A Scientific Workflow Supported Environment over Hybrid Infrastructure for Aerodynamics Design

Seoyoung Kim, Hyejeong Kang, Yoonhee Kim*
Dept. of Computer Science
Sookmyung Women’s University
Seoul, Korea
{sssyyy77, hjkang, yulan}@sookmyung.ac.kr
*Corresponding Author

***Abstract*— As computing technologies develop, various studies for e-Science have been actively developed for many years. The advent of Cloud computing, for example, enables scientists to expand their research environments over supercomputers to on-demand and scalable resources. However, several performance drawbacks on the cloud computing can cause considerable obstacles in scientific domains, despite its strong merits. A hybrid infrastructure which consists of the existing and cloud architecture thus can produce a synergy effect for utilizing resources efficiently.**

In this paper, we proposed hybrid infrastructure-supported a scientific workflow environment for Aerodynamics design and demonstrated its superiority. Hybrid infrastructure covers grid and private cloud computing in this paper. Especially, we focused on improving performance by supporting hybrid infrastructure and efficient usages of physical resources. Since it offers diverse types of computing infrastructures including cloud computing, it can serve elastic resources regardless of the number of tasks for experiments or limitations of space and can reduce costs of time as well as budget during simulations.

***Index Terms*—Hybrid Infrastructure, Science Cloud, Scientific Workflow, e-Science, Aerodynamics Design**

I. INTRODUCTION

As computing technologies develop, various studies for e-Science have been actively developed for many years. The advent of Cloud computing, for example, enables scientists to expand their research environments over supercomputers to on-demand and scalable resources. However, several performance problems on cloud computing are considerable tasks in scientific domain. Therefore, a hybrid infrastructure which consists of the existing and cloud architecture can produce a synergy effect.

In this paper, we proposed hybrid infrastructure-supported a scientific workflow environment for Aerodynamics design and demonstrated its superiority. Hybrid infrastructure covers grid and private cloud computing in this paper. Especially, we focused on improving performance by supporting hybrid infrastructure and efficient usages of physical resources. Since it offers diverse types of computing infrastructures including cloud computing, it can serve elastic resources regardless of the number of tasks for experiments or limitations of space and can reduce costs of time as well as budget during simulations.

The continuation of the paper is structured with the sections as follows; we introduce some related works in Section 2, and then explain properties of targeted scientific application in Section 3. In Section 4, we present architecture of our scientific workflow environment with hybrid infrastructure. After that, performance evaluation of the environment is discussed in Section 5. We also present implementations of the scientific workflow environment in final section, and conclusions with the future works are followed.

II. RELATED WORK

There are several integrated researches that can utilize the distributed storages and computing resources, orchestrate the scientific data analysis tools across various infrastructures as supporting scientific workflows. Most of these researches, such as GridFlow, P-GRADE, typically have been based on grid computing.

GridFlow [3] is a system which has offered workflow management services on large-scale grid computing environments. It enables scientists to execute and monitor scientific simulations by providing workflow management services via its workflow management system. However, this system requires additional processes of installation and advanced settings for connecting with computing resources. These processes make users difficult to use all alone.

P-GRADE [4] portal provides services enabling users to create, execute, and manage big scaled-jobs by allowing accesses to various grid resources (EGEE, UK, NGS, VO, OSG, TeraGrid, etc.). The portal offers high-level middleware to manage and to control resources on grids.

Meanwhile, recent advent of cloud computing makes scientists to expand their research environments over grids to on-demand resource based clouds. Hence, there exist various researches [5], [6], [7] about multi-platforms-based problem solving environments these days.

VGrADS(Virtual Grid Application Development Software) [5] project expanded its existing computing infrastructures for high performance computing to Amazon EC2 which is public cloud system with a scheduling policy for fault-tolerance.

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2011-0027486)

In case of ASKALON [6] project, it is still working on achieving an integrated environment for application development, where user can access to both grids and clouds.

