A Provenance approach to trace scientific experiments on a grid infrastructure

Ammar Benabdelkader[1], Mark Santcroos[2], Souley Madougou[2], Antoine H.C. van Kampen[2], Silvia D. Olabarriaga[2]

[1]Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Leiden, the Netherlands

[2]Dept. Epidemiology, Biostatistics and Bioinformatics,
Academic Medical Center (AMC)
University of Amsterdam, the Netherlands

a.benabdelkader@umail.leidenuniv.nl, {m.a.santcroos | s.madougou | a.h.vankampen | s.d.olabarriaga}@amc.uva.nl

Abstract— Large experiments on distributed infrastructures become increasingly complex to manage, in particular to trace all computations that gave origin to a piece of data or an event such as an error. The work presented in this paper describes the design and implementation of an architecture to support experiment provenance and its deployment in the concrete case of a particular e-infrastructure for biosciences. The proposed solution consists of: (a) a data provenance repository to capture scientific experiments and their execution path; (b) a software tool (crawler) that gathers, classifies, links, and stores the information collected from various sources; and (c) a set of user interfaces through which the end-user can access the provenance data, interpret the results, and trace the sources of failure. The approach is based on an OPM-compliant API, PLIER, that is flexible to support future extensions and facilitates interoperability among heterogeneous application systems.

Keywords: *Scientific workflow, workflow systems, provenance, metadata, data management, e-infrastructre, distributed systems, DCI, grid computing, e-science, bioscience.*

I. INTRODUCTION

Information management is challenging in e-science due to the variety and amount of data produced daily by the physical instruments. In addition to supporting the experiment execution, it has always been crucial (or sometimes even required by regulations) for the scientific research to document experiments so that it is possible to determine how the results have been produced. In general, this documentation is done by (electronic) lab journals; however, these do not suffice for the current scale of experiments and, therefore, new approaches are required.

In this paper we describe an approach for building a knowledge base (data repository) for the scientific experiments performed using the e-infrastructure for bioscience (e-BioInfra) at the Academic Medical Center of the University of Amsterdam (AMC). The e-BioInfra platform provides grid workflow management and monitoring services for biomedical researchers that use the Dutch Grid [10] for data analysis. Our approach focuses on gathering meaningful information from these services and populating it into the knowledge base, within its proper context. The screened data generated by a grid workflow in a distributed storage environment are re-organized into a coherent and consistent data model. A software tool (eBioCrawler) retrieves, classifies, links and transforms existing data into meaningful information. The Open Provenance Model OPM [6] is adopted to represent and store scientific experiments metadata into the knowledge base using the Application Program Interface (API) from the Provenance Layer Infrastructure for e-Science Resources, (PLIER) [7]). This API incorporates an optimal database schema supported by a complete set of functions to build, store, retrieve, share, and visualize workflow experiments metadata. In addition, generic and customized tools are designed and developed to retrieve the collected information and to support the specialized queries.

The remaining of this paper is organized as follows: section 2 addresses related work and the state-of-the-art, section 3 outlines the e-BioInfra environment, section 4 describes the adopted methods and the developed tools, sections 5 and 6 evaluate the results and discuss the scientific and social impacts of the approach, and section 7 concludes the paper.

II. RELATED WORK

The recent work in the STAMPEDE project [27] has similar goals and architecture to the work described here. Metadata and logging information, however, are concepts that have been always present in most workflow-based software applications, e.g., Taverna workbench [12], Kepler/pPOD [13], and Wings/Pegasus [14]. Deelman et al. [25] describe the various levels for provenance in e-science workflows. Other approaches that addressed the provenance of data are VisTrails [30] for recording the provenance of generated images and Janus [29] for semantic provenance. The primary goal of VisTrails is the interactive multi-view visualization and mainly focuses on modules used to generate a given image without fully addressing the actual data. Janus, currently implemented within Taverna, provides semantic provenance capabilities for a limited set of queries and publishes provenance graphs as linked data. Neither of the above follows OPM specifications.

