Project no. FP6-045520

EchoGRID

European and **Ch**inese C**o**operation on **Grid**

SSA Project

Information Society Technologies

D.2.2 – **Draft Roadmap**

Due date of deliverable: 31 December 2007

Actual submission date: 20 February 2008

Start date of project: 1 January 2007 **Duration**: 24 months

Organisation name of lead contractor for this deliverable: ATOS

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	PU

Keyword List: roadmap, vision
Responsible Partner: Atos Origin

MODIFICATION CONTROL			
Version	Date	Status	Modifications made by
0	DD-MM-YYYY	Template	Florence Pesce
1	29-01-2008	Draft	Florence Pesce
2	10-02-2008	Internal review	Michael Wilson
2.2	12-02-2008	Draft	Stephanie Parker
3	12-02-2008	Final	Ignacio Soler

Deliverable manager

- Ignacio Soler - Atos Origin

List of Contributors

ERCIM	ICCS/NTUA	ATOS	ENG	THALES	BUAA	ICT	CNIC	NUDT	HUW

List of Evaluators

- Michael Wilson, ERCIM/STFC
- Domenico Laforenza, ERCIM/ISTI-CNR

Summary

- A roadmap to achieve interoperable Grid systems between Europe and China is proposed for the 3, 5, and 10 year stages addressing issues which have been identified as of mutual concern to researchers in both geographical areas: new programming paradigms, Grid architectures, Grid management, virtual organisations, the component model, workflow – business progress. Recommendations are made for EU and Chinese funding and research planners.

Table of Contents

1 Introduction

The state-of-the-art background to the current interaction of European and Chinese Grid activities was presented in the first EchoGRID deliverable D1.1 in August 2007, describing EU and Chinese Research Initiatives/Grid strategic orientations. The orientation for the project presented there was to further interaction between researchers in Europe and China through several mechanisms:

- develop gateways
- develop common components
- test interoperability
- develop common standards,
- use common security and certification infrastructure supported by the IGTF
- establish common controlled vocabularies for policies and semantically rich descriptions
- agree common accounting mechanisms
- establish an international governance institution

The endpoint of this orientation is an interoperable European & Chinese grid provision for academic/research and commercial use, built on the generic OGSA architecture, with semantically rich descriptions of services, workflows, registries, quality of service, policies and accounting to move towards the SOKU vision of the EU Next Generation Grid (NGG) Expert Group by 2018.

The current document refines the roadmap towards this vision in two ways:

- By dividing the grid technological issues into six specific areas
- By defining the expected progress along the roadmap at three points, in three, five and 10 years.

The six technical aspects of grid computing considered in detail were identified at the first joint European/Chinese technical workshop of the EchoGRID project in Beijing, China held in February 2007. These technical issues were ones where both the Chinese and European experts believed that joint technical development was required to ensure the interoperability of the resulting grid services:

- New programming paradigms
- Grid architectures
- Grid management
- Virtual organisations
- The component model
- Workflow – business progress

In this report, each of these topics is addressed as a chapter, covering a detailed analysis of their expected state at each of the three points on the roadmap towards the vision of a globally interoperable grid infrastructure.

2 NEW PROGRAMMING PARADIGMS

2.1 Vision

Over the last ten years, grid computing has seen tremendous developments of research to build effective grid infrastructure and to a larger extent service infrastructures. Despite all these efforts to design effective and usable computing infrastructures, there is still a concern about how we can program such infrastructures with a higher level of abstraction than actual programming models do. To some extent, programming grids nowadays is very similar with what computer programmers faced before programming languages came to light. Taking into consideration the increasing complexity of existing Grid or service infrastructures, it is of prime importance to study novel programming models that can both hide the complexity of the underlying computing infrastructure and provide a certain level of abstraction to enhance programmers' productivity. This has been already pointed out by Ian Foster and Karl Kesselman in their second edition of their book "The Grid 2: Blueprint for a New Computing Infrastructure":

Grid environments will require a rethinking of existing programming models and, most likely, new thinking about novel models more suitable

Programming the grid means that a suitable way to express the coordination of computations among a set of distributed resources has to be provided. As the service-oriented paradigm has been adopted by the Grid research community, programming the grid can be seen as specifying the orchestration of services. For a given problem or applications, specification of the orchestration should be done using high-level abstractions having the following properties: intuitive so that non-expert programmers can use them, generic to handle a large spectrum of applications and parallelism should be implicit and fully hidden to the programmers. Moreover, to cope with the large scale and unreliable dimension of the grid, programming languages, providing these high-level abstractions, have to be associated with a distributed execution model to avoid any bottleneck.

2.2 State of the art & Limitations

Current grid programming paradigms came from existing programming practices. Some grid programming paradigms extend CORBA for the grid context, while some others adopt traditional master-slave models. Most programming models are based on Java, using RMI, sockets as the basic communication infrastructure, they provide features like multithreading, group communication, global synchronization, and so on. For most grids aiming at providing computation capability, MPI is still used as programming language. From the perspective of an MPI programmer, the grid is a single computer, and it is a specific computer according to the individual application. Although it is possible to provide high performance programs, MPI exposes too many low level architectural details to the application developers, so that programmers must deal with resources allocation, load balancing, fault tolerance, and so on. This makes computational grids quite limited for use by the traditional parallel application developers.

As to the interaction between programming components, flow-based language and approaches are used to indicate the communication and control flow at run-time. Grid workflow is used in many e-Science environments. This paradigm enables the orchestration of Grid services via GUI or XML-based standards that are used for the description of service infrastructures.

And also there are many rising distributed component and objects models, like map-reduce, they all give new possibilities for the ecosystem of current grid programming paradigms.

There are some limitations for current grid programming paradigms and practices. In current Grid programming models, physical resources are too much exposed to the programmers, in terms of number of resources, resources characteristics. It is difficult to deal with dynamic changing large scale Grid environment using the static information of Grid resources. Programmers must handle really low-level technical details to make their applications highly efficient, adaptable to the dynamic and complex context of Grids. For the feature of level-level abstraction, these paradigms are difficult to learn, they are restricted into these well-trained Grid programmers. Also for this reason, the reusability of current models is low. Existing Grid programming models lack of scalability in term of dynamicity. They lack autonomic adaptation to the changing environments. Deterministic execution is insufficient to support the development of complex systems.

2.3 New Ideas

In the area of programming, there are many unconventional approaches being currently investigated in the context of Grid: such as Chemical Programming, Autonomic Element Programming and Pipe-based Programming. We review briefly these approaches in the next paragraphs.

Chemical programming gets its inspiration from the chemical metaphor, formally represented here by a chemical language, such as HOCL that stands for Higher-Order Chemical Language. A chemical program can be seen as a (symbolic) chemical solution where data is represented by floating molecules and computation by chemical reactions between them. When some molecules match and fulfil a reaction condition, they are replaced by the body of the reaction. That process goes on until an inert solution is reached: the solution is said to be inert when no reaction can occur anymore. Formally, a chemical solution is represented by a multi-set and reaction rules specify multi-set rewritings. In HOCL, computation is seen as reactions between molecules in a chemical solution. HOCL is higher-order: reaction rules are molecules that can be manipulated like any other molecules, i.e., HOCL programs can manipulate other HOCL programs. Reactions only occur locally between few molecules that are chosen non-deterministically. The execution can be seen as chaotic (non-deterministic) and possesses nice autonomic properties. It is thus a good candidate to express the orchestration of computation or services within a grid. Moreover, it provides a high-level of abstraction since the chemical rules are very closed to the problem specification.

Most of the resources sharing activities are distributed, parallel, heterogeneous, dynamic, and most important, managed by different parties autonomously. For any grid programmer, he/she has to write the codes which manipulate the resources owned by autonomic entities. Being the fundamental building block of autonomic systems, autonomic element is a natural programming concept for modeling and encapsulating the diversity relating to the local resource sharing activities. Autonomic element programming is very similar to the agent-oriented programming paradigm in the heart of the programming model, but it targets more on providing dynamic modelling capability to support data resource aggregation tasks with a property of best-effort. These kinds of tasks are becoming pervasive in the field of information fusion and knowledge engineering, including search engine, knowledge management, and many other Web 2.0 applications. With the evolvements of grid applications and its marriage with the de facto Web 2.0 applications, we believe that the requirement of autonomic modelling of heterogeneous resources which may support dynamic service orchestration more naturally will become more and more pervasive. Modelling grid resources from an autonomous element's perspective is likely to become one of the most widely used programming paradigms.

