g-INFO portal: a solution to monitor Influenza A on the Grid for non-grid users

Trung-Tung Doan, Quang-Minh Dao, Trong-Hieu Vu, Hong-Phong Pham,
Vincent Breton, Hong-Quang Nguyen, J. Salzemann, Yannick Legre,
Thanh-Hoa Le, Johan Montagnat

HAL Id: hal-00683104

https://hal.science/hal-00683104

Submitted on 27 Mar 2012

g-INFO portal: a solution to monitor Influenza A on the Grid for non-grid users

Trung-Tung DOAN[a,1], Quang-Minh DAO[e], Trong-Hieu VU[f], Hong-Phong PHAM[e]
Vincent BRETON[b], Hong-Quang NGUYEN[a], Jean SALZEMANN[b], Yannick LEGRE[c],
Thanh-Hoa LE[d] and Johan MONTAGNAT[g]

[a] *Institut de la Francophonie pour l'Informatique,*
UMI UMMISCO 209 (IRD/UPMC)
42, Ta Quang Buu, Hanoi, Vietnam
[b] *Laboratoire de Physique Corpusculaire, CNRS/IN2P3,*
24 avenue des Landais, BP 10448, F-63000, Clermont-Ferrand, France
[c] *HealthGrid association*
36 rue Charles de Montesquieu, 63430 Pont-du-Château, France
[d] *Institute of Biotechnology, Vietnam Academy of Sciences and Technology*
18 Hoang Quoc Viet, Cau Giay, Hanoi, Vietnam
[e] *High Performance Computing Center, Hanoi University of Science and Technology*
1 Dai Co Viet, Hanoi, Vietnam
[f] *Institute of Information Technology, Vietnamese Academy of Science and Technology*
18 Hoang Quoc Viet, Cau Giay, Hanoi, Vietnam
[g] *Laboratoire d'Informatique, Signaux et Systèmes de Sophia-Antipolis*
I3S - UMR6070 - UNSA CNRS
2000, route des Lucioles - Les Algorithmes - bât. Euclide B - BP 121 - 06903 Sophia
Antipolis Cedex - France

Abstract. In this paper, we introduce a portal for monitoring Influenza A on a grid-based system. Influenza A keeps on being a major threat to public health worldwide; especially if one virus can mutate itself so that it acquires the capacity for human to human transmission of H1N1 as well as the high death rate of H5N1. The existing g-INFO (Grid-based Information Network for Flu Observation) project provides a complete system for monitoring flu virus on the Grid. We present here a portal that operates on top of the g-INFO system as a solution for non-grid users to utilize grid services for analyzing molecular biology data of Influenza A.

Keywords. Grid, portal, public health informatics, flu, surveillance network, phylogeny.

Introduction

The 2009 pandemic of H1N1 (or swine flu) has confirmed that continuing caution, preparation, and strong public health research competence are vital to face emerging

---

[1] Corresponding Author.

health threats. On the other hand, H5N1 virus (or avian flu) has continued to evolve and cause outbreaks. Therefore, more actions have been done on global influenza monitoring. Epidemiology is one of the bases to detect, identify and analyze health problems related to flu caused by Influenza A. Molecular epidemiology's issues concerning analysis of genomes, sequences and structures are helped by a wide-ranging of bioinformatics tools which require adequate computing resources [1]. Data for molecular analysis are provided by several public database resources such as NCBI (National Center for Biotechnology Information) [2] or BioHealthBase [3]. In some cases like H5N1, data is not always public; it remains non-synchronized with public database resources. Therefore, systems are non-interoperable, leading to data and application silos, duplication of work, and costly efforts to integrate data. The grid technology is one of the most capable and vigorous concepts to address such problems with computing resources and data sharing.

