Design of a Workflow-Based Grid Framework

Binh Thanh Nguyen, Duc Huu Nguyen, Thuy Thanh Nguyen, and Doan Bang Hoang

Abstract—This paper aims to present the design of the Grid Collaborative Framework which has been proposed in one of our previous work. Grid infrastructure for resources sharing is somewhat stable with the wide acceptance of the Open Grid Services Architecture (OGSA) and Web Services Resource Framework (WSRF), but Grid framework for collaboration is far from desired. Current Grid Collaborative Frameworks (GCFs) are domain specific and lack of plan-supported capability. These limitations make them less useful and narrow in scope of application. Our grid collaborative framework aims to improve these limitations. With the theoretical foundation based on the activity theory, workflow languages, and designed on top of existing OGSA infrastructure, our proposed framework aims at accelerating the development of grid collaborative systems that consider work plans as central role.

Index Terms—Activity theory, grid computing, grid collaborative framework, workflow language.

I. INTRODUCTION

Over the last decade, Grid Computing has become an important field of research and development, with the objective of resolving a real and specific problem of 'coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations (VO)' [1].

As a result, the Open Grid Services Architecture was developed and widely accepted as an appropriate infrastructure for such purposes. A description of this architecture has been presented by Ian Foster *et al*. [1] (see Fig. 1).
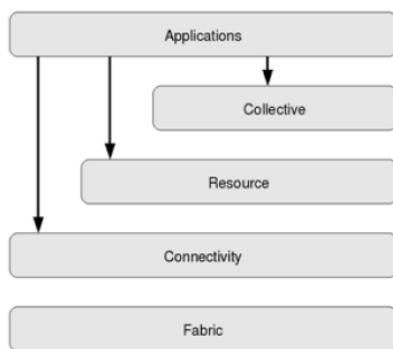

Fig. 1. The Grid architecture.

With the development of various platforms such as Globus Toolkit 4, Condor, GridWay, the functionalities and

Binh T. Nguyen and Duc H. Nguyen are with the Hanoi University of Science and Technology, Dai Co Viet St., Vietnam (e-mail: binh.nguyenthanh2@hust.edu.vn, duc.nguyenhuu@hust.edu.vn).
Thuy T. Nguyen is with the Hanoi University of Engineering and Technology (e-mail: nguyenthanhthuy@vnu.edu.vn).
Doan B. Hoang is with the School of Computing and Communications, Faculty of Engineering and Information Technology, the University of Technology, Sydney (UTS) (e-mail: dhoang@it.uts.edu.vn).

implementation for the three lowest levels (Resource, Connectivity and Fabric) of OGSA have been well established. However, those of the collective level are still far from the expectation of grid application developers. One of the necessary components in this level is a collaboration framework that supports the description, monitoring, and execution of multistep workflows. Recently, several efforts have been made in developing Grid Collaborative Frameworks such as GridCole [2], QuarkNet/Grid [3], Collaborative Design Grid [4], Grid-based Cooperative Work Framework (GCWF) [5]. However, the scope of their application is narrow because of two main problems:

- **Domain-specific**: the Grid framework is closely tied up with a specific application domain and makes it impossible to be reused as a framework for applications in other domains.
- **Lack of Plan-supported capacity**: With most of collaborative work, the very first step is planning which creates a work plan containing the sequence of the jobs that need to be performed by participants depending on their positions/roles. Then, following this plan, the work will be distributed, executed, and monitored in order to check the progress of the work execution and for further assessment. Without plan-supported capacity, it is very hard to keep track of the progress of the work, to support interactive collaboration and more importantly to modify the plan once it has been in place.

There are some reasons for these problems. Firstly, the nature of most grid applications is collaborative, distributed and resource-intensive, as Larry Smarr said, 'The Grid is about collaboration, about working together." [6] (p. 12). (See I. Foster and C. Kesselman [6] for existing typical grid applications). Secondly, it is not easy to develop a general purpose, effective grid collaborative framework for a wide range of grid applications because there are numerous modes and styles of collaboration and coordination. Thirdly and most importantly, due to the complexity of collaboration and coordination, it is crucial and yet difficult to establish a theoretical foundation that permits understanding the nature of collaboration and collaborative work, and allows feasible implementations.

To date, several theoretical foundations have been developed: Coordination Mechanisms [7], and Common Artifacts [8] and Activity Theory [9], [10]. Among them, the Activity Theory seems most appropriate for our purpose for a number of reasons. First, it takes the holistic view of human activity, especially collective activity. Second, it explains clearly the nature of collaboration and the different levels of collaboration in collective activities, and from that understanding necessary collaboration and coordination mechanisms for collective activity-collaborative work can be modeled, designed, and implemented.

Recently, we have proposed a Grid Framework for

Assistive Healthcare [11], [12]. Currently, we are designing and implementing a health monitoring system using wireless sensors, mobile Internet and Grid technologies. The objective of the system is to propose a solution to take care of expensive but non-critical health care components by deploying latest technologies in collaborative Grid infrastructure, secure and automated sensor data records and wireless sensor/actor networks.

Our infrastructure is based on three concepts which differentiate it from other existing work:
- Mobile sensing and actuating,
- Active health records, and
- Collaborative Grid.

Each concept realizes an essential component of the assistive healthcare system as shown in Fig. 2.

The overall framework of the proposal consists of three main components: a sensor/actor loop, sensor records and associated active services, and a collaborative Grid platform.

The sensor/actor loop consists of various body sensors, actuators, PDA devices, and servers. A sensing communication link is established between the sensors through the PDAs to the servers. The loop is for collecting relevant sensed data and to ensure that at least an appropriate means to deliver assistive advice regardless of the nature of the communication link.
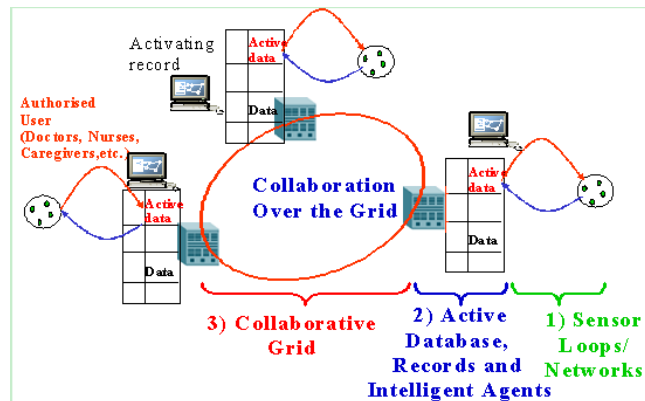

Fig. 2. Overall grid framework for assistive healthcare.