These researches which are mentioned above do not support scientific workflow systems yet. Therefore, it is difficult to serve management of complex flows which are composed of a number of tasks. As a workflow system is one of the essentially required facilities for scientists, they need to improve their system and to add a workflow system.

Similarly, Nimrod [7] project supports high-throughput computing by offering different members of tool family for large-scale and numerical researches. Nimrod tool family covers a variety of tools. The project has built and provides cooperated operations with other projects such as Globus, Kepler. For example, Nimrod/K [16] is a tool for integrating another Nimrod tools into the Kepler workflow engine along with a novel dataflow mechanism. Recently, the project added an interface to Amazon's EC2, allowing users to mix grid resources and clouds. The provided tools in the project, however, are too varied for non-experts like scientists to use commonly, even though it provides similar resource infrastructure support with our project.

Our project provides an intuitive problem solving environment for creating and managing scientific workflows as well as performance support with hybrid infrastructure.

III. Characteristics of Targeted Application

We target CFD (Computational Fluid Dynamics) [2] applications and set Aerodynamics as a prototype among a number of scientific domains. Aerodynamics is a branch of dynamics concerned with studying the motion of air, particularly when it interacts with a solid object. It usually uses HTC (High Throughput Computing) applications to solve numerical problems during simulations. For these simulations, high performance computers are essentially required to perform calculations of the simulation and also workflow management facility for controlling flows of numerous computational jobs is essentially needed.

The simulations as introduced in the previous paragraph generally have a similar pattern of executions. **Figure 1** shows an example of execution pattern for a simulation in aerodynamics domain. Each step contains a number of computational tasks requiring frequent data communications and can be constructed in different sizes of computational tasks.

Firstly, the simulation starts with 'Flow Analysis' step. This step typically has one or more sequential tasks. The task can be executed in a way of serial, parallel, or both. During this step, all executions require frequent data transmissions and result checking by users.

One step among these steps is for HTC and the second step in **Fig.1** corresponds to this step. This HTC step contains a set of executable files which consist of a number of parallel or/and serial applications that find optimal values of aircraft design for design optimization. In this step, users mainly carry out experimental design for analyzing parameter effects on output of application execution. To gain throughput necessary to get results in a timely fashion, during this step, it is necessary to use multiple distributed computing resources, which comprise varied hardware, and generally run different software stacks. However, it is difficult to utilize various computing resources including grids (such as PRAGMAGrid, EGEE, etc..,) which have not had a widespread adoption outside of scientific HTC, since it has been specifically tailored for limited application domain.

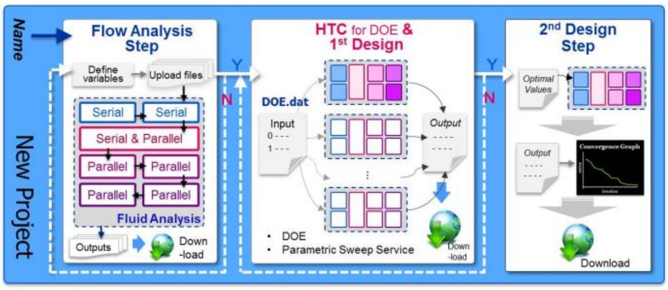


Fig. 1. Example of execution pattern from CFD applications

Recently, virtualization technology which is the key of Cloud computing, promises to become a good solution for the problems of resources. To achieve improvements of throughput for HTC, therefore, we expanded resource infrastructure of our experimental system by supporting cloud computing and finally composed hybrid infrastructure. Although virtualization provides greater flexibility, however, it has runtime overheads. Sometimes these overheads can be critical for scientific executions because a simulation includes a lot of tasks and each runtime affects total duration of the simulation in many cases. Overall, we utilize mixed infrastructure using various types of resource infrastructure and it may reduce or eliminate drawbacks of each structure.