Taking advantage of the grid power, the g-INFO project (Grid-based International Network for Flu Observation) provides a complete system for monitoring Influenza A. g-INFO focuses on running and connecting various bioinformatics programs, recognized for their accuracy and speed, to continuously reconstruct a robust phylogenetic tree from a set of sequences publicly available and daily updated. The sequences, extracted from existing data sources populated by the scientific community, are processed dynamically using a Service Oriented Architecture principles (SOA) to compose phylogeny pipelines and Grid technologies to handle the computing load. g-INFO supports both automatic pipelines [4] and dynamic pipelines [5]. The automatic pipeline contains three well-known algorithms commonly used for phylogeny analysis and runs daily on updated data. The dynamic pipeline is configurable so that an expert can choose which tools he wants as well as the order of the tools in the workflow for his specific analysis. In the g-INFO system, pilot jobs are automatically submitted to the Grid therefore users who do not have much grid experience can run their pipelines on the grid. However, sometimes they have to upload their input data (which was not published yet) to the Grid before running a pipeline. With non-grid expert users, a simple operation such as uploading files to the grid is not always an easy task.

In this paper, we present the g-INFO portal that eases the process of creating and running phylogenetic pipelines in the g-INFO system. With g-INFO portal, user can create and save workflow templates to run many pipeline instances. Outputs can be visualized with a built-in visualization tool. Non-grid expert users can use the portal to monitor Influenza A.

1. Implementation

The g-INFO system is implemented and deployed on the EGI (European Grid Initiative) infrastructure, which is based on a Grid Middleware stack called gLite [6]. gLite is widely use and extended in several EU Grid projects. It contains set of components that enables a production quality Grid. gLite is VO (Virtual Organization) based and grid services are shared via VOMS (VO Membership Service). In the infrastructures based on gLite, jobs are managed by Workload Management Systems (WMS) and run on Computing Elements (CE). Data are stored on Storage Elements with a file catalog system.

Besides gLite, a large-scale deployment of the phylogenetic pipeline requires the use of an environment for job submission and output data collection: the WISDOM

Production Environment (WPE) [7]. To enable workflows in g-INFO, we use the MOTEUR workflow engine [8]. The following subsections will describe briefly the WPE, MOTEUR and how the portal interacts with components available in the g-INFO system.

1.1. WISDOM Production Environment (WPE)

The WPE is a middleware designed as an experiment management environment that settles on top of grid systems or more generally on computing resources such as clusters. It handles data and jobs, and shares the workload on all the integrated resources even if they adopt different technology standards. Based on this middleware, it is possible to build web-services that interact with the system. The middleware is considered as a set of generic services acting as an abstraction level for the specific resources and therefore providing a generic management of data and jobs so that the application services can use any of the underlying systems in a very transparent way. Users are not interacting directly with the grid resources and they are not expected to know how it works since they are just interacting with the top-level services just like with any other web service. The three main components in the WPE are: the TaskManager, the JobManager and the WISDOM Information System.

- The Task Manager interacts with the client and hosts the tasks created by the client;

- The Job Manager submits the jobs to the Computing Elements (CEs) where the tasks managed by the Task Manager will be executed;

- The WISDOM Information System uses AMGA (ARDA Metadata Grid Application) to store all meta-data needed by the Job Manager.

One drawback of gLite is the waiting time for a job to be submitted and scheduled on a WMS. The WPE overcome this bottleneck by using pilot jobs, which are automatically submitted by the Job Manager. A pilot job is a generic job programmed to run many different tasks based on given parameters. Once a pilot job is submitted successfully on the Grid, it will look for a g-INFO's task in the Task Manager. If a pilot job finds a task, it will grabs and executes this task. The WPE keeps a pilot job alive as long as possible so that during its runtime, it can execute many tasks.

In the Task Manager, several tools are deployed for phylogenetic analysis:
- Blast is for searching regions of similarity among sequences [9];

- Muscle and Hmmalign are for sequences' alignment [10];

- Gblocks's package is for sequences' curation [11];

- PhyML and Fasttree are for constructing phylogenetic trees [12].

With these available tools in the TaskManager, users can use the MOTEUR workflow system to create phylogenetic workflows. MOTEUR is described in the next subsection.