The active health sensor record component consists of relevant electronic records and various active services. The records inherit a generic record which is extendible to accommodate different types of sensors and data. Selective components of the records are linked to appropriate active services which are intelligent agents that can be invoked automatically (as well as manually) to take appropriate actions and deliver relevant responses to the user.

The collaborative Grid platform houses our middleware components and various collaborative services. The Grid platform can be connected with other Grid platforms over the Internet to form a virtual organization in which all components can collaborate and share their resources.

In current research, our focus is only on the grid collaborative component of this system. In a previous work [13], a framework for this components has been proposed. The framework has four main characteristics:
- **General Purpose**: our framework is not limited to supporting a specific grid applications domain; rather it aims at supporting most basic coordination and collaboration primitives.

- **Plan-Supported**: our framework supports planning as an integral part of collaborative work. Each collaborative work composes of two components: a work plan and a work script. This characteristic will be explained in more details in the Section III.
- **Geographically Distributed Collaboration**: our framework supports geographically distributed collaboration, allowing participating members (or groups) located in geographically separated locations to collaborate.
- **Heterogeneity of Underlying Systems**: our framework supports the use different tools and systems by participating members at different locations.

To achieve two characteristics General Purpose and Plan-Supported, our framework bases on the workflow technologies and Activity Theory. Meanwhile the two rest characteristics can be achieved by using current grid infrastructure.

This paper aims to present the design of the Grid Collaborative Framework which has been proposed in [13].

The rest of the paper is organized as follows. Section II presents necessary background including Activity Theory, Grid infrastructures and Workflow technologies. The focus of Section III is on the architecture and design of our grid collaborative framework. In Section IV, related work will be reviewed and compared with our research. Finally, Section V is for conclusions and suggestion of future work.

II. BACKGROUND

A. Activity Theory

Activity Theory has had a long history when its research had been initiated by three researchers of the cultural-historical school of Russian psychology, L. S. Vygotsky, A. N. Leont'ev and A. R. Luria [9], [14], [15], in the period from 1920s to 1930s. However, until recently, this theory has again drawn much attention in some fields of information technology such as human-computer interaction [16], Computer Supported Collaborative Work (CSCW) [10].

In the context of collaborative work, Activity Theory helps to explain more clearly the role of planning of work and the relationship between planning of work and the work itself. In some domains such as healthcare, science research, planning really plays a central role, because without planning, it is difficult to see how the objective of the work can be achieved and achieved efficiently.
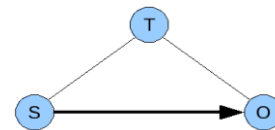

Fig. 3. Three components of an activity.

An activity has to be viewed in its entirety with its internal components and its specific dynamics. An activity is composed of a subject (S), an object (O) and mediated by a tool (T) (see Fig. 3). A subject is a person, an agent or a group engaged in an activity and who directs the activity towards an object. The mediation is through the use of tools. An object can be a plan or a common idea or something that can be

shared, manipulated, and transformed by the active participating subjects [16]. Tools can be a variety of artifacts such as electronic medical records, a computer system, a language. Each activity is distinguished from one another by their objects which manifest the objective of the activity. The object is connected to the motive of the activity and hence a subject's activity is motivated by the object (see *appendix A* for more details about this theory).

B. Grid Infrastructures

OGSA (Open Grid Services Architecture): is one of the outcomes of Open Grid Forum (OGF) that aims to solve the need for standardization of Grid systems. As said in [17]:

"Key to the realization of this Grid vision is standardization, so that the diverse components that make up a modern computing environment can be discovered, accessed, allocated, monitored, accounted for, billed for, etc" [17], (p. 4).

This architecture defines a set of core capabilities and behaviors needed by Grid systems. OGSA aligns with Web services technologies by using Web Services Description Language (WSDL) to gain self-describing, discoverable services, and adds some necessary extensions to support stateful services, lifetime management, notification, policy and credential management, and this new kind of services is called \textit{Grid service}.

Globus Toolkit 4: The open source Globus Toolkit (GT) is a one of the most widely adopted implementation of the OGSA. It allows organizations to share diverse resources (computers, storage, data, services, etc.) securely online across institutional and geographic boundaries without sacrificing local autonomy. The version GT4 [18] includes software components and libraries for resource management, data movement, resource access, resource discovery, security management, etc.

C. Workflow Languages and Systems

Workflow management aims to model and control processes in many areas such as business, science and engineering, with the objective to find out optimized processes in effective ways [19]. Using workflow languages, workflow management systems (WFMS) help model these processes by workflows. On other hand, workflows are means to represent some aspects of the processes, which help find out more easily the problems of the processes. From that, feasible solutions can be proposed and analysed. Finally, the best solution can be found: the expected optimized processes.

In our framework, a work plan is similar to a process, and a work script is similar to a workflow. That is why the workflow technologies have been chosen as one of foundations for our framework. More specifically, BPMN and BPEL are the two workflow languages used to express the work plans and work scripts.

III. ARCHITECTURE OF OUR GRID COLLABORATIVE FRAMEWORK

A. Objectives

The main objective of the architecture is to serve as a plan-supported Grid framework for a wide range of collaborative applications. The characteristic of plan supported of the framework can be explained in more details as follows.

Each collaborative work needs to have two related parts, a work plan and a work script. The work plan which corresponds to the activity level, consists of sequence of actions. Each action aims to achieve a goal among all goals of that plan/activity. The plan only takes care of what actions of work need to be done, and not of who will do those actions and how they can be done. In contrast, a work script needs to define clearly who will do what actions and how the actions can be implemented. Therefore, it composes of a sequence of operations and control structures. The operations are executable components such as programs, functions, services (Web and Grid services), etc.

In the near future, our framework aims to allow many users in concurrence to edit the work plans and scripts, as well as to run and monitor the status of the running scripts. During the process of editing a script, each user may try and select the best resources by his or her own experience, so that the script could be run most effectively.