GSML is proposed by ICT, CAS to simplify the developers' efforts of building Grid applications via the techniques of resource virtualization and abstraction, loosely coupled component architecture and high-level reusability.

Funnel is GSML component model. Funnel means pipe, it is a software module that conceals low-level details of resources and exposes to the developers only event-based interfaces: input events and output events. Funnels are made upon state-of-art middleware, toolkit and even resource APIs, and can be used to interact with Grid middleware, Web Service, WWW resources, API resources and legacy applications as well. In a Grid resource space, run-time funnels represent a set of available virtualized and abstracted resources usable by applications, which called *funnel library*. A *funnel* may be referred by a URI globally in a given resource space, and represents a *virtual resource* which may be served by more than one real *physical resource* together. The mappings between virtual resources and physical resources, also called *resource binding*, are performed automatically and transparently at runtime for load-balance and fault-tolerance.

Funnels in resource space are connected to form Grid applications according to users' application-level requirements. A grid application is specified as an XML document which defines which funnels and how they are connected. Funnels are defined and referenced by URI which implies the usage of a certain virtual resource. The connections between funnels are specified by how the output events from source funnels are disseminated and transformed to the input events of target funnels. The loosely coupled component architecture in GSML is event-based by nature: applications register the events that interest them, trigger developers' handlers when feed by the output events, and possibly perform some transformations and send out input events to funnels. The connections and transformation between funnels are made transparent to funnels: funnels neither know what destinations to send output events nor from what sources to receive input events. Therefore funnels may be developed and deployment independently, but are reused many times on demands of applications. When application-level requirement are altered, the changes can be adapted by the flexibility of event-based connections of funnels: no need to edit, compile and link the application when changing the high-level logic of grid applications in XML document. The new paradigm also encourages building a large grid application by connecting in event-based fashion some small parts, which are themselves funnels and built by smaller parts so on. The divide-and-conquer solution of complexity in Grid applications is enabled by the high-level reusability of GSML.

Apart from these aforementioned programming paradigms, aspect-oriented programming, skeleton programming, meta-programming or high-order languages could be explored in the context of grid programming.

2.4 How Grid programming is addressed in existing EC and Chinese R&D programmes

It is widely accepted that the programming paradigm is one of the most important facts for a new concept to be widely used and adopted. Meanwhile, it is also one of the most challenging and risky topics in software engineering, because it has so many facets, technical but also non technical, and requires efforts from both academy and industry.

In China, most of the research work programs including Natural Science Foundation of China (NSFC), National High-tech R&D Program (863), National Basic Research Program of China (973), have supports on the research activities of the new programming paradigm for grid computing. Most leading projects, such as Crown grid middleware (supported by NSFC and 863, internet-based Virtual Computing Environment (supported by 973) have sub-project on this topic.

During the last decades, there were also many funding projects from the EC on the new/unconventional programming paradigms especially by the Future Emergent Technology unit and its Global Computing Initiative. Research efforts were presented at the workshop 'Unconventional Programming Paradigm 2004', supported by the European Commission Information Society Technologies Programme, Future and Emerging Technologies Activity. We can also mention the CoreGRID network of excellence that has some research activities aiming at using chemical programming for workflow enactment.

2.5 Recommendations for EC and China

New programming paradigms are seen as long-term research on both sides (Europe and China). The main recommendation is to put these research activities in the mainstream of R&D projects. More experiments with real applications and mature implementation of unconventional programming paradigms should be encouraged. The two research communities: the one addressing unconventional programming and the grid community should participate to some joint projects to put these novel approaches into practice. Although we think that these new programming paradigms have a good potential to address the existing limitations, they have to be experimented with some industrial test cases.

3 GRID ARCHITECTURES

3.1 Vision

It is commonly accepted that the short-term architectural vision for Grids must include more successful paradigms from the industry such as Service-Oriented and Web-Service Architectures (SOA & WSA). In the short-run this will greatly reduce the interoperability issues between the various Grid designs and implementations assisting in the collaboration and convergence of the Chinese and European infrastructures.

These infrastructures will manage to support the virtualization of data and networks, removing complexity from the application layers of the Grids. Virtualization can lead to the support of flexible business models and application domains, thus increasing compatibility between existing and -perhaps- independently developed middleware.

However, it is expected that in the long-run grids will eventually move away from the traditional Web1.0’s fixed Provider/Consumer architecture. Moreover, extending the Web 2.0 principles in the grid area, we can model various resources as equal entities acting as not only providers but also consumers, whose actions are driven by on-demand aggregation and autonomic collaboration based on their own interests. This revolutionary approach will be driven by an effort to develop a Grid that will be seamless to the application end-user, hiding the complexities of the underlying mechanisms. The architectures in the long-run will manage to support a wider audience of end-users, providing them the tools to develop, provide and consume services concluding to an actual democratization of grids and through that, resolving numerous technological issues.

3.2 State of the art

In order to reach the short and long term visions, the two communities can build on some progress that has already taken place in some selected works that are already leading the way to the realization of the visions, especially the short-term.

As it is widely accepted that the short-term vision has already started to be applied, the main work taking place in China and Europe, is a mix of Grid and SOA infrastructures. The developed frameworks comprise the state of the art in the architecture area in Grids. This is witnessed in a number of projects for the development of Grid frameworks such as XServices, CNGRID, CROWN and UNICORE, GRIA and NextGRID ISTs.

The basic Grid architectures for China consist of CNGRID, CROWN and XServices. China has funded a series of Grid projects to establish Grid infrastructure in recent ten years. CNGrid and CROWN are two well-known projects for building Grid testbeds, which integrate high-performance computers with a new generation of information infrastructure and facilitate scientific research in different disciplines. The Grid architectures inside testbeds are both based on Service-Oriented Architecture and OGSA, while compliant to many Web Services standards of W3C, OASIS and WS-I. XServices seems to be an interesting case, as it is largely based on a web service computing principles. XServices is a software suite and an effective way to implement SOA, which consists of Web Services Runtime & Application Server, Web Services Workflow Designer & Engine, UDDI Server, Web Services-based Information Portal and several useful tools to facilitate service development, deployment & management, and to provide QoS & security support. XServices, only based on JDK, is open source and fully implements WSRF & WSN to support resource management and state notification.

Respectfully, the main representatives of Grid architectures for Europe in the context of this roadmap, seem to be GRIA, UNICORE and NextGRID. The UNICORE 6.0 is a WSRF based implementation of the UNICORE Grid middleware. It forms a software stack that implements an extensible service-oriented architecture compliant to current Web services standards. Similarly, GRIA is a service-oriented infrastructure designed to support B2B collaborations through service provision across organizational boundaries in a secure, interoperable and flexible manner. Again, it is based on a number of Web services standards. Finally, NextGRID is working towards the direction that OGSA has indicated, extending the standards and formulizing a prototype Grid architecture, targeting mainly in the economic viability of Grids.

Of course, important variations of the abovementioned architectures have appeared that worth being used for building towards the vision. Such are the SIMDAT, TrustCOM, BREIN and XtreemeOS ISTs. SIMDAT uses the GRIA framework and extends it with dynamic properties deriving from NextGRID in order to support a number of business applications. TrustCOM, focuses on frameworks where trust plays a central role. BREIN is dealing with the business perspective of the relationships between the actors in a SOA environment. Finally, XtreemeOS takes a different route by attempting a vertical virtualization of Grid layers. All these, along with two more interesting ISTs (Grid4ALL and SOA4ALL) can significantly contribute in the democratization of Grids, which seems to be the long-term vision.

3.3 New ideas & Limitations

In the way towards realizing the vision starting from the current state of the art there are some obstacles. These obstacles form the limitations that are likely to encumber further progress of the Grids. However, at the same time, there are some ideas the implementation of which will comprise a breakthrough in the area, bringing Grids one step closer to their vision while at the same time, contributing to the convergence of the two research efforts.

3.3.1 New Ideas

One idea that will greatly help in resolving the compatibility between Grid infrastructures is the development of architectures based on principles (e.g. standards) that will focus on the re-use of existing infrastructures. Moving to more service-oriented frameworks will render specific grid middleware obsolete, however specific techniques that have been applied there, are still useful. Moreover, as the “Grid Frameworks” will grow bigger to become large scale, distributed environments, they will very much resemble Enterprise Service Buses (ESBs). Each “service” on such an infrastructure can be a smaller framework that will have specific technical properties and may even satisfy different application requirements. Such a sub-framework could even be the Grid infrastructures themselves.