IV. Architecture

In this section, we discuss architecture of services and underlying resources. We especially focus on some extensions in both side of portal services and computing infrastructures.

A. Service Architecture

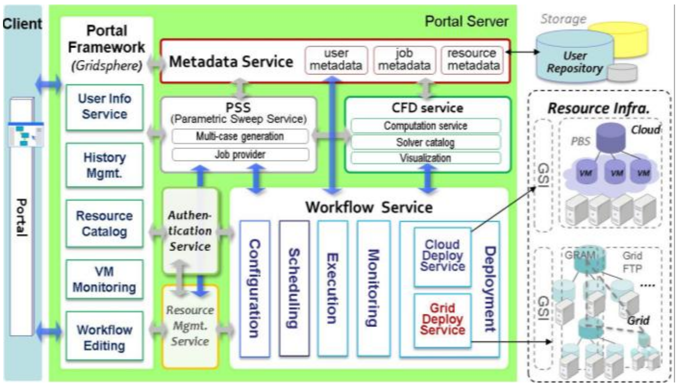


Fig. 2. Service Architecture

We have performed several researches of an integrated e-Science environment where scientific workflow management system is supported [8], [9], [10]. In this research, we extended the earlier work [8], [9] by serving additional resource infrastructure support. So, the expanded service organization of this research is similar with the earlier researches' [8], [9], [10], except services about cloud management and some modified features.

As main services of this system, there are **PSS** (Parametric Sweep Study) Service, **CFD** (Computational Fluid Dynamics) Service, **Workflow** Service and **Metadata** Service. In addition, several *portal framework* services, *authentication* service and *resource management* service are defined as complementary services. **Figure 2** depicts the whole architecture of these services. In this paper, we explain these services in detail more focusing on newly added parts. The details of rest services which are not mentioned in the below are described in the earlier paper [10].

Workflow Service supports execution and management of entire flow of tasks that is made of consecutive and parallel jobs as created execution procedure by users via a visual workflow editor of portal system. Sub-workflows or a set of jobs are configured to a workflow in *Configuration* phase. In this phase, it usually interfaces with Metadata service, so it allows users to manage history of workflow execution and to configure reformed workflow by enables users to re-load their works. In addition, the configuration phase deals with organizing a DAX files for workflow description. *Scheduling* phase handles scheduling tasks according to current status of resources. The phase also controls placement of jobs which are included in a HTC unit by considering locality between them. *Execution* phase means managing job execution using data which is generated by parsing the specified workflow file, DAX file. After then, *Monitoring* phase supplies some state information of jobs to user, and determines whether users want to proceed, or not. Moreover, the phase for monitoring serves state information of workflow execution in real-time. *Deployment* phase has two modules; one is for cloud infrastructure and the other is for non-cloud infrastructure (e.g. grid, local clusters). *Cloud Deploy Service* takes care of configuring as well as deploying VM and enables job submission into a virtually configured cluster which is based on Cloud. In addition, the service controls the number of VMs according to the current status of resources and workloads. Besides, *Grid Deploy Service* covers a range of different physical environments such as grid, clusters, and supercomputer, etc. We used Globus Toolkit [11] as a middleware and OpenNebula [12] for configuring cloud infrastructure. Details of resource architecture will be followed in the next subsection.

The rest of services like **PSS Service**, **CFD Service** and **Metadata Service** also perform interfacing with above services or each other. Users are able to access experimental services via a web portal, where *Graphical Workflow Editing* service with its historical data and resources information is available. *VM monitoring* usually provides status information of virtual machines with their properties. For some specialized users having administrative privileges, it is possible to deploy one or multiple VM(s) manually with the desired properties. For example of property, users can input virtual machines' properties that are the number of CPUs, memory size and the number of VMs being deployed.