B. Approaches

Activity Theory: There are two reasons for us to choose the theory as a theoretical foundation for our framework. Firstly, under the light of this theory, the role of work plan and its relationship with the work itself can be understood more clearly. As stated in [20]: *"plans as socially constructed and used artifacts,"* this means that on one hand, a plan is the object of a collective activity. On the other hand, when the plan has been completed, it again becomes an artifact for achieving the work. Then after having finished, the work in turn may become a plan for the next work. This understanding is crucial for the development of plan-supported collaborative framework. Secondly, the activity theory implies and suggests a comprehensive set of collaborative tools of a general-purpose collaborative framework. The application of this theory in our framework will be explained in more details in the next section.

OGSA and grid infrastructure: Recently, with the rapid development of both standardization and infrastructure, grid computing seems to become the most appropriate candidate for building geographically distributed and highly heterogeneous environments.

C. Architecture

Our framework composes of two layers (see Fig. 4):

1) Collective activity layer

This layer allows different users to collaborate in order to build work plans, edit work scripts, then run and monitor the edited work. Major components of this layer are described below (see upper layer in Fig. 4):

- **VO and Group Management**: This component is responsible for updating of Virtual Organizations (VOs), groups in the VOs, users in the groups. It also needs to manage access rights and roles of the users in VOs.
- **Activity Planning**: This component is responsible for creating a new work plan or updating existing ones.
- **Action Assigning**: This component is responsible for assigning the action(s) in the work plan to each user.
- **Selecting Resources and Artifacts**: This component allows users to find and select suitable resources used by

actions in the work plans as well as necessary collaborative artifacts for collaboration of the actions. The final result of the selection will be a work script.

- **Collaborative Artifact Store**: This store contains all collaborative artifacts.
- **Running and Monitoring**: this component is responsible for launching, running, monitoring, and terminating activities.
- **Resource Directory**: This component contains resources needed for running operations of work.

2) Resource coordination layer

The main task of this layer is to manage distributed resources and make them ready for usage of the upper layer. The existing grid infrastructure (Globus Toolkit 4) will be used for this layer (see bottom layer in Fig. 4).
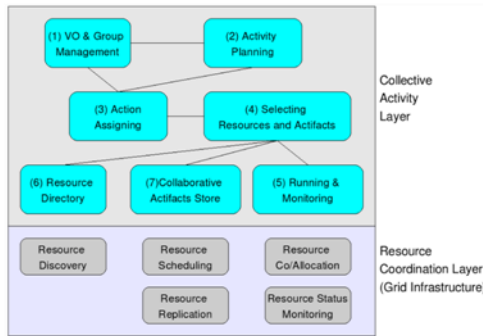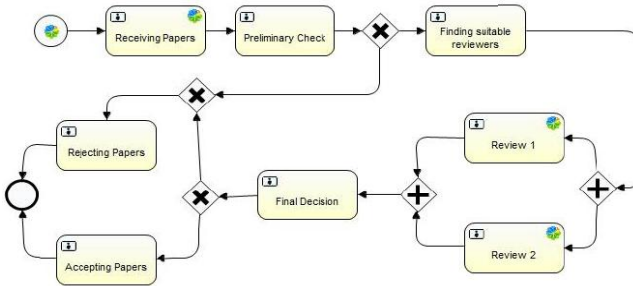

Fig. 4. Architecture of our GCF.

Fig. 5. A plan for the process of paper submission in conferences.

Let's take an example about what is a plan that our framework wants to support.

Example 1: In this example, we show a plan for the process of paper submission in scientific conferences. Fig. 5 shows the steps of the process as follows:

- Firstly, in task *Receiving Papers*, papers from all over the world will be received by a clerk in the organization commitee.
- This clerk then checks the preliminary conditions of the papers (task *Preliminary Check*) such as formats, main topics in order to eliminate the non-compliant papers.
- After that, each compliant paper needs be sent to two independent reviewers which have suitable experiences in the topic of the paper. The task *Finding suitable reviewers* aims to find such reviewers who then will accept review the paper (tasks *Review 1* and *Review 2*).
- The review results then will be sent to other person who will make final decision: whether accepting or rejecting the reviewed papers.

D. Application of the Activity Theory in Our Framework

For our framework, the application of the Activity Theory

(AT) is shown in Fig. 6. As can be seen in the Figure, a group of users has the role of Subject; Plan has the role of Tool; and the Object of this activity is a Workflow (work plan).
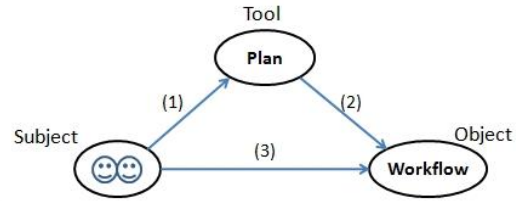

Fig. 6. Application of the AT for our framework.

Like the mediation role of Tool for Object, Plan also has a supporting role for Workflow. In reality, before construction of a workflow, its plan needs to be built and analysed to detect any potential problems soon. After the problems have been resolved, new workflows will be designed and analysed. This process may iterate several times to achieve the expected workflow.

To give an example, let's return back to *Example 1* about the plan for processing paper submission shown in Fig. 5. With the objective to achieve a complete workflow that can be executed by some resources which can be human (manual execution) or software applications (automatic execution), the above initial plan has some following important roles:

- It helps the organizers of the conferences to determine main activities and their order which should be followed.
- It also helps to establish the staff organization: how many people should be involved, and assignment of each person to the activities in the plan.
- Based on the above assignments, each person may need to develop a more detailed plan from the assigned activity.