1.2. MOTEUR

MOTEUR is a workflow designer and enactor, developed by I3S and CREATIS laboratories, which is interfaced with the gLite grid middleware and handles

application services asynchronously [8]. For this reason it is perfectly suited to handle long makespan workflows such as g-INFO. MOTEUR provides a very flexible framework to run g-INFO as the workflow can be built from a set of independent services, and can be modified interactively through a graphical interface. Furthermore, it provides advanced data parallelism constructs well adapted to exploit distributed grid resources. MOTEUR can also be run in command line allowing a daily and automatic execution of the g-INFO pipeline. The use of a workflow engine such as MOTEUR is very relevant in the context of a bioinformatics platform with modular services since designing as many pipelines as there are workflows, users or execution conditions would become untraceable. A bioinformatics platform should be a toolbox of independent tools and algorithms, and a workflow engine will be used to handle all those services altogether in a coherent way at runtime without adaptation of the users on the services themselves.

In order to create a g-INFO workflow with MOTEUR, we provide for each task in the Task Manager a corresponding asynchronous web service. Following the MOTEUR' convention, each web service contains at least 4 main functions:

- submitSequence: creates a corresponding task in the Task Manager, i.e. BLAST asynchronous web service will create BLAST task

- isFinished: to check if a task is finished so that the workflow can continue with the next task. If there are several instances of a task, the workflow runs asynchronously: it doesn't wait for all instances of a task to be completed before moving to the next task.

- getOutput: returns an LFN of a file on a Storage Element which contains the list of sequences as the output of submitSequence to be used for the input of the next step. For example, BLAST returns a list of hits, MUSCLE returns a list of aligned sequences, etc.

- getOutputArchive: returns an LFN of an archive of other outputs of a task including error or log files.

Because MOTEUR operates based on web services, it is very natural to import these web services into the g-INFO portal so that experts can create their workflows through a user-friendly web interface.

1.3. Portal

The g-INFO portal is developed using several web technologies: JSF 2.0 as the base framework, Ajax for a user-friendly interface and jax-ws web services to interact with the g-INFO system. The portal interacts with the g-INFO system via intermediate web services as described in Figure 1. These intermediate web services call the pre-defined web services in MOTEUR to create tasks in the TaskManager. Intermediate web services also connect with the g-INFO database to search for virus sequences or display information of user's pipelines.
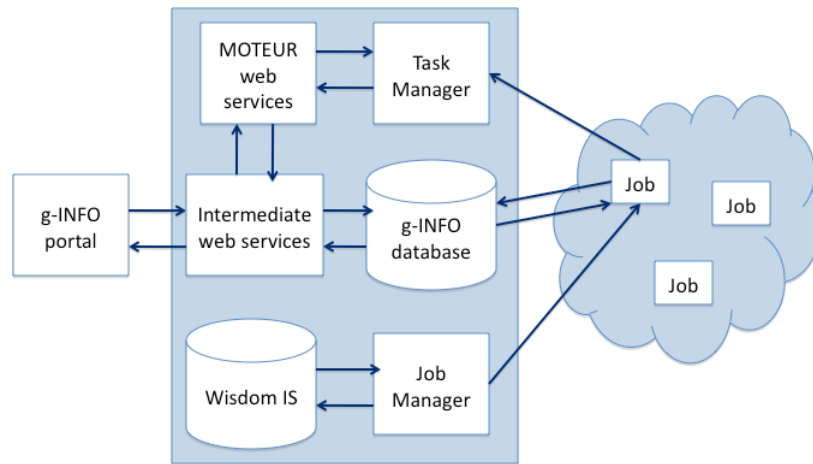
Figure 1 – Architecture of the g-INFO portal

The portal provides a simple authentication method which requires a username and password to ensure that only registered user can use grid resources provided by the portal. Every time a user accesses the portal, an authorization filter detects the area he wants to access. If the user wants to access a restricted area such as "Search" or "Manage Working Sessions", the filter redirects him to the authentication page, if he is not yet authenticated or has not the appropriate access rights. After having successfully authenticated, the user is redirected to the area he needs. User's information is stored in a session-scoped variable; so during the session the user does not have to authenticate again.

