Resource Broker to discover computing resources via the Information System and dispatch jobs to suitable locations. Resource Brokers can either be services on their own or directly available as part of a client tool for job submission.
  Note that the term Resource Broker is an over simplifications since the implementation of such a component typically consists of the following sub-components: resource discovery and match making, workflow management and job execution.

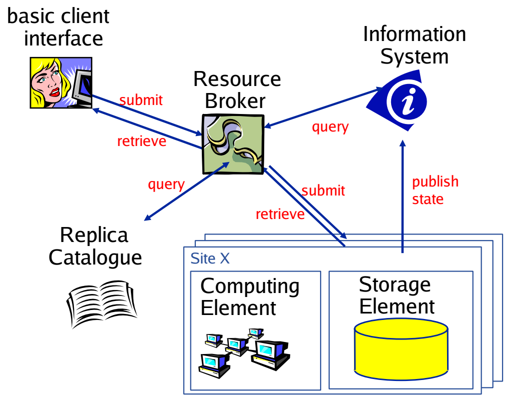

Fig. 4. A simplified Grid infrastructure with the interaction of the services. The end user typically uses the client interface to the Resource Broker in order to submit jobs to the Grid. However, direct interaction with all other services such as Storage Elements, Information System etc. is possible. All service interaction is usually done in a *secure* way where both users and services are authenticated using X.509 Grid certificates or similar authentication and authorisation schemes.

A simplified interaction between the basic services is given in Figure 4. The end user does not need to know where all services are located but mainly uses a command line or GUI/Web based interface to access Grid resources.

E. Usage in Physics Experiments

The four current LHC experiments at CERN have all built computing models that use Grid concepts and technologies to support job submission as well as data transfers and replication as reported in the respective Computing Technical Design Reports (TDR) of the experiments (as an example refer to the CMS TDR [4]). This shows the current success of Grid technology and its penetration in the field of HEP.

I was personally involved in the CMS experiment from the "first days of Grid computing" at CERN in early 2000. At this time, CMS as well as other HEP experiments, were among the early testers and customers of the Grid vision and the early prototype software. The experimental physics community has shown a good level of understanding and coping with early bugs etc. This allowed include the HEP community into several software release cycles and present them with new concepts etc. Much of our accumulated experience within the CMS experiment and Grid tools is reported in [3].

IV. Use Case: Bioinformatics

Whereas High Energy Physics deals with sub-atomic particles and interactions, biology deals with higher level "building blocks" of life: molecules, proteins, genes, DNA sequences etc. Let us now "zoom out" from the micro cosmos of High Energy Physics into the domain of bioinformatics and molecular biology.

A. Basic Bioinformatics Challenges

Bioinformatics itself is a rather broad field and basically deals with (the creation of) biological databases and the corresponding tools to analyse these databases. The most popular databases are genome databases that record either DNA or protein sequences (basic building blocks of live) of different species. Similar to HEP, several of the applications are very computing intensive and might require parallel algorithms in order to speed up the analysis of biological databases. Some of the problems are partially embarrassingly parallel (such as the sequence search and alignment process of DNA and protein sequences; drug docking etc.) and can be mapped to clusters or Grid systems.

One of the most popular sequence analysis tools is BLAST [1] that can be very CPU intensive if big databases are searched. If many people submit BLAST jobs at the same time, there is a need for a high throughput computing infrastructure that distributes the workload to many different machines in order to improve the overall throughput for the bioinformatics community.

Other fields in bioinformatics that deal with protein folding sometimes use massive parallelism rather than embarrassing parallelism. These types of problems can be put on clusters with software libraries such as MPI or PVM but are not so well suited for wide-area network links due to the different latencies of nodes that might be involved in the execution of the task.

Although the discovery and the creation of genome data or even new drugs is very cost intensive and is supported by bioinformatics algorithms, the bioinformatics community does not have these “centres of gravity” as we can find them in High Energy Physics. One of the early connecting ”institutions” is the EMBnet [7], which represents a network of molecular biologists mainly within in Europe. Furthermore, there are certain organisations that work on specific topics such as protein structure databases etc. As a result, the bioinformatics community works relatively more independent than the HEP community since connecting factors such as large virtual organisations are not (yet) in place. Therefore, Grid computing ideas and principles are differently perceived in the different communities.

B. Bioinformatics and the Grid

Bioinformatics and in particular biomedical applications have been part of EGEE and its predecessor project from the very beginning. Recently, the bioinformatics community has started with its own national and multi-national projects such as the EMBRACE [8] Grid project mentioned below in more details. A detailed list of projects is not the aim of this article, but I will mainly focus on projects that I am involved or where I have some experience with.