B. Resource Extention Architecture

To support multiple computing resources, we develop cloud adaptor (also called as *Cloud Deploy Service*) and associated modules which are capable of interfacing our system with cloud infrastructure. **Figure 3** depicts the underlying computing infrastructures. We name the architecture *Hybrid Infrastructure*, since it has two types of computing resources, physical and virtual. Currently, our system provides computation services on grids, clusters and private cloud as shown in **Fig. 3**. The system interacts with grids or physical clusters just via GRAM (Globus Resource Allocation Manager). In case of cloud, the system interacts with an entry machine which is a front-end machine for virtual clusters and simultaneously is an entry machine for virtual machines to interact with GRAM server.

The virtual cluster are composed with cloud middleware, OpenNebula, and each virtual machine of the virtual cluster performs as a slave computing node. Therefore, it is possible to increase the number of computing nodes dynamically and to create homogeneous computing environments with virtualization technologies. It would be better using a number of computing nodes with low performance than a node with high performance, since parallelism of task executions results in reducing the number of waiting jobs when a large number of jobs were submitted. Besides, once a front-end machine is configured, it is easy to add or remove computing nodes using XML-RPC interfaces in OpenNebula.

We also develop a module for managing pre-configured VM images. Initially we set some environmental settings (such as locations of shared directories, etc.) and installed software (such as PBS-mom, etc.) on a virtual machine. In addition, we created a shell script for setting IP address and SSH connections. With this pre-configured virtual machine and script, we duplicated VM whenever system needs, and then let the VM run the script as soon as it finishes booting.

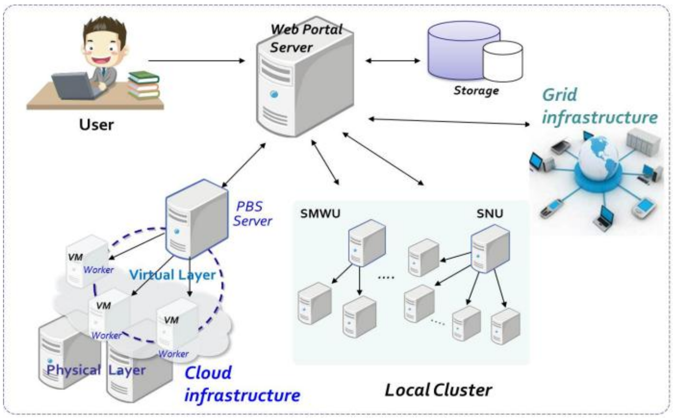


Fig. 3. Architecture of Hybrid Infrastructure

Currently, we support only private cloud. We have a plan to expand the architecture into public clouds such as Amazon EC2 [13] with a policy for economic computing.

V. PERFORMANCE EVALUATION

In this section, we verify improvements of performance by performing simple experiments. Executing an application for a simulation of aerodynamics design takes 24 hours or more generally. Total duration of a simulation, furthermore, takes one month and more [14]. So, we use the simplified forms with the same executional patterns of CFD applications and the formed applications are manufactured by experts in aerodynamics for demonstrations to be used for proof-of-concept. The simulation which is demonstrated by the simplified applications is performed in the same way of execution that was mentioned before and is carried on three conditions of infrastructures. The conditions are classified into two groups; *Single Infrastructure* and *Hybrid Infrastructure*. Single infrastructure covers two categories which are *Cluster* (or *grid*) and *Cloud* group. In case of our earlier works' infrastructure, their records belong to *Clusters* of single infrastructure. Additionally we perform simulations on Cloud as one of comparable conditions of single one. Hybrid infrastructure is composed of physical and virtual resources together. **Table I** describes information of resources and experimental settings. After finishing the simulations on each condition, we analyze the execution records from 100 times of simulations.