Another innovative idea that would benefit the Grid vision would be to design architectures that will support dynamic trust and security and SLAs. The components of these architectures must be deployed and configured in such a way that will enable trust and security assessment of the relationships in real time. The relationships must be fully described by SLAs that must also be dynamically assessed. Assessment will be achieved by profiling customers and providers, using historic information on their requirements and capabilities (e.g. QoS).

Another area that will help in the Chinese and European effort to materialize the vision is to make specific breakthroughs in the resource management mechanisms. Especially, it has been identified that designing and including mechanisms that will realize improved coordinating of the resource reservation. The idea is that the democratization of Grids will soon require a specific pool of resources to be used by a greater amount of applications. This means that resource reservation will become crucial in satisfying both the functional and non-functional requirements of the applications. It is expected that future resource reservation planning is a topic that, once properly addressed, will cover the needs of the vision.

By combining the European efforts around GRIA and the Chinese efforts around Xservices it should be possible to develop an open source framework for distributed SOA deployments, which should be able to compete with US developments like Websphere and WebLogic, which currently dominate this market.

3.3.2 Limitations

Standards have entered the world of Grid architectures in order to resolve interoperability issues, not only between the -by default- heterogeneous resources but also between the various designs. However, in practice, things have turned out differently as using a standard in the design-phase and implementing it in the development phase has often led to incompatibilities. Moreover, it is not uncommon to see two different standardization efforts to work separately towards the same goal (e.g. WS-Eventing and WS-BaseNotification). The result is a set of standards that on one hand specify and profile the operation of at least a part of a Grid architecture but on the other create conflicts as to whether a specific version is the appropriate one or not. This mainly yields from the different experiences as well as of different goals that each development team might have, leading architects to a varying assessment of the maturity and the need of standards.

Another issue that seems to be hindering the process of grids and of the convergence in the initiatives of the two communities is the different use of grids in general. It seems that Grids in China are designed with a strong focus on the particular application that it is intended to serve, usually coming from the public sector. Grids do not necessarily have a business-oriented perspective, at least to the extent that European research and development communities would like to have. Thus, there is a difference in the business models support, at least at a conceptual level.

Still, there is a perennial problem to all the architectures. This is the different understanding of an architecture, a problem which especially in Grids is much more evident. The multilayer nature of Grids (from a resource layer to an application layer through the middleware layer and so on so forth) usually causes problems to architects that tend to design based on a different model. E.g. one may consider that an architecture is needed in order to define the profiles of the interactions between conceptual models whereas one other may suggest to a specific component model, explaining in detail what classes are needed. In summary, the lack of formulization of grid architectures presents obstacles in realizing the vision.

Finally, country specific political policies which govern the nature of sharing resources, data, applications, user access to information, etc will play a major role in shaping the advances for this topic.

3.4 How it is covered in existing EC and Chinese Work Programme

It is commonly accepted that the both the Chinese and European workprogrammes for Grids have already paved the way for realizing what has been set as the short-term vision for Grid architectures.

In FP7, research in Grid Technologies is part of the ICT Challenge "Software & Services Architectures and Infrastructures". Its purpose is the continuity of the FP6 Grid Technologies projects and activities as well as the closer integration with current trends and initiatives towards service-oriented architectures and infrastructures.

Similarly in China, The National High-tech R&D Programme (863), sponsored by the MOST, is the major source of government funding on grid research in China. The biggest effort is the China National Grid (CNGrid). The first phase of CNGrid was supported by the project "High performance Computer and Core Software" from 2002 to 2005, with a government funding of 100 Million RMB. The second phase of CNGrid was launched in Sept. 2006 with a government funding of 640 million RMB from the new project "High Productivity Computers and Grid Service Environment" from 2006 to 2010, and one of the project's targets is to support SOA inside Grid environments, too.

3.5 Recommendations for EC and China

There are a number of steps that it would be advisable for the two research communities to follow in order to take the state of the art closer to the vision. These steps are:

- Put effort in dynamism (such as dynamic service selection, composition, business models support, etc)
- Emphasize the development of WS standards for grids that will also support backwards compatibility
- Develop open source competitors for tools that support distribute SOA infrastructures (e.g. WebSphere)

Starting from the visionary goal to move to SOA frameworks rendering obsolete monolithic grid systems, it is advised that effort must be set to support dynamism in various fields: service selection, service composition, SLA composition, business models, etc. Broadening the capability of Grid infrastructures to support more applications and end-users, more systems and services, greatly increases the dynamicity of the environment. Along to that must come management systems to enable the administration of these environments while encompassing policies and several important criteria (e.g. business-oriented criteria) for any selection made.

Moreover, the two research communities should invest more in developing web services' standards for promoting the grid concept. These standards must enhance the capability of the developed infrastructures to integrate with Grid systems that are now outdated. Moreover, it is important to find a way to design architectures that will be able to put together frameworks that perhaps are created to support a shortened range of applications but in a very successful way.

4 GRID MANAGEMENT

4.1 Vision

The term Grid Management is in itself quite self explanatory and also in the same sense as grid induces ambiguity to scope of this topic. The approach to define a vision is to start identifying the fundamental aspects associated with management and couple it with motivational scenarios for the usage of the Grid and its relevance to the stakeholders.

The grid in its basic essence has emphasized on “Widely Distributed Secure Sharing” aspect as applied to the shared entity usually referred to as resource. This driving force should be supported based upon the following:

a) Local provision considering security and privacy issues.
b) On-demand access to content and adaptation of the content to context.
c) An ecosystem of autonomic, self managed systems.
d) The contexts could be users, devices, location and policies including and not limited to state, organization and user.
e) Capabilities of handling very large data volumes with long term digital curation assessed by business. Eg: 2 to 10 PBs per year per business.
f) Implementing dynamic business models with clear accountability of consumption and pricing issues across industry, enterprise and research community.
g) Economy of scale considering factors like cost, GreenIT, energy, cooling – modelling renewable resources.

With the above motivations, we envision scoping the management of grid through a “Utility” model scaling across geographic zones of enterprises and the heterogeneous devices and appliances leading to a merged vision – Pervasive Utility model. This utility model will be applicable to the widely distributed secure sharing of resources including but not limited to computing, storage and bandwidth.

The management is best understood by analyzing based on the fundamental aspects as mentioned below and popularly termed as “FCAPS” management:

a) Fault Management
b) Configuration Management
c) Accounting or Charging management
d) Performance management
e) Security management: Authentication, Authorization, etc.

We elaborate a roadmap for Grid management for 3, 5 and 10 years by rationalizing the above fundamental aspects with a unison vision of grid layers as below:

a) Resource
b) Connectivity
c) Collective
d) Applications – supported by the Grid infrastructure
e) Users – To be analyzed from stakeholder perspective.

4.2 State of the art & Limitations

The current status of Grid management is still evolving in EU and China with both addressing specifics in resource management.

4.2.1 EU engineering initiatives in Grid management:

a) UK National Grid Service (NGS): STFC operates the UK National Grid Service (NGS) for researchers to use national and international computing and data resources. The NGS has about 400 real users of whom about 100 use about 90% of the resources. The STFC e-Science department of 140 people operate the UK NGS as well as other national resources for computing and data management including the 6 Petabyte Atlas data store, the national scientific visualisation service and compute servers for several research communities.

b) Efforts on resource management, scheduling and load balancing. Development of an Execution Management System (EMS) where QoS plays a significant role in the distribution of tasks. Efforts on SLA management and accounting for supporting systems that provide strong QoS guarantees.

4.2.2 China engineering initiatives in Grid management:

a) iVCE: Internet-based Virtual Computing Environment, is a 5-years project funded by National Basic Research Program of China (aka. 973 program).The first focus is Resource Aggregation Model, Mechanism and Computational Properties (iVCE-2), the other is called Resource Collaboration Model, Mechanism and Computational Properties (iVCE-3).

b) CNGrid: China National Grid is engaged in grid resource monitoring and management in CNGrid Environment.

c) SDG: Scientific Data Grid has focused on SDGFinder to search for registered services in a registry

d) Bird Flu Monitor Grid: This focuses on security of data resource sensors and mobility management issues with bird migration.

e) CNGrid GOS: Naming management in GOS is a resilience decentralized registry for variety kinds of global object. It provides low latency object locating by object GUID, high success rate searching by multiple attributes match, stable object view based on linked naming services to enable the effective-virtual-physical address space. User management, resource management, agora management functions are based on the naming. The static metainfo in HPCG is also stored in naming.