The **EMBRACE** Grid project is funded by the European Commission as a Network of Excellence in order to connect bioinformaticians throughout Europe. In particular, the communities that provide databases are linked with the communities that provide bioinformatics software tools. The main aim is to integrate software tools and biological databases using current state-of-the-art Web services and/or Grid technologies where appropriate. This should help ”embrace” the European bioscience community and provide them standards and common directions for integrating current and future services [8].

C. Some Results on Biological Sequence Analysis

Let us now dive a bit deeper into the field of sequence analysis within the framework of the EMBRACE Grid project. I will give some basic overview of our work on parallelising and ”gridifying” (a term commonly used in the Grid community) existing single processor code for sequence analysis [22].

Rather than using the common tool BLAST for sequence analysis, we use an approach that is based on Hidden Markov Models [13]. In particular, we use existing bioinformatics tools that are either designed for single processors or clusters rather than for Grids. We have built as system that wraps around existing code, ”extracts” the parallelism by creating data chunks which can then be executed in parallel on a Grid infrastructure or also on a single cluster. We designed our tool called `wg` (Workload Generator) for various Grid flavours (EGEE/gLite and NorduGrid/ARC) as well as clusters and utilised existing brokering techniques of the different middleware solutions. Originally, all data chunks had equal sizes in order to execute similar tasks on the Grid. However, due to the heterogeneous nature of hardware in the Grid, different execution times were achieved which resulted in non-optimal execution times for the overall sequence search and analysis process.

Consequently, we created a Run Time Sensitive scheduling and execution algorithm where the overall problem (i.e. biological datasets that are used as input parameters) is divided into tasks that are assigned on demand (during run time) to processors. This model with a Task Server is also common in projects like BOINC [2] and helps improve the overall performance of embarrassingly parallel bioinformatics code [22].

V. Discussion

In the two use cases we have seen projects with rather different scopes, goals and approaches to adopt new technology. EGEE, on the one hand, provides Grid middleware as well as a production Grid infrastructure that needs to be “permanently” operational. Additionally, the code base of EGEE is based on more than 5 years of Grid development and deployment in predecessor projects. Therefore, the middleware does not adopt the latest standards in Grid computing but rather follows a more conservative approach and relies on established standards that have proved production quality. The recent standardisation changes underlined once more that the conservative approach was suitable for a production infrastructure. On the other hand, EMBRACE is an example project that does not *yet* have direct customers, and therefore can closely watch technology and adapt/use it as appropriate. This is an observation that can be made in current research&development projects as well as production oriented projects.

Up to now, I only talked about scientific use cases but Grid computing is becoming more and more important in the business and industry world [19]. This is particularly visible at international Grid events, and it is also reflected in different Grid strategies mainly presented by IT or IT related companies. However, the Grid vision about resource sharing has some negative aspects that are sometimes difficult or impossible to overcome. Sharing resources means to some extend that knowledge or power is given away. For instance, running a proprietary application on a remote resource that is not owned by the organisation or company that actually launched the application might cause conflicts in several ways.

A typical example is the pharmaceutical industry: even if there is currently interest in Grid technology for drug docking and drug discovery (i.e. rather typical embarrassingly parallel Grid applications in bioinformatics), once the drug is in a very competitive situation where it could enter the market, companies tend to prefer running the computing intensive applications in-house rather than on a "public Grid". This is due to the fact that remote sites are not considered to establish the same trust relationship as local, in-house clusters might give. Therefore, there is still quite a long way to go to give companies the full confidentiality that data and resources can be used in a very secure way.

VI. Conclusion

The article gave a brief introduction and some highlights of using Grids in High Energy Physics as well as in a selected field of bioinformatics. Sometimes researchers from outside the Grid domain question the main research goals in Grid computing since superficially several of the current problems "have been solved already somewhere else". However, the real challenge in Grid computing is the scale: having 5 sites working together is still quite simple. However, it is a major challenge to make operational more than 150 sites in several different continents with all kinds of different cultures and administrative policies.

Another challenge is the adoption in a scientific domain. High Energy Physics has already adopted the ideas of Grids and distributed computing many years ago but the technology needs to be stable and reliable. The bioinformatics community is also approaching this path. Now it is important to continue build high-level tools for application scientists to use Grid and Web Services technology in a transparent way without exposing too much the details of distributed software. Furthermore, people are not requesting anymore simple computing or data intensive applications. There is much more demand for applications that understand the *semantics* of a particular science domain. If these steps are reached, Grid technologies can also be adapted by a wider, non-scientific user community.

Acknowledgement

Thanks to Kurt Stockinger (Berkeley Lab) and Gert Vriend (University of Nijmegen) for useful comments on the paper.