The efficient use of metadata and logging information, however, was not fully successful in many cases due to the

lack of standardization for their representation and the difficulties to integrate metadata resulting from different software applications. These concepts are now better developed and mature standards emerged in the fields of metadata and data archiving [11], data provenance [6], and information logging and bookkeeping [15]. The integration of these metadata concepts in a coherent manner and their standardization is a key solution to enable their exploitation and also facilitates the processes of reviewing, documenting, and publishing large scale scientific experiments. Therefore, since the release of the Open Provenance Model OPM in December 2007, the number of efforts addressing its implementation has grown, particularly within the provenance challenge series [24]. These implementations are very diverse, ranging from systems covering additional information such as the descriptions of virtual data sets and computational procedures [21], to systems capturing provenance information at the operating system level [22] and [23]. Lim et al. [4] describe an OPM-compliant data provenance system for scientific workflows (OPMProv), focusing on the database design and its reasoning capabilities. Another scientific workflow provenance system is described in [26]; it is designed to capture provenance data for a particular workflow system, e-BioFlow [27]. Both systems bear some similarities with PLIER API, which is part of the work presented in this paper.

III. THE ENVIRONMENT

The e-bioscience group at the AMC aims at filling the "gap" between biomedical researchers and the Dutch e-Science Grid with high-level services that support data analysis experiments on grid resources. To achieve this goal, the group designs, develops and operates the e-Infrastructure for Biosciences (e-BioInfra [8]). The e-BioInfra has originated from the VLEMED platform for medical imaging on grids, which is described in detail in [9]. It currently supports a wide range of applications, e.g. next generation sequencing, medical imaging, and different -omics experiments.

A. e-infrastructure for biosciences: e-BioInfra

As illustrated in Figure **1**, the e-BioInfra adopts a layered architecture, mainly constituted of the following components (bottom-up):

- The Dutch *Grid fabric* and services for distributed storage and computation [10]. All clusters run Scientific Linux and gLite grid middleware, being part of EGI. Services include a gLite Logical File Catalog (LFC), a Virtual Organization Membership Service (VOMS) for authorization using X509 certificates, and computing and storage elements (CE, SE).
- *Generic services to access the grid resources*. (e.g. workflow management, monitoring and data transfer) from various front-ends, see more details below.
- Web-based *User interfaces* for specific applications that execute customized grid workflows, and a custom Java-based front-end (Virtual Resource Browser [31]) for file manipulation and execution of MOTEUR workflows.

The following tools and technologies are used; see more details in [9] and [19]:

- Workflows are described in the GWENDIA language [1]. This consists of an abstract description that is instantiated at run time with input data. The execution of a workflow with data is called an *experiment*.
- The MOTEUR workflow engine [2] is used to execute workflows on grids. The workflow components are Linux executables wrapped on-the-fly into grid jobs by the Generic Application Service Wrapper (GASW) [32]. MOTEUR can interface with various back-end systems to run workflows in various infrastructures.
- DIANE Distributed Analysis Environment [3] provides a pilot job framework that is used as back-end for MOTEUR to execute *workflow tasks* as *grid jobs*.

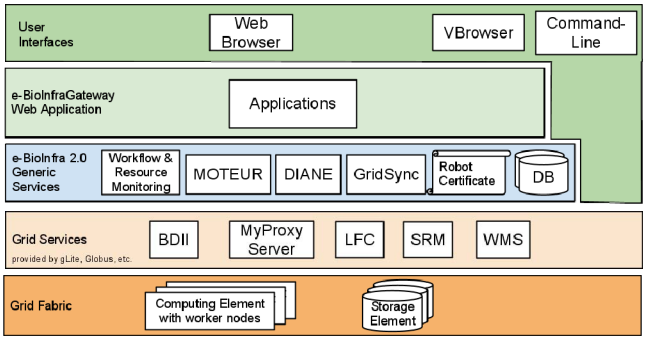

Figure 1. *Layered architecture of the e-BioInfra*

B. Data/Information management problem

The deployment of the rich e-BioInfra platform facilitated researchers to perform large grid experiments. As a consequence the complexity and the amount of (intermediate) results generated by experiments have also significantly increased. Not only the execution of the scientific experiments has become more complex, because they involve more components, computation, and data, but also the data management for such large and rich datasets has become more challenging. The data/information management is particularly challenging because the operating environment is composed of heterogeneous third-party components, each with its own custom logging system. Moreover, it is not trivial to correlate pieces of data; in particular tracing the origin of events requires much effort and detailed knowledge of these systems. All together, it is currently difficult to manually validate experiments performed in the e-BioInfra, as well as to trace back the source of failure that can occur at the various levels of distributed components. Furthermore, there is no coherent approach to further document the experiments and to exploit the acquired expertise in a productive way.

During routine operation by administrators and usage by biomedical researchers, there is a strong need to:

(a) Keep tracking information for performed experiments;

(b) Interpret the results, diagnose the problematic cases and trace the source(s) of failure;
(c) Support the reproducibility of scientific discovery and be able to review, document, and publish experiments;
(d) Expand and disseminate the acquired expertise in experiment design and troubleshooting to other groups;
(e) Exploit the acquired expertise to better re-design scientific experiments and avoid sources of failure.