4.2.3 SDO initiatives in Grid management:

OGF has dedicated an area called Management Area which has the following working groups active and proposed many recommendation and information documents:

a) Usage Records (UR) – Working Group: It has produced the Usage Record Format recommendation (GFD.98)

b) Configuration Description, Deployment and Lifecycle Management (CDDLM) – Working Group: It has produced the Smart Frog – Configuration Description Language, a CDDLM component Model, CDDLM API for deployment description. It defines a coherent CDDLM framework too.

c) Grid Economic Services Architecture (GESA) – Working Group: It mainly concerns the economies/trading in Grid. It has output a Grid Economy Use cases document.
d) Application Contents Service (ACS) – Working Group: This group mainly concerns to manage application contents.

The state of the art and limitations in Grid management are presented in the below descriptive table:

	Fault	**Configuration**	**Accounting**	**Performance**	**Security**	**Dynamic capabilities**
Resources	Tools for individual resources	Tools for individual resources	Tools for individual resources	Tools for individual resources	Tools for individual resources	Tools for individual resources
Connectivity	ITU-T TMN, MSeries	Mseries	MSeries	MSeries	MSeries	Proprietary implementations possibly based on the CMIP stack or SNMP.
Collective	(gLite); Globus GOS GRIA (Bridge) Monitoring & notification	gLite (resource discovery) GOS (UDDI-like registry), iVCE	EGEE (resources not money; compute not storage) GOS (no accounting)	Some monitoring	trust certificate PMA	GOS (monitoring)
Applications (supported by Grid) solutions)	Some jobs reallocate resources on failure. No check-pointing framework	Stand-alone with much local knowledge required; limited hot service deployment	Not even defined	Isolated applications only; no performance metrics	No application-level security	No autonomic management. Heterogeneous application profiling/management requirements.
User	User currently manages faults	Aligns application requirements to resources	nothing	Tunes job submission	Certificates with proxies (manages own credentials)	No portable user context

Table 1: State of the Art and Limitations in Grid Management

To summarize, the management issues that a Grid Service Provider will face from the current date to 2018 are as follows:
a) Regulation around Grids with emphasis on Certification, Accounting rules, establishing trust domains.
b) Technologically, standardization of interfaces and protocols.
c) Management of Fault, Configuration, Accounting, Performance, Security and Dynamic capabilities dependent on resource, environment or the object.

4.3 New ideas

The new ideas proposed are primarily aligned as 3 milestones to be achieved over the next 10 years. This aims to bring about a consistent thought process of what is achievable and what really requires long term research within the time frames. The new ideas will evolve over time; the format provided below suggests a rationale to align the focus as innovations are achieved through research.

a) 3 years roadmap: Standardization and prototype
b) 5 years roadmap: Declarative semantics based management
c) 10 years roadmap: Autonomic Grid management

4.3.1 Three years roadmap:

Emphasis is on Standardization and prototyping of the vision of Grid management to make it consistent across EU and China. It demands extensive use of existing technologies and development of consistent frameworks across vendors. The below table illustrates the vision:

2011 - STANDARDIZATION & PROTOTYPE	**Fault**	**Configuration**	**Accounting**	**Performance**	**Security**	**Dynamic capabilities**
Resources	Cross-vendor standardization					
Connectivity	Better wireless (heterogeneous) support within the current policy framework					
Collective	Improved accuracy in fault location; fault correlation	Standardized definition & procedures	Defined common cost model for (bandwidth), data storage, & processing	Defined monitoring performance, management of failure & standardized SLA.	Interoperability of diverse security management systems	Extending existing management information models like CIM to cope with heterogeneity & dynamics of Grid
Applications (supported by Grid) solutions)	Defining check-pointing framework for basic recovery.	Generic description of applications (prototypes)	Definition of application cost model & prototype optimisation tools.	Definition of application metrics.	Rationalize application-level security	Specify supporting environment for Virtual Organizations.
User	Fault-analysis tools	Configuration management tools	Define framework for grid economics	Performance analysis tools	Standardization of user access to grids & associated tools	Specify framework for analysis of FCAPS analysis and management

Table 2: Vision 2011 - Standardization and Prototype

4.3.2 Five years roadmap:

Emphasis is on Declarative semantics based management for grids. This will meet the primary aspect of grid – Interoperability in dynamic environments. Rules based management of Grid systems is suggested to be the focus. The below table illustrates the vision:

<table>
<tr><th>2013- GRID MANAGEMENT BASED ON DECLARATIVE SEMANTICS</th><th>Fault</th><th>Configuration</th><th>Accounting</th><th>Performance</th><th>Security</th><th>Dynamic capabilities</th></tr>
<tr><td>Resources</td><td colspan="5">Standard specifications for FCAPS management</td><td rowspan="5">Interoperability of management across the areas of management considering heterogeneous site Grid.</td></tr>
<tr><td>Connectivity</td><td colspan="5">Research on declarative semantics (rules) based management of Underlying networks.</td></tr>
<tr><td>Collective</td><td>Automated resource & connectivity re-allocation</td><td>Automatic configuration tools & tools-assisted configuration & composition.</td><td>Declarative semantics for accounting policy specifications</td><td>Declarative semantics for Performance management specifications</td><td>Declarative semantics for security management specifications</td></tr>
<tr><td>Applications (supported by Grid) solutions)</td><td>Standards for check-pointing & fault reporting. Demonstration of migration of processes across nodes after faults</td><td>Standardisation of application descriptions & mapping to configuration. Workflow libraries.</td><td>Development of optimisation tools covering costs semantics</td><td>Development of application level semantics for statistical analysis</td><td>-</td></tr>
<tr><td>User</td><td>Helper tools for fault recovery, management.</td><td>Tools and Semantics for capturing inputs for self configuration and reconfiguration</td><td>Declare requirements for Grid usage accounting & decision-making</td><td>Helper tools for Performance management</td><td>Rationalize requirement for universal identity management for Grid.</td></tr>
</table>

Table 3: Vision 2013 - Declarative semantics based management

4.3.3 Ten years roadmap:

Envisioning the usage scenarios of Grid in 10 years, the proposal is to focus on Autonomic Grid management across federated systems with reasoning capabilities. The below table illustrates the vision:

2018 - AUTONOMIC GRID MANAGEMENT	Fault	Configuration	Accounting	Performance	Security
Resources	A framework for self management for reasoning and executing run-time policies governing FCAPS management				
Connectivity	Self Healing Networks	Context Aware self configuring/re-configuring networks	Dynamic and adaptive Grid economics implementation like 1+1+1=2 cost units. (Sum of parts is not equal to the whole)	Self tuning/ optimizing networks	Self aware networks. Develops Dynamic shield across different types of network attacks.
Collective	Self Healing enablers and management	Self-configuration & composition. Autonomic framework with semantic-based configuration.	Requirement & resource-aware accounting system (context aware)	Self tuning / optimization enablers and management	Self awareness enablers and management
Applications (supported by Grid) solutions)	Automated management of fault	Application workflow/semantics to consider configuration semantics	Adaptive accounting system to optimize resource usage based on cost	-	-
User	Physically replace resources.	Policy specification that would enable auto-configuration or self configuration/ re-configuration	Decision delegation for accounting needs.	No action required	Advanced security interfaces including biometrics.

Table 4: Vision 2018 - Autonomic Grid Management

4.4 How is it covered in existing EC and Chinese WP

Grid management has been the non-priority topic in the research world. The main focus is usually on the functional aspect of the grid. The functional aspect has been evolving and aimed at interoperability in the existing grid infrastructures. The management aspect of the grid had been only seen as a tool for easing the workload upon the stakeholders for assisting analysis of resource metrics. Both EU and China might not be thoroughly aware of the need for establishing strategic partnerships in Grid Management.

4.4.1 EU Work Programme Initiatives Status

The EC funded Framework Programme (FP) projects have been instrumental in the development of Grid research and have attained considerable maturity in all aspects of Grid. The FP6 projects like NESSI-Grid, BEinGRID, Akogrimo, XtreemOS and many more have been focusing on various grid aspects for specific resources and are evolving specific management functions. There is now a need to bring in all the common functions and tabulate them more formally and understand the implications for the standard interactions for management across resources. The EC funded FP7 initiatives should consider explicit focus on Grid Management.