This work was funded in part by the EU project EMBRACE Grid which is funded by the European Commission within its FP6 Programme, under the thematic area "Life sciences, genomics and biotechnology for health", contract number LUNG-CT-2004-512092.

References

[1] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, D.J. Lipman, D.J. Basic Local Alignment Search Tool." *J. Mol. Biol.*, 215:403-410, 1990.
[2] D. Anderson, E. Korpela, R. Walton. High-Performance Task Distribution for Volunteer Computing. *IEEE Int. Conf. on e-Science and Grid Technologies*, Australia, Dec. 5-8, 2005.
[3] J. Andreeva, et al. Distributed Computing Grid Experiences in CMS, *IEEE Transactions on Nuclear Science*, 52(4):884-890, Aug. 2005.
[4] CMS Collaboration. CMS Computing: Technical Design Report, CERN-LHCC-2005-023, CMS TDR 7, 20 June 2005. http://doc.cern.ch//archive/electronic/cern/preprints/lhcc/public/lhcc-2005-023.pdf
[5] EGEE: Enabling Grids for E-Science, http://www.eu-egee.org, Oct. 2006.
[6] M. Ellert et al. The NorduGrid Project: Using Globus Toolkit for Building Grid Infrastructure. *Nucl. Instr. and Methods* A 502:407-410, 2003.
[7] EMBnet, European Molecular Biology Network, http://www.embnet.org, Oct. 2006.
[8] EMBRACE Grid, http://www.embracegrid.info, Oct. 2006.
[9] I. Foster et al. The Open Grid Services Architecture, Version 1.0, Grid Forum Document, GFD-I.030, 29 January 2005.
[10] I. Foster, C. Kesselman. The Grid, Blueprint for a New Computing Infrastructure. Morgan Kaufman, 1998.
[11] Gartner's Hype Cycle. http://www.gartner.com/it/products/hc/hc.jsp, Oct. 2006.
[12] gLite: http://www.gLite.org, Oct 2006.
[13] A. Krogh, M. Brown, IS. Mian, K. Sjolander, D. Haussler. Hidden Markov models in computational biology. Applications to protein modeling. *J Mol Biol.*, 235(5):1501-31, 1994.
[14] LCG project and middleware, http://cern.ch/lcg, Oct. 2006.
[15] M. Litzkow, M. Livny, M. Mutka. Condor - A Hunter of Idle Workstations, *8th International Conference of Distributed Computing Systems*, San Jose, California, June 13-17, 1988.
[16] Nordugrid, http://www.nordugrid.org/, Oct. 2006.
[17] Open Grid Forum (OGF), http://www.ogf.org, Oct 2006.
[18] Open Science Grid (OSG), http://www.opensciencegrid.org, Oct. 2006.
[19] E. Schikuta, F. Donno, H. Stockinger, E. Vinek, H. Wanek, T. Weishäupl, C. Witzany. Business In the Grid: Project Results, *1st Austrian Grid Symposium*, OCG Verlag, Hagenberg, Austria, Dec. 1-2, 2005.
[20] Scientific Linux. https://www.scientificlinux.org/, Oct. 2006.
[21] H. Stockinger et al. Defining the Grid. A Snapshot on the Current View. To be published in the *Journal of Supercomputing*, 2007.
[22] H. Stockinger, M. Pagni, L. Cerutti, L. Falquet. Grid Approach to Embarrassingly Parallel CPU-Intensive Bioinformatics Problems. *2nd IEEE International Conference on e-Science and Grid Computing (e-Science 2006)*, Amsterdam, The Netherlands, Dec. 4-6, 2006.
[23] Virtual Data Toolkit (VDT), http://vdt.cs.wisc.edu/, Oct. 2006.
[24] Web Services Resource Framework (WSRF), http://www.oasis-open.org/specs/index.php#wsrfv1.2, Apr. 2006.

Biography

Heinz Stockinger's main research interests are in Grid and distributed computing. He has been working in several Grid projects in Europe (European Organization for Nuclear Research, CERN) and in the USA (Stanford Linear Accelerator Center, SLAC) for many years and in various scientific, technical and management functions. Within the European DataGrid project (the predecessor of EGEE) he was the Education and Outreach Manager as well as responsible for replication software in the Data Management work-package.

Heinz is currently affiliated with the Swiss Institute of Bioinformatics (Lausanne, Switzerland) where he works for the EMBRACE Grid project. He has been appointed associate professor ("Privatdozent") at the University of Vienna (Faculty of Computer Science), where he was leading the Research Lab for Computational Technologies and Applications in 2005. Currently, he is also a lecturer at the Swiss Federal Institute of Technology (EPFL, Lausanne). Heinz holds a Ph.D. degree in Computer Science and Business Administration from the University of Vienna, Austria.