The following features of the portal will be explained in details in the next subsections:

- Search sequences imported from the public Influenza A database NCBI

- Create and manage workflow templates

- List and view status of user working sessions

- Create and run workflows in a working session

- Visualize outputs

1.3.1. Search

User provides search parameters in a query form associated with a backing bean SearchParameterBean. The form loads default parameters from the backing bean and transfers selected values to it when the user presses the "Submit" button. The backing bean uses the parameters as input parameters to call the corresponding intermediate web service. These web services process and return the result. Pagination techniques are used when results contain too many sequences. We use AJAX to quickly count the number of sequences found. If the user sees that there are too many sequences, he can add more searching parameters to reduce the search's result. Users can select a sequence to view its detailed information or select some sequences to download in FASTA format. Figure 2 illustrates a sequence diagram for the Search use case.
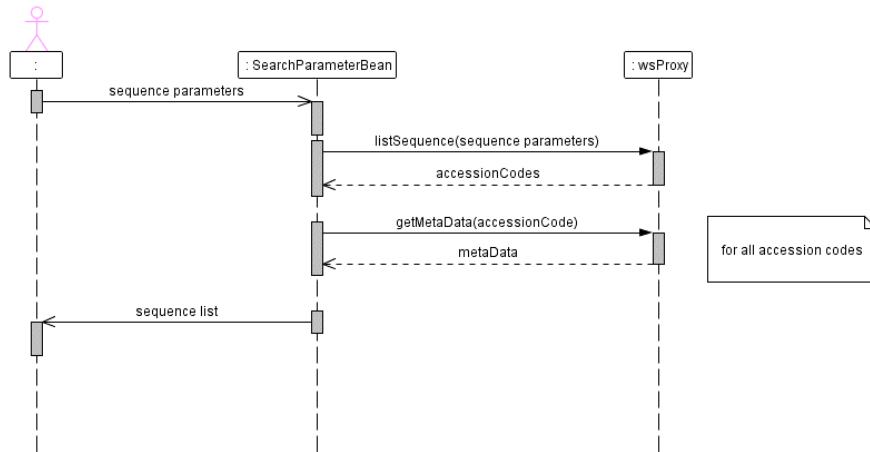
Figure 2 – Sequence diagram for 'Search' use case

1.3.2. Workflow template

Before running a new working session, a user has to select a workflow template and provide inputs for the pipelines. The portal queries in its database to find all user's predefined workflow templates and list them for selection. The user is also able to create a new workflow template. A workflow template contains a set of tools and their orders in the workflow. The user selects each tool and defines its arguments. AJAX is used for accelerating this tool selection phase. After being created, the new template is stored in the database for further use. Figure 3 displays a sequence diagram for this use case.
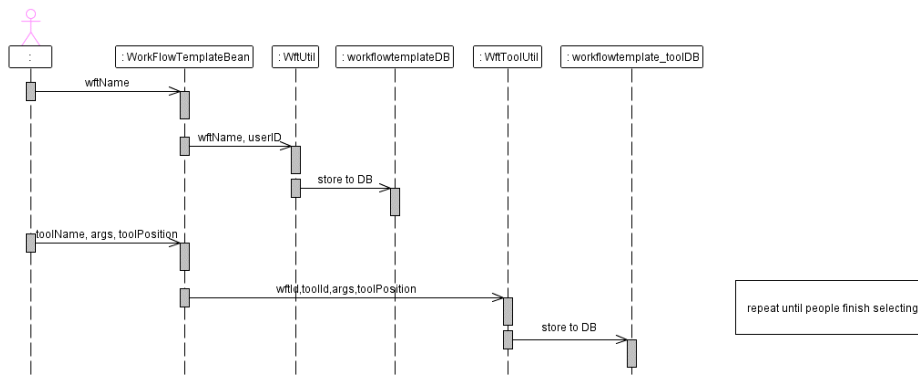