4.4.2 China Work Programme Initiatives Status

The National High-tech R&D Programme (863), sponsored by the Ministry of Science and Technology (MOST) of China, and National Natural Science Foundation of China (NSFC) are the two major sources of government funding. The National Basic Research Program of China has also involved in funding through the 973 projects. The above initiatives have made considerable efforts for general Grid initiatives in China. The projects under these programs like iVCE, CNGrid have taken considerable initiatives in pursuing challenges for Grid Management as illustrated in the State of the art.

4.4.3 SDO Initiatives

OGF (Open Grid Forum) has been a pioneer in Grid standards initiatives and has dedicated efforts in evolving the Grid management aspects through its MANAGEMENT area working group. The WG has proposed some recommendations for Grid management as illustrated in the State of the Art. There are many other SDOs like OASIS also specifically focusing on developing web service standards for management. In view of industry standards for connectivity management, the telecommunication management networks framework has been standardized through ITU-T and TMF. These industry standards for management of telecommunication resources can have a major impact on the evolution of Grid management standards. These SDO initiatives have significant involvement from EU and China.

4.5 Recommendations for EC and China

To provide further inputs and realize the above mentioned visions, the following recommendations are proposed:

a) Grid management should be an explicit focus of research in both EU and Chinese research initiatives like FP7 and 863 programmes and focus should be on SMEs and industry to experiment with innovative business models upon them.
b) Grid management should take considerable input from the Telco standard fraternity like ITU-T, TMF for Grid to enter into a clear managed scope of business enabler. Today we have considerable working grid efforts and the motivation to keep these efforts functional will require the accountability of these Grid infrastructures in a global business scenario. Without Grid management practices in place, the funding for these initiatives will lack clear motivation.
c) Additionally, ETSI could take the role for Grids in Europe, the Ministry for Information in China that ITU does for Telcos, as Regulator and Standardization body.

5 VIRTUAL ORGANISATION

5.1 Vision

A virtual organization (VO) is a dynamic group of individuals, groups, or organizations who define the conditions and rules for sharing resources. The concept of the VO is the key to Grid computing. It is a buzzword in recent years. Different projects have different definition for their Virtual Organization. But all VOs share some characteristics and issues, including common concerns and requirements that may vary in size, scope, duration, sociology and structure.

From our survey among some partners of EchoGRID, a VO should have following functionalities, such as user/role management, resource management, trust and security management; and a VO should have the properties including: dynamic, QoS guarantee, scalable; also for a VO that wishes to exist stably for longer time, it should have a business model.

The rising of SOA (service-oriented architecture) has impacted the grid technologies. Since SOA encompasses architecture issues in both business and information systems, it has great potential to enable alignment of VO approaches with IT-based business networking approaches, such as enterprise interoperability. So while SOA can contribute to VOs, VOs can also make their own demands from the new architecture.

5.2 State of the art & Limitations

VO is a key part of a grid platform; almost every project on Grid will do research on VO, or use the mature techniques of VO. We list projects from EchoGRID partners that deal with VO here.

One of the efforts visible today is XtreemOS. They are concerned with building and promoting Linux based operating system to support virtual organizations for next generation grids. It is compliant with OGF SAGA (Simple APIs for Grid Applications) specification. Installed on each participating machine (personal computer, cluster of workstations, mobile devices), the XtreemOS system will provide for the Grid what a traditional Operating System offers for a single computer: abstraction from the hardware and secure resource sharing between different users. It will thus considerably ease the work of users belonging to virtual organizations giving them the illusion of using a traditional computer, and releasing them from dealing with the complex resource management issues of a typical Grid environment. By integrating Grid capabilities into the Linux kernel, XtreemOS will also provide a more robust, secure and easier-to-manage infrastructure for system administrators. This will be experimentally demonstrated with a set of real applications, provided by well-known industrial partners that cover a large spectrum of application fields.

Started in December 2005, project iVCE, Internet-based Virtual Computing Environment, is a 5-years project funded by National Basic Research Program of China (aka. 973 program). There are 7 sub-projects in this project. One of the sub-projects is called Resource Aggregation Model, Mechanism and Computational Properties (iVCE-2). One of the research topics of this sub-project is to figure out basic mechanism which supports the process of establishing virtual community/organization. Virtual commonwealth is a basic mechanism for establishing and managing VOs within a task context. It specifies abstract services to publish, discover and organize grid resources for aggregation, and describes the closure of autonomic elements and environments needed for current task. The service interfaces are realized as a set of common protocols and common rules between autonomic elements and virtual commonwealth. The dynamic binding of autonomic elements and virtual commonwealth is achieved by join/adapt semantics.

In the project CNGrid which is funded by China National High-Tech 863 programme and China Science Grid supported by Natural Science Foundation of China, VO is implemented to maintain the user, resource, security information of a Grid community, and it is named as Agora. Agora provides address mapping between physical and virtual resources and a cross domain access control facility combined with DAC and MAC. Agora keeps records of runtime sessions; it provides resource abstraction to applications through a runtime construct: Grip.

Although many VO projects exist, there are still limitations that slowdown the step towards a practical VO.

For the functionalities, current VO lacks of:

- Supporting from the underlying OS and Grid middleware
- Supporting of commercial applications
- Underlying administration facilities for VO bootstrapping

Current VO lacks some good properties for better resource sharing and better user usages:

- Consuming too much resources, heavy-weighted;
- QoS guarantees
- Dynamic mechanisms
- Scalability

Furthermore, most of the Grid resources are bound together by government funding, seldom industries participate into the VO operation, there is no stable and long-term model to maintain the VO. Current VO lacks:

- Business and economic models for long-term running and operation.

5.3 New ideas

In order to minimize the gap towards the vision, there are several new ideas proposed from academia to achieve the goals mentioned above. We list them below.

Adding advanced functionalities into VO to better support for resource sharing.

- Better VO support for coordinated job execution, meta scheduling, job migration. Job execution in grid environment will focus on a) state-based sessions or services or jobs, b) deployment times lasting more than several minutes or hours, and c) restricted mobility. Jobs (services or sessions) would be lasting from several minutes to a few of days. Coupled with dynamic allocation or de-allocation of resources, job monitor should be able to execute migration of jobs. Meanwhile, cooperation between jobs restricts the mobility of jobs, which should be guaranteed by job monitor to prevent jobs becoming uncontrolled. Check-pointing is also adopted to make sure secure execution of jobs, especially for parallel applications. It seems parallel jobs are more vulnerable in grid environment since members (user and resources) in grid are permitted to modify in runtime.

- Adding people as an interaction entity of VO. Grid-enabled workspace for people collaboration: Desktop tools incorporating all kinds of medias, IM, emails, software … etc.
- Semantic support in VO. Semantic description of service and information is an enrichment of Service-Oriented architectures. This significant mechanism will bring great benefits to the scalability and efficiency of resource discovery, service adaptation and service composition. Knowledge-based techniques should be revisited in the dynamic and asymmetric grid context, how they will be used based on incomplete and partial-correct information, how to minimize the complexity while still get benefit from it?

We need a dynamic, easy and full life-cycle management infrastructure of VO.

- Basic infrastructure to create, deploy the VO. Current VOs lack of self-features for the management of itself. Self-configuration, self-deployment, self-adaptation are important properties for the autonomic administration of VO. This provides great help to the operation of VO.
- Dynamic policy-driven user, resource and trust relationship management. Current VOs make accounting and trust policies according to the identification, this kind of information will not change in the dynamic and large scale Grid context. User access permissions to resources are usually defined by the local system administration or the grid administration centre, that won't be changed in a relative long time period. Dynamic information like user behaviours is not taken into consideration. As the VOs may evolve over time, to our point of view, this should have great impact on trust relationship definition. For example, a user who often exceeds his resource quota or often incur the crash of the hosting environment should be limited for his access to Grid resources. Just like the social trust system, a person's rights should be defined according to his reputation.
- More efficient and finer-grained QoS control mechanisms. Quality of a VO may include performance, reliability, SLA. So that we think a VO should provide following mechanisms to facilitate fine-grained quality control:
  - Fine-grain resource and job monitoring. The information of resource performance/cost, availability, job submission, job queuing may be needed for quality control.
  - User history profiles and resource usage tracks. The information may give important hints for Grid workload description, usage pattern and resource assumption pattern.
  - High efficient information retrieval utilities. Past and current system status is critical to decide future actions, it is a fundamental underlying service for Grid platform. Many modules will get information from it. So it is important to provide a high performance information retrieval service, and also some semantic info may be used to locate a term with higher correct.
  - Job scheduling strategies taking into consideration load balance, user reputation, resource reliability and so on.
  - Failure detection and fault tolerance framework. Failures in VO should be detected in time, and provide applications with flexible and practical recovery and fault tolerant mechanisms, such as reboot, process migration, multi-version processes, etc.