Manual data collection is an error-prone task and requires enormous manpower, due to the amount of logs registered by the various processes in the distributed resources. A key solution in this case is to record and gain access to a complete set of metadata describing the derivation history of scientific experiments and their results. A comprehensive knowledge base that gathers and annotates relevant information would help scientists clarify research questions and validate their operational tasks.

IV. METHODS

Building and populating a knowledge base about grid experiments, with proper and detailed information resulting from different sources, is a challenge in itself. An ideal solution would be to design and build a single and coherent system consisting of a number of well integrated set of databases, workflow systems, and applications services. This is generally not the situation for grid infrastructures as the e-BioInfra, where a loosely coupled architecture is adopted by design. Instead, we collect metadata about the results generated by workflow activity in the various components, and link them all together by adopting a structured and semantically rich data provenance framework.

Our approach consists of a comprehensive solution including the design and implementation of a knowledge base to store and instrument scientific experiments supported by a set of services and tools to gather, classify, and make efficient use of the collected metadata. We followed a structured approach for the data provenance collection starting from the abstract workflow description, which guides data collection at the various systems and execution levels. This also provides context for linking the collected data and reconstructing the concrete workflow execution information. This approach allowed us to control the amount of data to collect and to enforce the data quality by maintaining knowledge about the workflow experiment context and environment.

Figure 2 shows the designed architecture with three main components to support storage, population, and access to the provenance data. The *provenance repository* complies with the notion of graphs outlined by the OPM model. The *eBioCrawler* automatically gathers provenance data from existing logs that contain information generated by different resources (e.g. MOTEUR). The *browsing tools* implement a generic provenance query tool and a web-based browsing interface for the end-user to access the information about performed experiments, interpret the results, and trace the sources of failure. In the following sections, we describe in more details these components and their implementation.

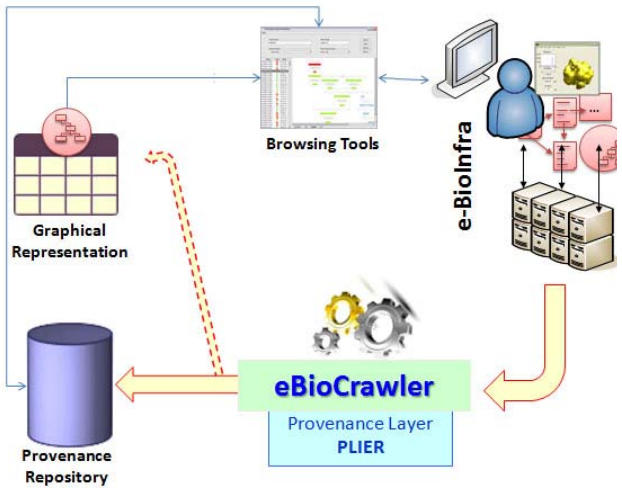

Figure 2*: Components of the adopted architecture*

A. *Provenance repository*

The provenance repository is implemented using a relational database schema that captures the concepts of the OPM model. OPM was adopted as base to represent scientific experiments and their execution path because it is flexible to support future extensions, as well as it promotes and facilitates interoperability among heterogeneous systems. Figure 3 illustrates the Entity-Relationship diagram of the underlying database schema.

The provenance data is collected and stored into the repository using the PLIER API [7], which allows developers to build, store and share workflow experiments as OPM-compliant data objects. PLIER implements an optimal relational database schema, using most recent standards and mechanisms, namely the Java Persistence API (JPA 2.0) and Hibernate [17]. It also provides specific interfaces (JDO 3.1 [16]) to transform or serialize the provenance data into specific formats (e.g. RDF, XML, and DOT). In this implementation the PLIER API is deployed within a stand-alone application (e-BioCrawler), which controls the amount and quality of data collected. Another use of the API is to integrate it within the workflow management system as a plug-in component that can be enabled for provenance data collection, by the application user.

B. *eBioCrawler*

The eBioCrawler is a java-based tool that screens the logs files, classifies and links the data, and stores it into the provenance repository. Three main functions are provided by the eBioCrawler: data collection, statistics calculation, and generation of graphical representation.