Figure 3 – Sequence diagram for 'Workflow template' use case

1.3.3. List working session

Users can view the status of their working session. Each working session contains information on its pipeline runs by one workflow template. Each pipeline contains

information of its components. Figure 4 is a sequence diagram for listing working sessions status.
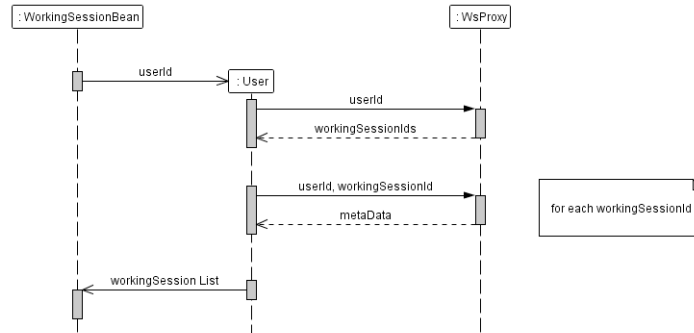

Figure 4 – Sequence diagram for 'List working session' use case

1.3.4. Run working session

After choosing a workflow template, a user needs to provide inputs to run the workflow. User can paste sequences data in a text box, or upload them from his local machine. We use an ExtensionsFilter to detect multipart/form-data requests. The filter also supports converting upload stream to String. Figure 5 shows the sequence diagram for this use case.
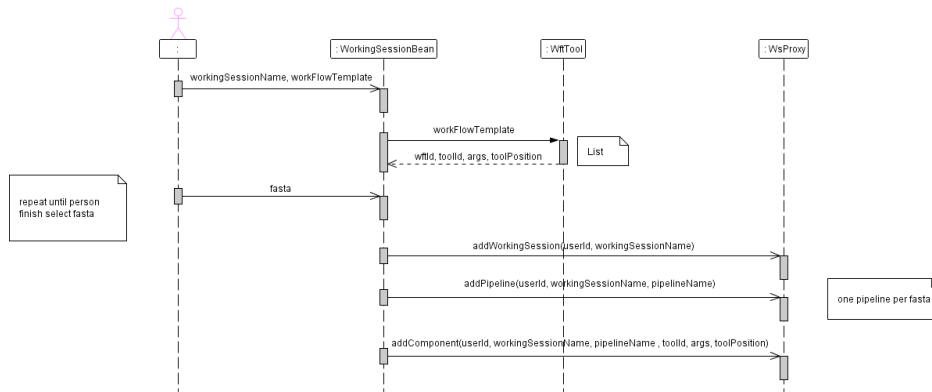

Figure 5 – Sequence diagram for 'Run working session' use case

1.3.5. Visualization

Visualization tools for phylogenetic trees have been developed widely with operations such as: *set labels, color, zoom and select* the entire branch or only nodes. Here the essential task is to a visualization tool into portals to enable specialists to manipulate the pipeline's result that is stored on the Grid. Among the available open source tools for visualization we choose PhyloWidget [13] to integrate into the g-INFO portal.

PhyloWidget is an open source program for viewing, editing, and publishing phylogenetic trees online. PhyloWidget contains a simple and powerful user interface and useful features that are not available in other phylogenetic viewers. Apart from

PhyloWidget, there exists other software such as Archaeopteryx, TreeViewJ, and BaoBab, which are widely used. Table 1 summarizes these visualization tools and their features.

Features	(1)	(2)	(3)	(4)
Support for the Newick format	y	y	y	y
Labeled, highlight nodes	y	y	y	n
Zoom	y	y	y	Y
Save as FASTAformat	n	n	n	n
Add and delete nodes or branches	y	y	n	y
Display on difference views	y	y	y	n
Easy to use the interface	y	n	n	n
Simple source code	y	n	y	y

Table1 – Summarization of visualization tools
(1) PhyloWidget, (2) Archaeopteryx, (3) TreeViewJ, (4) Baobab

Table 1 shows that PhyloWidget provides almost necessary requirements for the phylogenetic tree visualization problem. This tool runs smoothly and supports keyboard shortcuts; this helps user to manipulate trees quickly and efficiently. The operations on PhyloWidget are quite flexible and highly customizable. PhyloWidget has full support for the manipulation of Tree Edit (add children, add sisters, delete sub-trees, delete nodes), Node Edit (name, branch length, annotations), Clipboard (cut, copy, paste, swap), and Layout (swap children, flip sub-tree, re-root).