Enabling VO across different platforms and infrastructure

- Interoperability between different VOs. Grid applications and services may access resources belonged to multiple management domains and hosting environments. That may incur policy level interoperability problems. Different Grid services and resources may require various certificates and access control policies, thus, the VO should record application session information and transfer roles between different sessions.
- Single system image of different VO information, resources, users, and so on. Different VOs may have different users and resources registration services, making them seeing and understanding each other is a problem of interoperability. It may require some standards to describe the information of resources, users profile, policies, and some credential data that are involved during an interoperation process.

5.4 How it is covered in existing EC and Chinese WP

Many projects from EC and China are involved with VO.
In China, the National High-tech R&D Programme (863) and the National Basic Research Program of China (973 program), sponsored by the Ministry of Science and Technology (MOST) of China, and the National Natural Science Foundation of China (NSFC), provide government funding for VO development and operation. The virtual organization (Agora) in CNGrid is sponsored from the key research project "High performance computers and software" in 863. The guideline of 863 project proposal mentions VO research as follows: "Provide general support for various applications and resources, such as virtualization and service mechanisms, naming, resource and service access, dynamic service management, batch and interactive job scheduling, Grid user management, Grid resource information service, Grid security."

In EU, we have a partial list of projects on VO in FP6 or before, Akogrimo, Daidalos, EGEE, HPC4U, NextGRID, EU-Provenance, SIMDAT, TrustCoM, UniGridS, InteliGrid,etc. In the FP7 work programme 2007-08 includes VO related topics, ICT-2007.1.2, ICT-2007.1.4, focusing on "service and software architectures, infrastructures and engineering", and "Secure, dependable and trusted infrastructure".

5.5 Recommendations for EC and China

As making a roadmap for future 3, 5 and 10 years, we suggest that priorities be given to VO research themes presented above.

For the first 3 year period, a clear definition of VO framework on the functionalities and management infrastructures should be proposed. This may involve the discussion from Europe and China academy, some standards and interfaces may be proposed during this period. It is important to involve related industries to the discussion, they will provide requirement from real end-users, and help with setting up the business model.

For the middle-term aim, advanced functionalities of VO except for semantic support, cross platform VO and high efficient VO should be provided to the users.
For the 10 year goal, semantic support in VO should be realized, and relevant standards should be raised. Also, autonomic capabilities for full life-cycle management of VO should be provided.

6 COMPONENT MODEL

6.1 Vision

It has been widely accepted that the component-based software development is an effective means to tackle the complexity of modern software, such as software to support Grid applications. With the evolution of internetworking and pervasive computation, it can be anticipated that in the near future, it is likely that we will witness large scale interactions between components developed by different parties across the Internet. It is an improbability that component developers will be able to develop such kind of internet-scale applications with today's component models and tools, which are mostly stemmed from the enterprise computing environments of the near past.

If we could efficiently handle the predicted complexity derived from the large-scale interactions between future Grid applications, we could conclude to the following visionary ideas that may be proved useful as a guide in the technological advance of this area:

1. Produce component model to support autonomic software development so as to improve adaptation and reusability at a high level.
2. Develop a unified component model, that will fit the needs of future industrial use cases as well as those of today's Grid applications.
3. Clearer separation of concerns regarding the functional and non-functional requirements to ease the maintenance and the flexibility.
4. Standardize software certification models (such as the testing model for Grid software) at the component model level in order to improve software quality assurance.
5. The construction of reliable and scalable software systems require better behavioural guarantees. Elements towards this goal include: more expressive specification formalisms, development environment providing "correct by construction" code, dynamic adaptation techniques, etc.
6. Ensure QoS at the component level which will allow component applications to provide support SLA requirements.

6.2 State of the Art & existing limitations

Reusability has been improved with component models originating from industry (e.g. EJB, CCM, etc.) as well as from academia (e.g. Fractal, GCM, SCA, etc.).

Furthermore some work has been done regarding the integration of various component models and tools. In detail, effort has been focused in integrating PKUAS and Jonas (Chinese component model), while ProActive/GCM has been used in Plugtests between European and Chinese Grid platforms. With respect to SCA, some path has been attempted to integrate component ideas into Service Oriented Architectures. However, such a platform does not yet feature the power of full-fledged components with, for instance, the capacity to impose -when needed- the interoperable and flexible deployment of components upon execution.

Besides the improvement above, it is clear that academia and industry are still lacking mechanisms to support adaptation and reconfiguration of components during execution, based on the dynamic event occurring in real-time; for instance machine failure or overload. Moreover, we are still missing the mechanisms to take into account application level requirements, such as Service Level Agreements, in order to improve contracted Quality of Services.

Some of the component models are too complex to be widely adopted by the industry and domain users. Part of the problem is the lack of tools to assemble component configuration, to deploy components onto various platforms, and to monitor them at execution.

The difficulties to identify the basic requirements for a given component in order to dynamically select the right set of machines on which they can be executed is yet another problem that prevent components from being used in a SOA. Such functionality will require a unified model to describe application needs, which shall be valid for most of the Grid application. In turn, this will require a standardization of applications description in order for them to really be promoted at the level of reusable components.

Finally, as in practice we are dealing with various component models, in various languages, there would be a need for interoperability in component management at execution (life cycle management, reconfiguration, migration from one machine to another). A challenge is to achieve a full differentiation of functional and non-functional part of the codes, in order to ease the maintenance and the flexibility. Overall, we believe we have to achieve a full integration of components into Service Oriented Architecture, with the capacity to monitor and dynamically maintain the Quality of Services at execution.

6.3 New ideas

New light-weight component model for the autonomic support in Grid applications
It is widely accepted that a few component models tend to be over-weighted for developing autonomic software entities. But in some of the applications, not all the properties are needed as well. If the component model itself can be configurable or reconfigurable according to the development contexts, developers can have much more freedom in choosing those real useful functionalities.

New light-weight component model is needed to give more support on the modelling of autonomy of software. This component model can be derived from existing models such as CCM but must be light-weighted to preserve only the functionalities which are of necessity.

The main research topic in this area is on how to implement the semantics of autonomic support in a component-based way. Or, put it in another way, what are the necessities a component model should have in order to fulfil the basic functionalities an autonomic software entity. We argue that this is important and feasible because (1) there are many component models available but they are targeted for different domains; and (2) the component model should be kept light-weight enough and at the same time sufficient to support the dynamic nature of Grid application.

Component model supporting service composition

Extend component model, like GCM, to provide a programming framework fitted for both scientific and business applications. Also, this will have to bridge the gap between component technologies and Service Oriented architecture (SOA) in order to achieve code and service composition for flexibility, agility, etc while keeping the benefits of the component programming model.

Standardization of platform independent models to let the different component models to "talk" to each other

Component-based paradigm has been one of the dominant programming paradigms for its modularity, reuse, and runtime flexibility. Quite a lot of research initiatives and projects have been launched on this topic and has achieved many advantages including the GCM, the Fractal, and the GridCCM component models, to name a few. In order to make them "talk" to each other in a scalable way, platform independent models are necessary. There were/are a lot of standardization institutions and works relating to component-based software engineering in an enterprise computing context, such as the OMG's PIM/PSM models in MDA. The practice of future Grid component standardization may benefit from them.

Aspect-oriented programming combined with component model

Focusing on various aspects of Grid environments, there are system characteristics such as user interfaces, component persistency and distribution, collaborative work and end user configuration support that can be found everywhere. Component developers tend to use aspects to describe component requirements, and use aspects to guide component implementation as well. With the development of aspect-oriented programming methods, it can be anticipated that AOP will contribute to tackle the coupling of functional and non-functional codes within components. Previous research works relating to component-based systems engineering could be a very good start towards this research direction.

Quality Assurance for Component Model design

What about the component model? The component model seems to be a very good way to develop Grid software. What is the connection between it and the quality assurance of the final product?

CM design has a direct impact on the development phase. So the final product can get many advantages by good CM design that may have aspects that ensure that the software produced is high quality software and is characterized by some unique features that the architect demands (e.g. security).

So the CM design is an important step towards the need of high quality Grid Software. CM can act as a Quality Assurance mechanism, but this is true only if a standard shared and accepted model is reached and it is well applied. A good idea could be to propose/develop something similar to GRID-QCM that is oriented to test the result of the design phase by automatable metrics. This can be done only having a really strong and accepted standardisation of the CM so as to find really generally applicable metrics.