E. Structure Design

From the above architecture of the framework, the structure of the framework has been designed. This structure consists of the following modules:

1) **VO and group management module (VGMM)**: This module is responsible for updating VOs and user groups of the VOs.
2) **Activity planning module (APM)**: This module is responsible for updating work plans that need to be expressed by some workflow language. Through analysing existing workflow languages, a workflow language called BPMN (business process modelling notation) [21] has been chosen because of the following reasons:
- Firstly, the development of BPMN aims to unify existing workflow languages in order to achieve an unified workflow language. Moreover, this language has been standardized and mature enough with version 2.0. This process-oriented language is also suitable for describing high level business processes (what to do with a business process, not how to do it), therefore it is quite suited to express work plans.
- Secondly, the translation from BPMN to BPEL(Business Process Execution Language) which has been selected in Activity Execution Module, has gained much attention in research recently with many positive outcomes [22]-[25].
3) **Activity execution module (AEM)**: The responsibility of this module is to execute the work plans from the above

APM module. However, because that work plans usually describe high level business processes which are not easy to be executed directly, the more feasible solution is to transform the plans into lower level business processes which can be executed directly. Among existing workflow languages, BPEL [26], [27] is our selection to express this low level business processes because of the following reasons:

- Firstly, BPEL is a workflow language that can be used to describe both high level business processes (also called *abstract business processes*) and low level business processes (also called *executable business processes*). Executable business processes can be executed by BPEL engines such as Active BPEL, ODE, ect.
- Secondly, BPEL, as a service-oriented workflow language, has standardized the ways of executing workflows through invocation of Web services. This feature has a significant meaning in extension of this workflow language in order to connect to other environments. One of our research objectives is to integrating BPEL into Grid environments by enabling BPEL processes to invoke and execute Grid services.

4) **Grid infrastructure**: Grid infrastructure will store and manage resource storages which will be used by the modules mentioned above. This infrastructure will help to execute and monitor plans and workflows in the APM and AEM.

TABLE I: MAPPING OF FUNCTIONALITIES OF THE FRAMEWORK INTO THE MODULES

Functionalities	Modules
VO and Group Management	VO and Group Management Modul (VGMM).
Activity Planning	Activity Planning Modul (APM): Using BPMN to describe plans.
Action Assigning	Activity Planning Modul (APM) & Activity Execution Modul (AEM): Using translation tools to transform BPMN into BPEL.
Selecting Resources and Artifacts	Activity Execution Modul (AEM): Using BPEL to describe work plans.
Collaborative Artifacts Store	Integrated in APM & AEM
Running and Monitoring	Workflow engines integrated in AEM & Grid infrastructure.
Resource Directory	Resource Directory

Table I shows the mapping of functionalities of our framework into the modules mentioned above.

The relationship between the above modules is shown in Fig. 7. As can be seen in the figure, the sequence of steps (operations) used to connect these modules as follows:

- **Step 1**: a user login into the system by VO and Group Management Module. Besides of a normal account (like a pair of username and password), each user in Grid environment needs a valid certificate which has been issued by a trusted authorization authority. After successful login, the user then can registry in Virtual Organizations to start working session.
- **Step 2**: The user uses a suitable plan editor (one of BPMN editors will be chosen) to edit and update plans about some business processes. The edition can be proceeded in group of users that will discuss and produce the best plan at that time. However, this plan can be modified to adapt new circumstances in the future.

- **Step 3**: The plan edited in step 2 will be transformed into the new workflow expressed in other form (BPEL used in this form) which will be more suitable for the execution and monitor of the workflow in the Grid environment.
- **Step 4**: Before the execution of the workflow produced in step 3, all necessary resources (hardware and software) need to be searched and then allocated.
- **Step 5**: The users will start the execution of the workflow, and then monitor the execution progress. The results and speed of the execution will be compared with the plan in order to detect as soon as possible the possible problems and to find ways to solve them. Sometimes, it is necessary to modify the plan in order to resolve these problems.
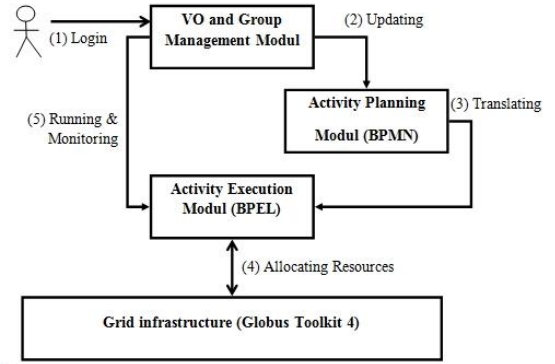

Fig. 7. The relationship between the modules.

F. Implementation

The objective of integration of BPEL into Grid environment has been achieved through our extension of an open source BPEL engine (the ODE Apache Engine [28]) (our extension called *G-ODE*) which allows invocations of Grid services from BPEL processes [29]-[31].

Using Globus Toolkit 4, a Grid networked environment as the Coordination of Resource level has been set up, and the implementation of the components in the Collective Activity level is also in progress.

IV. RELATED WORK

This section reviews existing Grid Collaborative Framework (GCF) and other kinds of related work.

A. Existing GCF

1) The PATIENT SCHEDULER [10]

It is a prototype developed in the project SAIK whose objective is to investigate how network-based computers could improve cooperation and coordination of patient treatment. This objective shares some similarities with our project.

The development of PATIENT SCHEDULER aims at illustrating how the coordination and collaboration of healthcare work can be supported by computers. However, this product is only a prototype, and is only applied into the healthcare domain.

2) GridCole [2]

It is a collaborative E-learning system that supports the realization of scripted learning situations which each of them consists of sequence of activities. In addition, with the

desirable feature of tailorability, end-users of this system (educators and students) can integrate external tools into the learning situations. By using the grid services approach, this integration enables different kinds of tools; even those require supercomputer capabilities and specific hardware. In this system, IMS Learning Design (IMS-LD) specification has been used to describe learning situations. There are two kinds of external tools in GridCole, individual and collaborative. Among two of them, collaborative tools will be used to coordinate activities within each learning situation.

The description of the collaborative learning situations has been provided by means of a unit of learning which is according to IMS-LD specification. Two types of unit of learning can be used: complete and incomplete. Complete units of learning are those that contain all necessary information for integration of actual tools in the stage of realization of the learning situations. Otherwise incomplete units of learning do not have such information, but only a generic description of needed tools. Therefore, incomplete units of learning can not be realized until they have been transformed to complete ones.

The main limitation of this system is not plan-supported. Even though that the incomplete units of learning seems to play the role of plans, but actually they are not independent plans.