TABLE I. COMPUTATIONAL RESOURCES USED FOR EXPERIMENT WITH EXPERIMENTAL SETTING

	Single Infrastructure		**Hybrid Infrastructure**
	Cluster	***Only Cloud***	***Cluster + Cloud***
Specifi -cations	AMD Phenom(tm) II X4 945 Processor	Intel(R) Core(TM)2 CPU	AMD Phenom(tm) II X4 945 Processor + Intel(R) Core(TM)2 CPU
The # of physical cores	**40**	**4**	**40**(local cluster) + **4**(cloud)
VM Properties	-	1 Core, Mem.1GB (per VM)	1 Core, Mem.1GB (per VM)

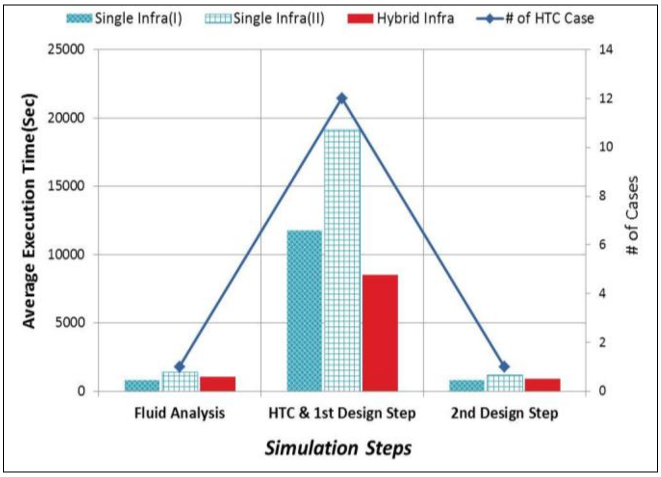


Fig. 4. Comparisons of Avg. Execution time among Single Infra(I), (II) and Hybrid Infrastruncture

TABLE II. COMPARISON OF AVERAGE JOB EXECUTION TIME

Type Step	***Single Infra(I)***	***Single Infra(II)***	***Hybrid Infra***
Fluid Analysis Step	818	1418	1055
HTC & 1stDesign Step	981 (total 11765)	1593 (total 19124)	711 (total 8542)
2nd Design Step	811	1179	924
Total	13394	12190	10521

(Unit: Seconds)

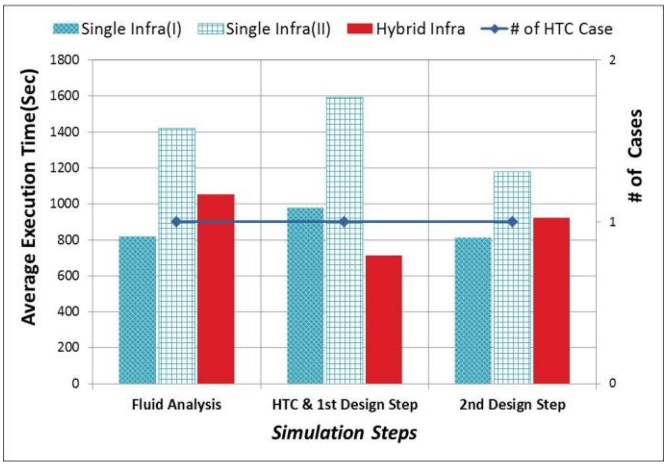


Fig. 5. Comparisons of Average execution time

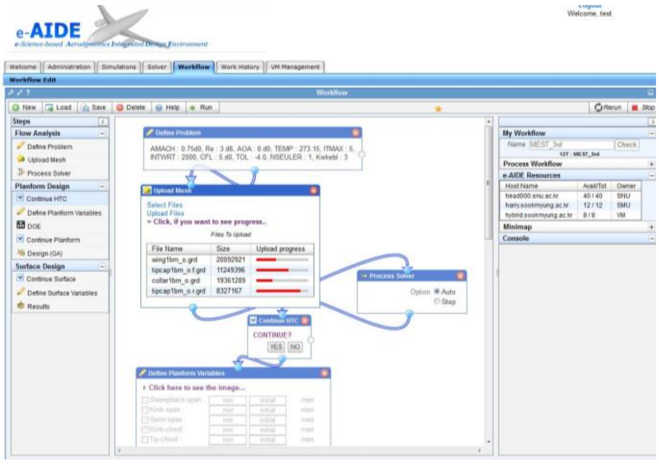
For comparisons, we use average execution time per task and average duration time during the simulation as metrics. **Figure 4** shows result of the evaluations. Average execution time is plotted along the left **Y** axis in the graph (**Fig. 4**) and the average number of submitted tasks (we also call a HTC job as a case) on each step is plotted along the right **Y** axis. And each step of the simulation is presented along **X** axis.