Some new features have been added to PhyloWidget so that it can work with the g-INFO portal:
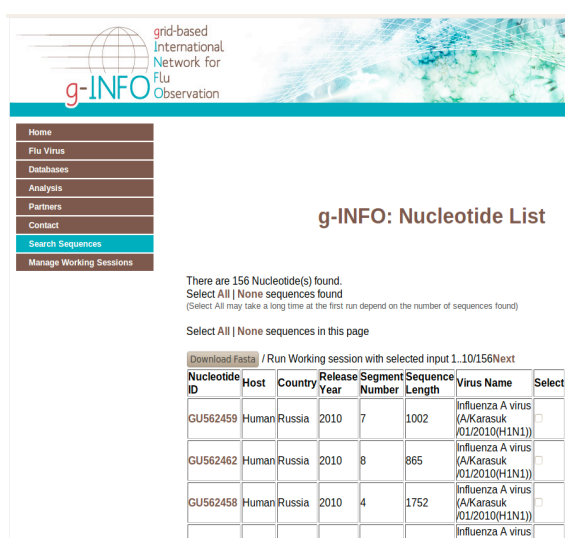
- Receive output information of a phylogenetic pipeline via intermediate web services.

- Download the output from a Storage Element on the Grid to visualize it.

- Save the edited phylogenetic tree in FASTA format. Trees in Newick format contain only sequence identifications. To get sequence data, PhyloWidget calls intermediate web services to query in the sequence database of g-INFO. This new feature helps users to utilize the output of a phylogenetic tree for other analysis

- Multi select nodes / branches on the tree to edit (coloring, removing or saving). This feature is very useful but it is missing in most of visualization tools.

2. Results and discussions

2.1. Current status of the g-INFO portal

The g-INFO portal is currently in the last phase of internal testing and will be soon switched into production mode. The current portal provides most of necessary features to help epidemiologists in monitoring Influenza A viruses.

Portal users can search sequences data with common criteria such as host, name, segment, protein, etc. Results can be downloaded to users' machines or can be used for running pipelines. Figure 6 illustrates a result page returned by a query.

Figure 6 – Searching H1N1 sequences on g-INFO portal

Portal users can define reusable workflow templates to run many instances of workflows. Workflow templates are configurable with several built-in tools and their parameters.

When running a workflow defined by a workflow template, portal users can supply many inputs at a time. Each input will create a corresponding pipeline and all the pipelines will run asynchronously on computing elements of the Grid. An example of running a workflow is shown in Figure 7. When a user clicks on the 'Finish' button, the workflow with selected workflow template will run with two inputs supplied in the previous step.

Figure 7 – Running a workflow with 2 inputs

Status of a working session including workflow parameters, instance pipelines' status, input and output are logged so that users can consult them later.

A built-in visualization tool modified from the PhyloWidget is available for visualizing the output of phylogenetic pipelines. Users can manipulate branches and nodes in the result trees, and then save the edited tree in Fasta/Newick format to their machine. Figure 8 illustrates the use of the visualization tool in the g-INFO portal.

Figure 8 – Visualization tool in g-INFO portal

2.2. Future work

The current authentication method is quite simple. We need a more reliable authentication method when switching the portal in production mode to be widely used by epidemiologists.

The visualization tool is provided only for the last result of phylogenetic pipeline herein phylogenetic trees. The intermediate phases such as Blast or alignment also need to be visualized in a step-by-step running mode that will be implemented in the future development of the portal.

This portal operates with the Grid behind the scene; consequently it makes our solution differ from others. The workflow exploits computing elements and storage elements through the Wisdom Production Environment. This makes it very easy to change the chain of bioinformatics algorithms but also to mobilize as many computing resources as needed. In the present version, the power of the grid is not yet fully exploited because parallelism is currently only implemented on the input data not on the algorithms. The cost of communication on the Grid needs to be considered if we want to parallelize the algorithms of existing tools in the g-INFO system in the future.