3) Open collaborative grid service architecture (OCGSA)

The OCGSA [32] aims to provide a common framework for collaborative applications. In this architecture, the Grid service concept in OGSA (as low level service) is extended to Collaborative Grid service (high level service), by the extension of Grid service portType with metadata for group management and security. In parallel, the notification mechanism is also extended with the ability of predefinition of notification topics. Another new component in OCGSA compared to OGSA is the Event Archiving service that is responsible for managing the logs/messages exchanged between users/groups. However, this architecture only offers a basic level that does not include adequate concrete mechanisms for supporting realistic collaboration. This makes it very hard to be applied in development of real grid collaborative frameworks or applications.

4) Collaborative design

The collaborative design grid (CDG) [4] is a framework that aims to resolve two main problems in collaborative design: resource sharing and geographically distributed collaboration. The architecture of this framework bases mostly on OGSA, implemented Grid services on Globus Toolkit 3. This framework, however, has neither focused on supporting scripted work nor work plan.

5) Grid-enabled large scale

A framework called grid-based cooperative framework [5] has been developed aiming to build grid-enabled large-scale collaboration environment. This environment aims to support users to create large-scale and real/natural collaborations with some main features:

- Large scale collaboration (deeper and wider collaboration, hierarchical structures).
- Various cooperative modes (syn or asyn, intra-group or inter-group).

- Various coordination mechanisms (explicit or implicit or improvise).
- Integration of several coordination mechanisms into a single one

Even though this framework aims to develop large-scale collaboration environment, but it has not supported scripted learning situations which play an important role as work plans for learning processes. Without these plans, it is very difficult to manage the sequence of activities in learning processes, and this may lead to ineffective and uncontrollable learning processes.

V. CONCLUSIONS AND FUTURE WORK

Clearly because of the lack of understanding of the nature and the complexity of collective and collaborative activities, it has been difficult to build a generic collaborative Grid framework over a resource sharing infrastructure such as OGSA. This paper connects main ideas of the Activity Theory to Grid by proposing a Plan-supported Collaborative Grid Framework which allows three levels of collaborative from coordinating activity, cooperating activity to co-constructive activity. Interactions between these levels and components of the framework allow collaborative plans to be created and dynamically modified; objectives to be shared and co-optimised; and actions to be distributed and optimally executed by participants. The aim is to provide a generic Grid Framework for supporting collaborative work applicable to a wide range of application domains.

We have achieved the objective of integration of BPEL into Grid environment through the extension of the ODE Apache Engine (called G-ODE) which allows invocations of Grid services from BPEL processes.

We have also set up a Grid networked environment using Globus Toolkit 4 as the Coordination of Resource level, and in the process to implement the components of the Collective Activity level of our framework.

Last but not least, finding suitable techniques for automatic translation from BPMN to BPEL for integrating into our framework is also one of our research directions in the near future.

APPENDIX A: ACTIVITY THEORY

An activity has to be viewed in its entirety with its internal components and its specific dynamics. An activiy is composed of a subject (S), an object (O) and mediated by a tool (T) (see Fig. 3 in the Section II). A subject is a person, an agent or a group engaged in an activity and who directs the activity towards an object. The mediation is through the use of tools. An object can be a plan or a common idea or something that can be shared, manipulated, and transformed by the active participating subjects [16]. Tools can be a variety of artifacts such as electronic medical records, a computer system, a language. Each activity is distinguished from one another by their objects which manifest the objective of the activity. The object is connected to the motive of the activity and hence a subject's activity is motivated by the object.

By Leont'ev [9], an activity has three functionally subordinated hierarchical levels: activity, action, and operation.

- **Activity**: this level defines the components, the motivated subject, the object with its motive and objective, and the mediating artifacts.
- **Action**: this level explains what must be done to achieve the objective of the activity. Actions are goal-directed processes that are carried out to achieve different results, which in turn realize the object of the activity.
- **Operation**: operations are the processes that carry out the action. This operational level explains how the activity is performed automatically.

One important characteristic of these levels is the constant transformation that takes place between these levels (see Fig. 8).

Fig. 8. Transformation between three levels of activity.

For collaborative work, we are more interested in the collective aspects of the activity theory, especially the distributed collective activity. A collective activity is one that has more than one subject (a co-subject) and common object(s). The co-subject accounts for the processes of communication and coordination among individual subjects.
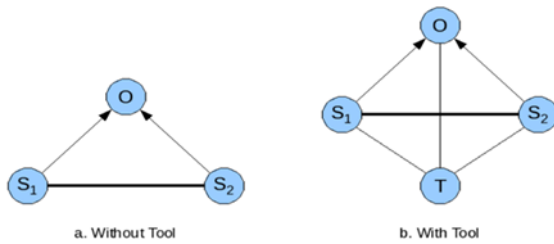

Fig. 9. Two kinds of collective activity.

The involved subjects must have act in common, must have a common motive associated with a common object. They must understand the common sub-goals of the action, and how these goals are supported by different artifacts. There are two kinds of collective activity (see Fig. 9).

A tool that has a role of collaboration between actions in a collective activity is called Collaborative Artifact (CA) [10]. Also in [10], the author presents three aspects of co-ordination of a Collaborative Artifact (CA).
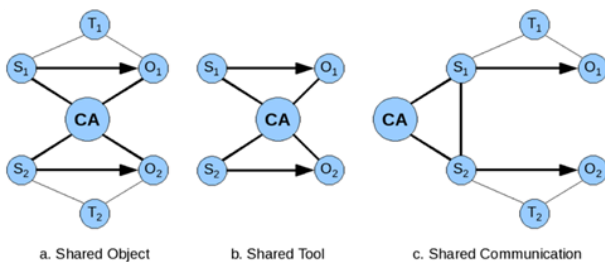

Fig. 10. Three coordination aspects of a CA.

- *Shared Object* (see Fig. 10a): With this aspect of coordination, subject S1 can see the object O2 of subject

S2 and vice versa. It means that two objects O1 and O2 can be shared by two subjects S1 and S2. This aspect needs access mechanisms that control the ways the subjects accessing the objects.
- *Shared Tool* (see Fig. 10b): With this aspect of coordination, the CA becomes a shared tool for both subjects S1 and S2. This aspect needs an allocation mechanism that controls the policy to use the tools.
- *Shared Communication* (see Fig. 10c): In this coordination aspect, the CA has a role as means of communication between subjects. Therefore, mechanism of identification is needed to identify members involved in communication.