In the first step (Fluid Analysis), single infra (II) has the highest time costs among the conditions. As mentioned in Section III, this step is comprised of one or sequential tasks typically. Therefore, it is individual job's duration that has a big effect on total duration of this step. In cloud computing, performance overheads occurred frequently, because it is based on virtualization technology. When we compare single infra (I) and hybrid infra which have the lower time costs than single infra (II), single infra (I) has a lower value than hybrid in the same reason of single infra (II).

In the second step (HTC and 1st Design step), there has been a relative rise in terms of the mean number of jobs among whole steps. In this step, you can notice that average duration of hybrid infrastructure is the shortest among the other conditions' that. That is, utilizing both of resources (physical and virtual) properly results in improvements of performance, since they can be compensated each other for their own problems. Grid or cluster environment is capable of offering high performances in computations with limited resources and cloud environment can increase parallelism of executions with flexible virtual resources.

The final step (2nd Design step) has the similar results with the first step as well, since they have similar characteristics of execution. **Table II** shows expression of the results numerically and **Fig. 5** describes comparisons of average execution time per job. When we compare total execution time of hybrid infra with single infra (I), (II), result shows that hybrid infra has the shortest duration among three of conditions.

VI. IMPLEMENTATIONS



AMACH : 0.75d0, Re : 3.d6, AOA : 0.d0, TEMP : 273.15, ITMAX : 5, INTWRT : 2000, CFL : 5.d0, TOL : -4.0, NSEULER : 1, Kwkebl : 3

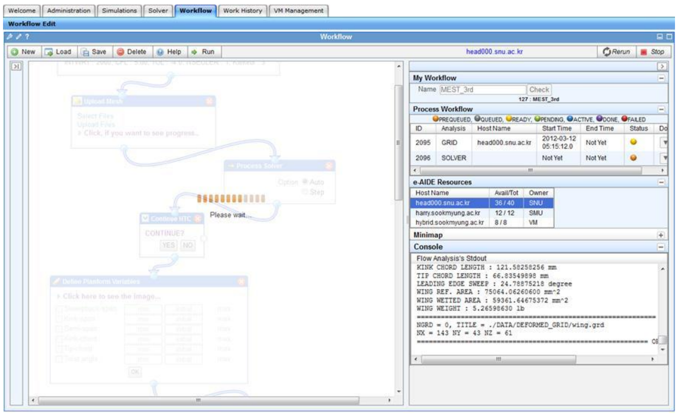
File Name	Size	Upload progress
wing1bm_o.grd	20092921	
tipcap1bm_o.f.grd	11249396	
collar1bm_o.grd	19361289	
tipcap1bm_o.r.grd	8327167	

Name MEST_3rd

127 : MEST_3rd

Host Name	Avail/Tot	Owner
head000.snu.ac.kr	40 / 40	SNU
harry.sookmyung.ac.kr	12 / 12	SMU
hybrid.sookmyung.ac.kr	8 / 8	VM