3. Conclusion

The g-INFO portal continues the success of the g-INFO system in terms of international collaboration. Indeed the g-INFO project is lead by Vietnamese and French researchers having a common goal: showing the relevance of grid computing to address and impact emerging health threats, particularly the flu pandemics. The portal itself is a subproject of the EUAsiaGrid project (http://www.euasiagrid.org/) and it is a work of members from 3 Vietnamese institutes that share common interest in developing grid applications.

The g-INFO portal is expected to help epidemiologists to avoid the heavy tasks of collecting all data available and analyzing them on his/her own machine. It does not try to replace the existing services made available on databases such as NCBI where experienced users can design their own workflow. It is aimed at providing a complementary service to the public health research community by producing common interest epidemiologic indicators on all available data.

The current g-INFO portal paves the way for the implementation of grids for pandemics monitoring and represents a step forward responding to the requirements of the research community relating to the federation of all the influenza data sources to avoid incompleteness and provision of tools not limited in terms of sequences' length, number of sequences, or workflow possibilities.

4. Availability and requirements

Project name: g-INFO portal

Home page: http://ginfo.ifi.refer.org:8080/

Operating systems: g-INFO system requires Scientific Linux with gLite middleware. g-INFO portal requires a Unix-based OS with Glassfish web server installed.

Programming languages: Java, C, Bash script

License: GNU-GPL

References

[1] Frederica P. Perera, I. Bernard Weinstein, Molecular epidemiology: recent advances and future directions, carcinogenesis, vol. 21, n°3, pp 517-524, 2000.

[2] Bao Y., P. Bolotov, D. Dernovoy, B. Kiryutin, L. Zaslavsky, T. Tatusova, J. Ostell, and D. Lipman. *The Influenza Virus Resource at the National Center for Biotechnology Information*, J. Virol. 2008 Jan;82(2): 596-601.

[3] Squires et al. BioHealthBase: *Informatics support in the elucidation of influenza virus host pathogen interactions and virulence*, Nucleic Acids Research (2008) vol. 36 (Database issue) pp. D497.

[4] Doan TT, Bernard A, Da-Costa AL, Bloch V, Le TH, Legre Y, Maigne L, Salzemann J, Sarramia D, Nguyen HQ, Breton V. *Grid-based International Network for Flu Observation (g-INFO)*. Studies in Health Technology and Informatics, vol. 159, pp. 215-226, 2010.

[5] Doan TT, Bernard A, Da-Costa AL, Bloch V, Le TH, Le DH, Legre Y, Maigne L, Montagnat J, Salzemann J, Sarramia D, Nguyen HQ, Breton V. *A new flexible workflow on the Grid for monitoring H5N1*. ISGC 2011 conference (accepted paper).

[6] gLite middleware, [Online], Available: http://glite.web.cern.ch/glite/default.asp

[7] V. Breton, A. L. D. Costa, P. D. Vlieger, L. Maigne, D. Sarramia, Y. Kim, D. Kim, H. Q. Nguyen, T. Solomonides, and Y. Wu, *Innovative in silico approaches to address avian flu using grid technology*, Infectious Disorders Drug Targets, Nov. 2008.

[8] T. Glatard, J. Montagnat, D. Lingrand, X. Pennec. *Flexible and efficient workflow deployement of data-intensive applications on grids with MOTEUR*, International Journal of High Performance Computing Applications (IJHPCA), 22 (3), pages 347-360, 2008.

[9] Altschul, Stephen F., Thomas L. Madden, Alejandro A. Schaffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J. Lipman (1997), *Gapped BLAST and PSI-BLAST: a new generation of protein database search programs*, Nucleic Acids Res. 25:3389-3402.

[10] Edgar, Robert C. (2004), *MUSCLE: multiple sequence alignment with high accuracy and high throughput*, Nucleic Acids Research 32(5), 1792-97.

[11] Castresana, J. *Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis*, Molecular Biology and Evolution 17 (2000), 540-552.

[12] Guindon S, Gascuel O., *A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood*, Systematic Biology. 2003 52(5): 696-704.

[13] Gregory E. Jordan, William H. Piel. *PhyloWidget: web-based visualizations for the tree of life*. Bioinformatics (Oxford, England), Vol. 24, No. 14. (15 July 2008), pp. 1641-1642.