Activity theory is useful for analyzing different levels of a collective activity and hence different types of collaboration. Three levels, arranged in a hierarchical structure, is identified according to Fichtner [33], Engestrom [34], and Bardram [10].

- *Co-ordinated activity*: At this level of collaboration, all the subjects (actors) work and coordinate according to a defined script. The subjects are working together to achieve a common objective, but they are not aware of the common objective.
- *Co-operative activity*: At this level of collaboration, the subjects focus on a common object and thus share the objective of the activity. The common aims are placed above the individual actions and their aims and only achieved through co-operation. In cooperative activity, the object is stable, but how to realize the activity might not be present or known. It means that the mediating actifacts/tools have not been recognized but they already exist, thus they do not need to be constructed.
- *Co-constructive activity*: In contrast with the two first levels, object of the activity at this level is not stable or even not existing. Hence, the object of work has to be constructed by coordination of the subjects. This level of activity typically takes place at organizational level.

APPENDIX B: WORKFLOW SYSTEMS

A. Basic Concepts

Business process and workflow

Workflow systems aim to describe and control business processes. Until recently, there has still been many different definitions and understanding of the two concepts, *Business Process* and *Workflow*, in both literature and technical reports of workflow management systems, as said in [35], p.4:

"There is little agreement as to what workflow is and which features a workflow management system must provide. Under the umbrella of the term `workflow', which is often used casually, people may be referring to a business process, specification of a process, software that implements and automates a process, or software that simply supports the coordination and collaboration of people that implement a process."

Therefore, it is necessary to clarify these concepts in order to make our discussion on these concepts and related ones more clear.

Business process (also called *procedure*) is defined in [36]:

"Procedure where documents, information or tasks are

passed between participants according to a defined set of rules to achieve, or contribute to, an overall business goal."

Fig. 11 shows an example of a simplified business process for processing credit requests in banks. The BPMN has been used to model this business process.

As shown in the Figure \ref{apd1.1}, the process starts when a customer fills in and sends a credit request to the Credit Card Management Office (task \textit{Sending Request}). Then, an employee in the Office will receive and check the personal information of the customer and the customers' requirements about processing card in the request. If all required information is acceptable (such as all required items have been filled in, and customer's income satisfies the bank's conditions, ect), then the request will be accepted (task \textit{Accepting Request}). Otherwise, the request will be rejected (task \textit{Rejecting Request}), and this decision will be announced to the customer (task \textit{Announcing Rejection}). After the acceptance of the request, two independent tasks must be done: \textit{Producing Credit Card} and \textit{Announcing Acceptance}: confirming the acceptance of the request and informing the customer about the period of time when the card will be sent. Because these two tasks are independent, they can be processed in parallel to reduce the processing time. After all of these two tasks have been finished, the credit card will be sent to the customer (task \textit{Sending Card}). Finally, the process finishes right after whether the card or the rejection desicion sent to the customer.
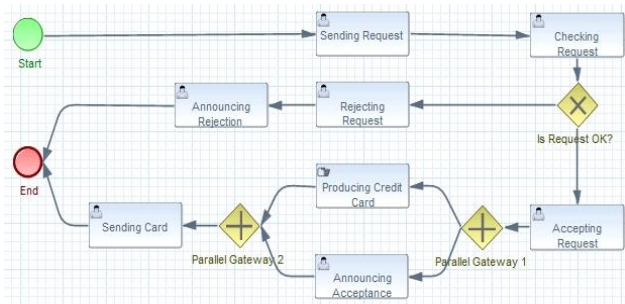

Fig. 11. The business process of processing credit requests.

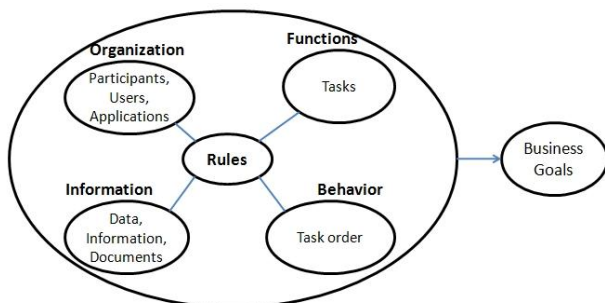Fig. 12 shows the structure of a business process following the above definition.

Fig. 12. Structure of a business process.

As can be seen in Fig. 12, a business process involves several aspects [37] (also called *perspectives* in [38]):

- **Organizational aspect**: this aspect (also called *Resource*) describes the resources (human, work tools or software applications) needed to execute the tasks in the process. For our example, the main resources are customers, bank employees, card production machines.

- **Functional aspect**: this aspect determines all necessary tasks (or activity) needed to be done to accomplish the process. For a complex business process, one task of it can become a sub business process which in turn can be decomposed into other sub-tasks. And the process of decomposition can repeat until all the decomposed tasks are small and simple enough so that they can not be split any further. This kind of task is called *atomic task*. And the other kind is called *non-atomic task* or *compound task*.

- **Informational aspect**: this aspect covers all data used in the process. As classified in [38], there are two types of data: control data and production data. While control data is used to control the execution order of tasks, production data relate directly to the tasks themselves, like their input and output.

- **Behavioral aspect**: this aspect expresses constraints of control flows among the tasks in the process. Besides basic control flows such as sequence, branch and iteration, business processes require other advanced control flows like AND-split, AND-join (or synchronization), OR-split, OR-join [39].

- **Operational aspect**: this aspect concerns how to execute the business process in a specific environment. This requires the clear and specified descriptions of all components of the business process. For example, for every task in the business process, the detail information, such as who or what application assigned to accomplish the task, is required.

Workflow is defined in [40] as:

"Computerized model of the business process, which specify all the parameters involved in the completion of this process"

As mentioned above, a business process consists of several aspects. Because a workflow is a model or representation of a business process, it also must represent all of these aspects. However, for theoretical and practical reasons, only one workflow never represents fully all of the aspects of one business process. Because if such workflow existed, it would become extremely complicated, and would be a combination of too many different modelling techniques for organizatiion, for functions, for data, for control flows, ect. Therefore, like in the software engineering where an expected system will be analysed and designed by many different tools and techniques, a business process should be represented by different workflows, and each one should only models one or several aspects of the business process.