Fig. 6. Implementation of Scientific workflow portal



Name MEST_3rd

127 : MEST_3rd

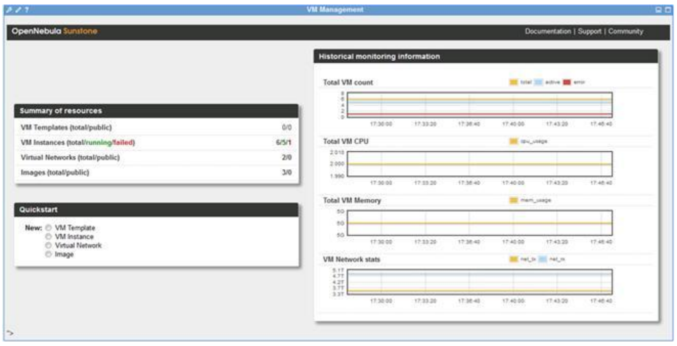
ID	Analysis	Host Name	Start Time	End Time	Status	Do
2095	GRID	head000.snu.ac.kr	2012-03-12 05:15:12.0	Not Yet	●	▼
2096	SOLVER		Not Yet	Not Yet	●	▼

Host Name	Avail/Tot	Owner
head000.snu.ac.kr	36 / 40	SNU
harry.sookmyung.ac.kr	12 / 12	SMU
hybrid.sookmyung.ac.kr	8 / 8	VM

Flow Analysis's Stdout

```
KINK CHORD LENGTH : 121.58258256 mm
TIP CHORD LENGTH : 66.83549898 mm
LEADING EDGE SWEEP : 24.78875218 degree
WING REF. AREA : 75064.06260600 mm^2
WING WETTED AREA : 59361.64675372 mm^2
WING WEIGHT : 5.26598630 lb
==============================================================
NGRD = 0, TITLE = ./DATA/DEFORMED_GRID/wing.grd
NX = 143 NY = 43 NZ = 61
============================================================= OI
```

Fig. 7. Screen of Scientific workflow portal during Running state



Summary of resources

VM Templates (total/public)	0/0
VM Instances (total/running/failed)	6/5/1
Virtual Networks (total/public)	2/0
Images (total/public)	3/0

Fig. 8. Virtual Machine Monitoring page which is based on Sunstone[15]

This paper extends and refines a web-based scientific workflow editing environment [10] to exploit multiple infrastructures. **Figure 6** shows a screenshot of a designed workflow on our scientific workflow portal environment which loads pre-figured template and creates a workflow by setting parameter values and creates input files.

Layout of the portal is same as before [10], but a lot of interfaces are added to deliver more information and interact. The right panel includes a table of resource list, console panel to show standard outputs as well as real-time job status for a workflow. A resource list consists of host name, type, and the number of available cores. A console panel which exists on below part of right in **Fig. 7** provides real-time standard out during the run a workflow. Once a workflow is completed to construct, users can submit the workflow by clicking *RUN* button on toolbar and can submit the workflow to resource that they selected in a resource list. Users can select a resource whatever grids or clouds, and then hostname of selected resource is displayed at toolbar.

During the workflow running, a center panel is disabled to prevent editing or actions which can effect on a running workflow as shown in **Fig. 7**. When a job is finished, a download button in the Process Workflow panel is enabled and by clicking this button, users can download result files besides create a graph. If users detect abnormal parameter values, they can rerun or stop the job by clicking **Rerun/Stop** button on the toolbar.

In addition to these abilities, VM Mgmt. tab is offered to monitor VMs in cloud resource, as shown in **Fig. 8**. This VM monitoring interface consists of some functions of Sunstone [15] which is a monitoring tool of OpenNebula [12] used to build cloud environment in this portal. This tab provides the number of available cores, status, and information about used VMs until now. To the scientists who are the main user of this portal, previous work data of workflows is matter to compare and analyze, so it is important to provide interfaces for checking these historical data. Thus, this scientific workflow environment offer Work History tab to users as shown in **Fig. 7**. Work History tab is separated by each management unit as table and bottom table which depicts **JOB**'s records is same as *Process Workflow* panel in downloading result files and creating a graph.