Following this path we can reach the goal to make CM a certified way in the production of good Grid software.

The next 3 years could be a good time to standardise a common way to project CM designed software. The next 5 years should be enough to define a quality certification model for CM design. This could be added as integration of existing models (e.g. GridQCM) or a standalone model. Finally, it is expected that in the next 10 years it could be made possible to model a standard for the Grid community.

5.4 How it is covered in existing EC & China WP

A number of projects are conducting research on component models under the frame of the:
- Chinese 863 programme (NUDT and PKU).

- CoreGRID FP6
In CoreGRID, the institute on programming models aims to deliver a definition of a lightweight, component programming model that can be usefully exploited to design, implement and run grid applications.

- GridCOMP FP6
GridCOMP main goal is the design and implementation of a component based framework suitable to support the development of efficient grid applications. It provides the reference implementation of the GCM defined in CoreGRID.

- FP 7 Challenge 1 - CNR-ISTI

- FP 6 ETICS (March 2008, version 2, FP7)
The ETICS (now ETICS2) project is going to provide the users a free platform to automatically build and test their grid and non-grid software. This platform is complemented by GridQCM, a proposal for a quality certification model that can be automated by the implementation of its metric in an automatic build and test tool.

5.5 Recommendations for EC and China

Based on the above analysis, we would like to present the following recommendations.

(1) Clarify the relationships among component-based and service-based models, environments and tools, in particular concerning the possibility to migrate the innovative component features of Grid component model (such as GCM) to WS/SOA platform (such as X-Service)

(2) Both the EC and China need to put together research groups that will tackle these important issues.
(a) New component models capable of supporting new properties of future Grid software, such as autonomic execution, dynamic roles, adaptive behaviours, QoS etc;
(b) Collaborations should be funded to speed-up the development of programming framework fitted for both scientific and business applications;
(c) Interoperability of different component models, at different layers.

7 WORKFLOW – BUSINESS PROGRESS

7.1 Vision

It is well-known that Grid workflow/business process is a critical part of the Grid environments, which captures the linkage of constituent services together in a hierarchical fashion to build larger composite services. It can enable collaborations between large-scale resources to support QoS-aware, secure guaranteed, optimized complex scientific applications and various business applications in the short-term vision.

For the moment, there are over twenty kinds of grid workflow systems, and each is based on specific workflow description language and execution enviroment. Within five years, it is urgent to decouple the workflow systems with their specific Grid environments, and various legacy workflows can be incorporated despite the discriminations, for reducing costs as well as enabling dynamic collaboration and boundary-crossing problem solving.

Hopefully, in the long term, the business process decision-makers can use simulations seamlessly without problems supporting the high-level decision analysis. And this can be achieved by modelling the performance of detailed problem features by workflows and the integration of these workflows in a hierarchy of abstraction levels step by step, providing advanced workflow composition, execution and debugging tools support this integration at each abstraction level, and also matching the real-time events driven by the workflow with the simulation level parameters.

7.2 State of the art & Limitations

Some of research and engineering work has already taken place in related projects of both EU and China to achieve above visions. Typical research work and engineering practice involved Grid workflow/business process, funded by Ministry of Science & Technology (MOST), National Natural Science Foundation (NSFC) in China and vendors, are CNGrid GOS Grid Workflow, VINCA, CROWN and XServices Workflow, Protein Analysis Scientific Workflow, InforFlow etc.

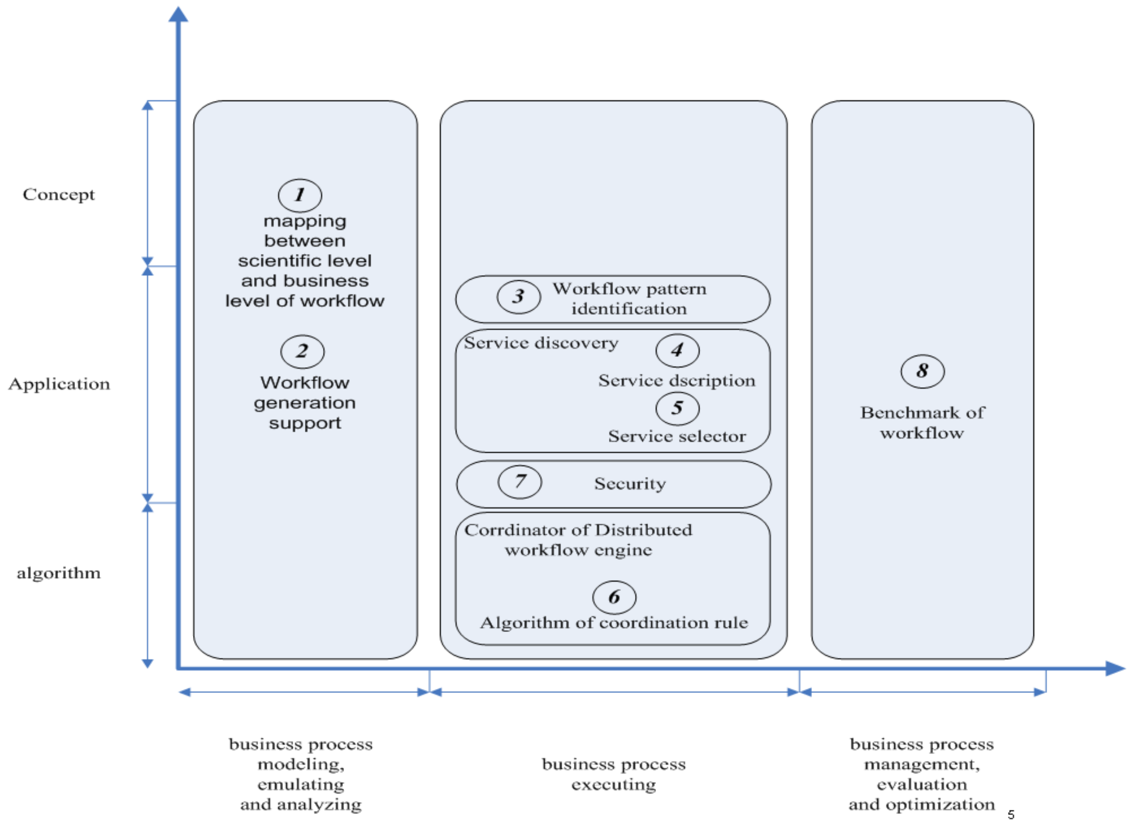
At present, several workflow systems exist in the CNGrid, CROWN and XServices, such as BPEL4WS engines, JSDL processing engines, and others. From the perspective of interoperability, these legacy workflow systems should be candidates to be included into a grid as a specific kind of resources providing orchestration capabilities, for lowering costs as well as enabling dynamic collaboration and boundary-crossing problem solving. GOS Grid Workflow extends previous work VINCA to a meta-workflow system, so that other workflow capabilities in several business and scientific communities can be incorporated. To this end, a concept model is proposed, in which the capabilities of legacy workflow applications and the enactment engines are virtualized as abstract services that can be further used to construct applications from business level. Protein Analysis Scientific Workflow project's main purpose is to deal with the distributed, dynamic grid workflow collaboration problem. Not only can it provide business workflow function, but also deal with the data produce in the scientific workflow. InforFlow provides the Human workflow development, integrated workflow development, the development tools, management tools and operating environment support for the customers. Meanwhile, it combines with Web2.0 technology to package different service of the workflow engine into different widgets, and this function will support customers to add new workflow function in their system.

Correspondingly, the main representatives of workflow/business process evolvement in Europe are SIMDAT, BRIDGE, PoSSIS, ASKALON, etc. In SIMDAT and BRIDGE research on Workflows in the context of grid is a major topic. As part of SIMDAT and BRIDGE or in close relation with the work of SIMDAT, the following workflow tools have been enhanced and partially grid enabled: TAVERNA (IT Innovation), OPTIMUS (Noesis), Modelcenter, isight and KDE (Inforsense). Interoperability in the grid contexts between these workflow tools have been demonstrated by several distributed workflows involving several tools in particular for applications in the development of airplanes. Initial work was performed on an advisor tools for workflow generation. The same cases are also in the PoSSIS, ASKALON and other projects.