For the above reason, our framework aims to support two types of workflows, BPMN and BPEL, that cover all of the aspects of business processes. The details of these types will be presented in the next section. Moreover, the capacity of automatic transformation between these two languages is also one of our research directions.

B. Workflow Languages

This section will present briefly two workflow languages which have been chosen to apply to our framework: BPMN and BPEL.

1) BPMN

BPMN has been adopted among domain analysts as a language for defining business process models. It has also

been used in specifying software requirements. In essence, BPMN provides a graphical notation for modelling business processes. Each BPMN file defines a Business Process Diagram (BPD), which is a kind of flowchart. A BPD consists of BPMN elements which can be devided into the following groups:

- **Flow Objects**: a flow object can be an activity, an event or a gateway.
- **Data**: representing data objects, input and output of activities.
- **Connecting Objects**: used to make connection between two flow objects or between a flow object and a data object. There are three types of connecting objects: sequence flow, message flow, and association.
- **Swimlanes**: When a process has two or more participants, it uses pools for representing these participants. Each pool represents a participant which can be a specific Partner Entity (e.g., a company) or a more general Partner Role (e.g., a requester, a consumer). A process that has two or more pools is called collaboration. A pool acts as a container of a process. Therefore, a collaboration aims to collaborate two or more processes to make a new collaborated process.
- **Artifacts**: Used to describe additional information for the processes. There are several types of artifacts: text annotation, group.

Among the aspects of workflows mentioned above, BPMN can only represent business processes in the three aspects: *Organizational*, *Functional* and *Behavioral*, as can be seen in two workflows in two Fig. 5 and Fig. 11. For example, the component *Swimlane* can be used to represent *participant* in the *Organizational*, the components *Activity* representing *Tasks* in the *Functional*, and the components *Gateways* and *Connecting Objects* representing *Control Flows* in the *Behavioral*.

Because the focus of BPMN is to represent business processes at high and abstract levels, it lacks the ability to represent them at low and executable levels such as the aspect Operational. Our framework overcomes this limitation of BPMN by using BPEL which will be presented in more detail in the next part.

2) BPEL

WS-BPEL (Web Service Business Process Execution Language) (or BPEL for short) is a language for specifying business process bahavior that is the interaction and composition of Web services. The processes in BPEL interact with external parners through Web service interfaces described by WSDL and manifest themselves as Web services. In BPEL, processes are defined by the XML language.

For composition of Web services, there exist two approaches: *orchestration* and *choreography*.

By orchestration way, only one service plays the role of main service which will invoke other subordinated services. Therefore, only the main service knows the sequence of activities, request and response states of the invoked services. In contrast, the role of main service does not exist in choreography way. The interactions between the involved services need some appropriate coordination mechanism.

The structure of a BPEL process (shown in Fig. 13) consists of two components:

- **Process**: which consists of activities, data, and related relationships. There is one type of activities called *invoke activity* which can be used to invoke (call) external partners.
- **Partners**: Other services that will be invoked from the process. Each partner exposes its accessible operations to the process through an interface called *portType*.

Fig. 13. Main components of a BPEL process.

Because the focus of BPEL is to represent business processes at the low level, where the workflows of the business processes can be executed automatically, it supports the ability to represent the business processes at the Operational aspect. Although BPEL can also be used to represent business processes at high levels such as Functional and Behavior aspects, but these representations of BPEL are more complicated and hard-understood when comparing with those of BPMN. For this reason, BPMN is often used to represent business processes at the high levels, and then the high level representations will be transformed into lower levels.

REFERENCES

[1] C. K. I. Foster and S. Tuecke, "The anatomy of the grid," *International Journal of High Performance Computing Applications,* 2001.
[2] V.-G. Guillermo, Y. A. Dimitriadis *et al.*, "Gridcole: A tailorable grid service based system that supports scripted collaborative learning," *Computers and Education,* 2007.
[3] E. G. M. Bardeen, T. Jordan, P. Nepywoda, E. Quigg, M. Wilde, and Y. Zhao, "The QuarkNet/Grid collaborative learning e-Lab," *Future Generation Computer Systems,* 2006.
[4] X. J. Z. Li, Y. Cao, X.-Y. Zhang, Y.-Y. Li, "Architecture of collaborative design grid and its application based on LAN," *Advances in Engineering Software,* 2007.
[5] Y. Y. Li, J.-L. Jiang, M.-L. Shi, "Build grid-enalbled large-scale Collaboration Environment in e-Learning Grid," *Science Direct,* 2006.
[6] I. Foster and C. Kesselman, *The Grid2: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, 2004.
[7] K. Schmidt and C. Simone, "Coordination mechanisms: Towards a conceptual foundation of CSCW systems design," *Computer Supported Cooperative Work,* 1996.
[8] M. Robinson, "Common artefacts in the design of computer support for collaborative work," *Developing CSCW Systems: Design Concepts*, 1993.
[9] A. N. Leont'ev, *Activity, Consciousness, and Personality*, Englewood Cliffs, 1978.
[10] J. E. Bardram, "Collaboration, coordination and computer support — An activity theoretical approach to the design of CSCW," *Aarhus*, 1998.
[11] D. Hoang and E. Lawrence, "An active Grid infrastructure for elderly care," *Journal of Telemedicine and Telecare,* vol. 14, pp. 363-367, 2008.
[12] D. B. Hoang, "Wireless Technologies and architectures for health monitoring systems," *Telemed,* 2007.
[13] B. T. Nguyen and D. B. Hoang, "Building a plan-supported grid collaborative framework," in *Proc. 2nd International Conference on Communications and Electronics*, Golden Sand Resort, Hoi An City, Vietnam, 2008, pp. 150-155.

[14] P. H. Miller, "Vygotsky and the sociocultural approach," *Theories of Developmental Psychology*, pp. 364-419, 2002.

[15] L. S. Vygotsky and A. Luria, "Tool and symbol in child development," *The Vygotsky Reader,* pp. 99-174, 1994.

[16] K. Kuutti, "Activity theory as a potential framework for human-computer interaction research," in *Context and Consciousness: Activity Theory and Human Computer Interaction*, B. Nardi, Ed. Cambridge: MIT Press, 1995.