VII. CONCLUSION AND FUTURE WORKS

So far, we introduced a scientific workflow execution environment which is based on hybrid infrastructure. We specially targeted various computational scientific domains and showed the implementations for Aerodynamics as a prototype. Besides, we showed performance evaluation with a purpose of verifying improvements when compared to the earlier works. Overall, the expanded system resulted in reduction of total durations of simulation, since it is able to provide appropriate computing infrastructure regardless of various characteristics of each step. In addition, we also presented the implementations of the system.

In the future, we will expand our resource infrastructures into public clouds such as Amazon EC2 and apply some scheduling policies considering cost and deadline.

ACKNOWLEDGMENT

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2011-0027486).

REFERENCES

[1] Kumwon Cho, Sangdong Lee, Jeesoo Lee, “Domestic/Foreign present conditions and trends of e-Science”(KOREAN), Special issue of e-Science, Pub. of Special issue on Physics and High Technology, pp. 10–16, Oct.2005.

[2] CFD, http://www.cfd-online.com/

[3] Junwei Cao, et al. “GridFlow: Workflow Management for Grid Computing”, Third IEEE International Symposium on Cluster Computing and the Grid(CCGrid’03); ISBN: 0-7695-1919-9

[4] Kacsuk, P., Kiss, T., Sipos, G., “Solving the grid interoperabliility problem by P-GRADE portal at Workflow level”, Future Generation Computer Systems Vol.24, no.7, pp.744-751, 2008.

[5] Ramakrishnan, L., et al. “VGrADS: Enabling e-Science Workflows on Grids and Clouds with Fault tolerance”, In Proceedings of SC, 2009.

[6] Fahringer, T, Prodan R, Duan R, Nerieri F, Podlipnig S, Qin J, Siddiqui M, Truong HL, Villazon A, Woeczorek M, (2005, Nov.13-14) “Askalon: A Grid application development and computing environment”, In Proceeding of the 6th IEEE/ACM international conference on Grid, 2005, IEEE,Seattle, Washington DC, pp.133-141.

[7] Bethwaite, B, Abramson, D, Bohnert, F, Garic, S, Enticott, C, Peachey, T, “Mixing Grids and Clouds: High-Throughput Science Using the Nimrod Tool Family”, Cloud Computing Computer Communications and Networks, Springer London 2010, Vol.10, Part 2,pp.219-237.

[8] Seoyoung Kim, Hyejeong Kang, Kyoung-a Yoon, Yoonhee Kim, Cinyoung Hur, Chongam Kim, "Scientific Workflow support of multi-stage experiments on an e-Science environment", ICONI 2010, Crimson Resort, Mac tan Island, Philippines, pp. 663~667, Dec. 16-20, 2010.

[9] Seoyoung Kim, Kyoung-a Yoon, Hyejeong Kang, Yoonhee Kim, Chongam Kim, "Supporting an Interactive Scientific Workflow in Aerodynamic Analysis over e-Science Environment", Asia-Pacific Network Operations and Management Symposium (APNOMS) 2011, Taipei, Taiwan, Sep. 21-23, 2011.

[10] Seoyoung Kim, Kyoung-a Yoon, Yoonhee Kim, Chongam Kim "A Scientific Workflow Service with Intuitive Problem Solving Environment for Aerodynamics", ICACT 2012, Phoenix Park, Pyeongchang, Korea(South), Feb. 19-22 2012.

[11] Globus Toolkit, http://www.globus.org/

[12] OpenNebula, http://opennebula.org/

[13] Amazon EC2(Elastic Compute Cloud), http://aws.amazon.com/ec2/

[14] http://en.wikipedia.org/wiki/Aerodynamics

[15] SunStone, http://opennebula.org/documentation:archives:rel3.0:sunstone

[16] Abramson D, Enticott C, Altinas I, “Nimrod/K: Towards massively parallel dynamic grid workflows”, Proceedings of the 2008 ACM/IEEE conference on supercomputing, IEEE Press, Austin Texas, pp.1-11, 2008..