Nevertheless, although current research and engineering progress have achieved important outcomes and many workflow/business process engines and tools occur now, there are still some remarkable limitations to be mentioned. First, the gap between the business process decision-maker and the application developer is obvious and wide. Secondly, it seems to be very difficult to precisely compare and evaluate the performance of different workflow engines; moreover, it is even hardly to improve the performance of execution of designed workflow engines. Thirdly, QoS-based resource scheduling, service selection and service composition are less considered, and the co-ordination of different distributed workflow engines needs to be considered and designed in more efficiency. Finally, security is still a vital element here, which needs to be well provided by all the workflow/business process engines, but few existed engines achieve this maturely now.

7.3 New ideas

From the state of art and limitations summarized above, it is seemed that there will be a long way and difficult to achieve the vision. Luckily, at the same time, as research and engineering practices on workflow/business process become in-depth, new ideas for solving these limitations and challenges occur gradually. In the figure below, it will orderly explain these new ideas from two directions. One direction is from the algorithm, application and concept levels, while the other is through the fields of business process modelling/emulating/analyzing, executing and management/evaluating/ optimization. Each idea is corresponding to a point of intersection (x, y) from two directions.



① **Mapping between scientific and business levels of workflows**: The gap between the business process decision-maker and the application developer is wide. While business decision maker usually use BPEL for the description of workflows, each engineering or scientific discipline has its own workflow language and established tools. BPEL itself lacks certain structural elements like loops, which are essential for optimization workflows, for example, to implement the vision for improving the decision making processes by including simulation related workflows into the decision process the gap between several abstraction levels have to be closed by support for seamless inclusion of low level workflows into high-level decision support workflows.

② **Workflow generation support**: The generation of workflows is a time consuming tasks, which need experiences about the workflow tool as well as the application domain. On the other hand, companies will develop and archive a huge number of workflows for different purposes. By enhancing the workflows with appropriate semantic information it will be possible to simplify the generation of workflows and make usage of the workflow repository. For a new problem the improved workflow environment will allow to find a set of workflows, which have been used already in a similar context. For prototypes of workflows, it will be possible to find similar ones, which might be a source for modifications or even alternatives. In a second step, the semantic information for workflows can be generated automatically from the workflow itself as well as from the actual usage of the workflows.

③ **Precisely identifying the pattern of the workflow**: As the different pattern of workflow (concurrent or sequential task), the performance of workflow engine is different. If the pattern of workflow is orderly, the engine execute the workflow is better. If the pattern of workflow is concurrent, engine compiles the BPEL to the middle language for high concurrent pipelines is better than sequential run. So identifying the pattern of business process will develop within pattern recognition evolution in the future.

④ **Improving the performance of semantic tagging of service description**: Service description language will contain the QoS and ontology information of the service. For the reason of performance of semantic tagging, before the add the semantic tag to the WSDL, create an inverted-index to search a similar WSDL set and tag them with semantic information, the methods which mixes the similarity search and semantic matching process.

⑤ **QoS and resource scheduling**: Selecting suitable service with global constraints. Context information (including context of different engines) is focused on the services composition, respecting the global constraints, service selector choose the suitable solution reducing the re-optimization overhead to compose an optimal process from candidate service sets.

⑥ **Coordinating the distributed workflows**: The coordinator cooperate the different engines according and follow the coordinate rules. Fuzzy logic of expert system affects the reasoning process, and now, Rete II algorithm of Forward-Chaining is applied in rules decision because considering the expected efficiency, the methods of Backward-Chaining in induction will create a precise pattern matcher in next years.

⑦ **Workflow Security Support**: Due to the security requirements in workflow and business process, it should provide special security services with different security level for such kind of applications. Moreover, a fine-grained and extensible security framework, trust federation and negotiation for resource sharing and collaboration are also needed to apply now and in the near future.

⑧ **Evaluating the workflow engine**: The Benchmark of workflow engine evaluates the performance of workflow engine. In the business process, the environment is often change in every time (such as the workflow of Grid, no a killer benchmark for evaluating the resource of VO (Visual Organization)), so the fair property of benchmark must be considered, the consumer must select the suitable workflow engine under the fair judgment methods.

7.4 How it is covered in existing EC and Chinese WP

In general, both China and EU are aware of strategic essentiality to the research of workflow and business process. Up to now, they have already set down some Work Programmes and funded several projects for achieving short-term and middle-term vision.
The National High-tech R&D Programme (863), sponsored by the Ministry of Science and Technology (MOST) of China and National Natural Science Foundation of China (NSFC) are the two major sources of government funding on Workflow and business process research in China. Many efforts have been put into research projects. As mentioned above, CNGrid GOS Grid workflow, VINCA and VINCA4Science, etc., are all illustrations for this issue. For the current 5-year programme (2006-2010) of 863, there are a number of projects for supporting to reach the short-term and middle-term vision.

European Union, the same as China, according to the Research priorities highlights of NESSI during 2009-2010, Research area of business process modelling is included. To meet the demands of paving the way towards the collaborative executable enterprises, the researchers will focus most of their attention on how to support dynamic formation, formalization management and interaction of business processes implemented through services, and how to provide long-term transactional business collaboration. In the Service and software architectures, infrastructures and engineering Theme, ICT-1-1.2 of FP7, Project NEXOF - Reference Architecture (FP7-216446)'s WP2 - Service-centric systems engineering, is to define and describe architectural components for creating, describing, managing services and composing them into workflows in order to meet specified business requirements. This is an exemplification to show that, fulfilling the projects (e.g. TRUSTCOM, GRIA, NESSI-Grid, BEinGRID, etc.) of FP6, EC will continue to supporting the research of workflow and business process in FP7.

7.5 Recommendations for EC and China

In order to come closer to the short-term (three years), middle-term (five years) and long-term (ten years) vision mentioned above, it is recommended that the research and engineering communities from both EC and China pay more attention to the following important issues:

First, to make the composition of services an explicit research topic in the revised EU Work Programme is a way to match its emphasis on the Chinese Research Programme, for example, to close the gap by mapping between the scientific level and the business level of workflows, support workflow generation by mining workflow repositories for similar workflows and workflows solving similar tasks, etc.

Secondly, advanced workflow technology is central to bring grid technology to decision maker's workbench. Moreover, research programme should not only care for grid architecture and typical grid applications themselves, but also pay more and more attention to the workflow/business process technology.

Finally, research should not focus on the development of new workflow tools, but concentrate on far-reaching advances, such as integration of these workflows from different workflow tools in a hierarchy of abstraction levels, designing advanced workflow composition, execution and debugging tools which support this integration at each and between abstraction levels, matching between the real-time events driven by the workflow and the simulation level parameters.

8 Conclusion

The recommendations to the European and Chinese research funders, and research planners for each of the six topics are stated at the end of the each of the preceding six chapters. A very brief summary of these recommendations is:

- Programming paradigms research activities should be put in the main stream of R&D projects instead of being left peripheral.
- Put effort in dynamism (such as dynamic service selection, composition, business models support, etc).
- Emphasize on the development of WS standards for Grids that will also support backwards compatibility.
- Develop open source competitors for tools that support distribute SOA infrastructures (e.g. WebSphere).
- Grid Management should be an explicit focus of research in both EU and Chinese research initiatives and focus should be on SMEs and industry to experiment with innovative business models upon them.
- Grid Management should take considerable input from the Telco standard fraternity. ETSI could take the role for Grids in Europe, the Ministry for Information in China that ITU does for Telcos, as Regulator and Standardization body.
- For the first 3 year period, a clear definition of VO framework on the functionalities and management infrastructures should be proposed.
- For the middle-term aim, advanced functionalities of VO except for semantic support, cross platform VO and high efficient VO should be provided to the users.
- For the 10 year goal, semantic support in VO should be realized, and relevant standards should be raised. Also, autonomic capabilities for full life-cycle management of VO should be provided.
- Clarify the relationships among component-based and service-based models, environments and tools, in particular concerning the possibility to migrate the innovative component features of Grid component model (such as GCM) to WS/SOA platform (such as X-Service).
- Both the EC and China need to put together research groups that will tackle these important issues.
    - ① New component models capable of supporting new properties of future Grid software, such as autonomic execution, dynamic roles, adaptive behaviours, QoS etc;
    - ② Collaborations should be funded to speedup the development of programming framework fitted for both scientific and business applications;
    - ③ Interoperability of different component models, at different layers.

- To make the composition of services an explicit research topic, for example, to close the gap by mapping between the scientific level and the business level of workflows, support workflow generation by mining workflow repositories for similar workflows and workflows solving similar tasks, etc.

- Research programme should pay more and more attention to the workflow/business process technology.
- Research should not focus on the development of new workflow tools, but concentrate on far reaching advances.