[17] H. K. I. Foster, A. Savva, D. Berry, A. Djaoui, A. Grimshaw, B. Horn, F. Maciel, F. Siebenlist, R. Subramaniam, J. Treadwell, and J. Von Reich, *The Open Grid Services Architecture*, *Version 1.5*, 2006.

[18] I. Foster, "Globus toolkit version 4: Software for service-oriented systems," *Computer Science & Technology,* 2006.

[19] M. Papazoglou, J. W. Schmidt, J. Mylopoulos, W. Van Der Aalst, and K. M. Van Hee, *Workflow Management Models, Methods, and Systems*, MIT Press, 2002.

[20] J. E. Bardram, "Plans as situated actions: An activity theory approach to workflow systems," in *Proc. the 5th European Conference on Computer Supported Collaborative Work*, Lancaster, UK, 1997, pp. 17-32.

[21] O. M. Group, "Business process model and notation (BPMN) version 2.0," Object Management Group, 2009.

[22] C. Ouyang, V. D. Aalst, M. P. Wil, and H. M. Arthur, "From business process models to process-oriented software systems: The BPMN to BPEL way," *ACM Transactions on Software Engineering and Methodology,* vol. 19, pp. 1-37, 2009.

[23] J. Blox, *BPMN 2 BPEL - Research on Mapping BPMN to BPEL*, Department of Industrial Engineering, Eindhoven University of Technology, 2009.

[24] H. M. Arthur, V. D. Aalst, C. Ouyang, M. Dumas *et al.*, "Pattern-based translation of BPMN process models to BPEL web services," *International Journal of Web Services Research,* pp. 42-62, 2008.

[25] J. C. Recker and J. Mendling, "On the translation between BPMN and BPEL: Conceptual Mismatch between process modeling languages," in *Proc. The 18th International Conference on Advanced Information Systems Engineering*, 2006, pp. 521-532.

[26] OASIS, *Web Services Business Process Execution Language Version 2.0, OASIS Standard*, OASIS, 2007.

[27] F. Leymann, D. Roller, and S. Thatte, *Goals of the BPEL4WS Specification*, 2003.

[28] A. S. Foundation. Apache ODE. [Online]. Available: https://cwiki.apache.org/confluence/display/ODExSITE/Index

[29] B. T. Nguyen, D. B. Hoang, and T. T. Nguyen, "Enabling Grid Services from BPEL process using ODE engine," in *Proc. ICCSIT 2011*, Chengdu, China, 2011.

[30] B. T. Nguyen, D. H. Nguyen, and D. B. Hoang, "Towards a grid collaborative framework," *International Journal of Machine Learning and Computing (IJMLC),* vol. 2, no. 2, pp. 99-106, April 2012.

[31] B. T. Nguyen, D. H. Nguyen, and T. T. Nguyen, "G-ODE, an extension of the Apache ODE for grid services," *Journal of Computer Science and Cybernetics,* vol. 28, 2012.

[32] G. V. L. K. Amin and S. Nijsure, "Open collaborative grid service architecture," *Euroweb,* 2002.

[33] B. Fichtner, "Co-ordination, co-operation and communication in formation of theoretical concepts in instruction," Aarhus: Aarhus University, Psykologisk Institute, 1984.

[34] Y. Engestrom, "Learning by Expanding: An activity theoretical approach to development research," Helsinki: OrientaKonsultit Oy, 1987.

[35] D. Georgakopoulos, M. Hornick, and A. Sheth, "An overview of workflow management: From process modeling to workflow automation infrastructure," *Distributed and Parallel Databases*, vol. 3, pp. 119-153, 1995.

[36] D. Hollingsworth, *The Workflow Reference Model,* 1994.

[37] W. Mathias and G. Vossen, *Workflow Languages*, Springer Berlin Heidelberg, 1998.

[38] M. P. V. D. A. Wil, W. Mathias, and W. Guido, "Advanced Topics in workflow management: Issues, Requirements, and solutions," *J. Integr. Des. Process Sci.,* vol. 7, pp. 49-77, 2003.

[39] W. M. P. van der Aalst, A. H. M. ter Hofstede, B. Kiepuszewski, and A. P. Barros, "Workflow patterns," *Distributed and Parallel Databases,* vol. 14, pp. 5-51, 2003.

[40] G. Alonso, D. Agrawal, A. E. Abbadi, and C. Mohan, "Functionality and limitations of current workflow management systems," *IEEE Expert Intelligent Systems And Their Applications,* vol. 12, pp. 1-25, 1997.

Binh T. Nguyen is a PhD student at the School of Information Technology and Communication of the Hanoi University of Science and Technology. He received his master degree in computer science at the Francophonie Institute for Information (IFI).

His research interests include grid computing, cloud computing, collaborative systems and workflow languages.

Huu-Duc Nguyen is a lecturer at the School of Information Technology and Communication of the Hanoi University of Science and Technology. He received his Ph.D. degree in computer science from the Japan Advanced Institute of Science and Technology.

His primary research interest is in the area of foundation for programming languages, specifically design and implementation of parallel programming languages for multicore/manicore architechtures. He also has interests in problems relating to high performance computing, grid computing and cloud computing.

Thuy Thanh Nguyen is a vice-rector of the Hanoi University of Engineering and Technology which is a member university of Vietnam National University. His primary research interests include uncertainty, fuzziness and knowledge base systems, decision support systems. He is also interested in soft-computing and hybrid intelligent systems, data mining and knowledge discovery from data base, grid and parallel computing.

Doan B. Hoang is a professor in the School of Computing and Communications, Faculty of Engineering and Information Technology, the University of Technology, Sydney (UTS). He is a director of iNEXT - UTS Centre for Innovation in IT Services and Applications, a research centre at the University of Technology, Sydney for developing and nurturing innovation for the NEXT generation IT services and applications, including internet-enabled business applications, mobile health services, high-end visualization technologies, novel image processing architectures, and advanced video surveillance systems.

His research interests include next generation networks (security, quality of service, mobility, service-oriented architecture, peer-to-peer), broadband service architecture, collaborative grid and cloud computing, wireless sensor networks and e-health. He is currently leading research into establishing an innovation culture, reducing the cost of healthcare system through advanced technologies and assistive health grid/cloud, and generating wealth through innovative use of the broadband internet.