Data collection is based on log and description files that are available for each experiment/workflow on the e-BioInfra. These logs include status and history information

generated by the various distributed components, for example, workflow descriptions (in GWENDIA), MOTEUR and DIANE execution and status reports, standard system outputs, user information, and execution of jobs. The collected information is stored into the knowledge base using the mappings illustrated in TABLE I. Each concept in the abstract design and concrete execution of workflow experiments is mapped into a specific OPM concept. Although event-driven mechanisms are common in scientific workflows, OPM does not clearly define a corresponding concept for them. The e-BioInfra, however, requires capturing the occurrence of events during the execution of a workflow experiment (e.g. timestamp when a process is queued, running, and completed or failed). To cope with that, eBioCrawler adopts the OPM Annotation concept to represent the occurrence of events during the different processing stages.

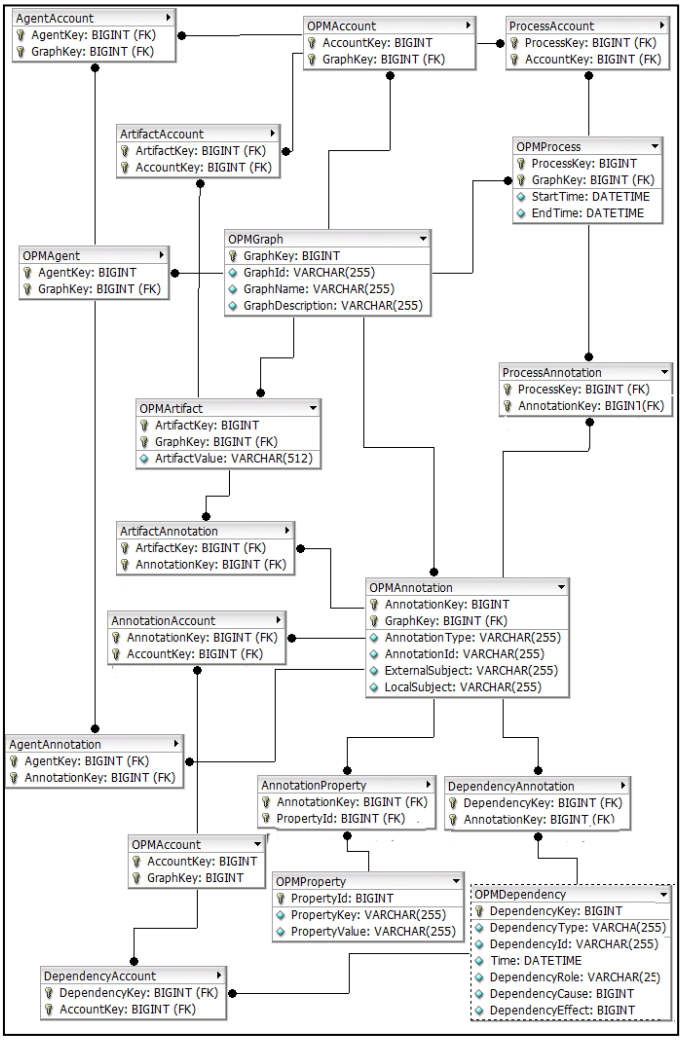

Figure 3: *ER database schema of the provenance repository*

To add value to the collected data, additional computations are performed to provide high-level summary information and statistics about each experiment. Examples are final status of a workflow (successful or failed), experiment and job duration, and count of completed, failed, and retried jobs. Execution and job queuing times are also calculated by extracting and computing the timestamp events of each job from the database. Such computations require reasoning mechanisms to infer the data dependency among collected information. For example, it is not obvious to detect the final status of an experiment from job logs, because failing jobs are re-submitted until they succeed to perform a workflow task or exhaust the number of retries. As an example, an experiment composed of 5 tasks could finish successfully after the submission of 10 jobs. In such a case, the eBioCrawler is capable of determining the succeeded, failing, and retried jobs, as well as the final status of the experiment ("successful" if 5 jobs succeeded).

TABLE I. WORKFLOW-OPM CONCEPTS MAPPING

Workflow concept	***OPM Concept***
Users, host machines	AGENT
Jobs	Process
Input/output files	Artifact
Parameters	Artifact
Data dependencies	Causal Dependency
Properties	Property
Anthologies	Annotation
Events	Annotation/Property

Figure 4 illustrates an abstract definition of a workflow experiment, drawn as an OPM graph. A high-level view of the run-time execution of that experiment is depicted on Figure 5, in which the eBioCrawler encodes data in the graph as visual properties (shape and color). By zooming into the experiment graph (Figure 6), the user could observe that the fourth job has failed and then succeeded after re-submission.

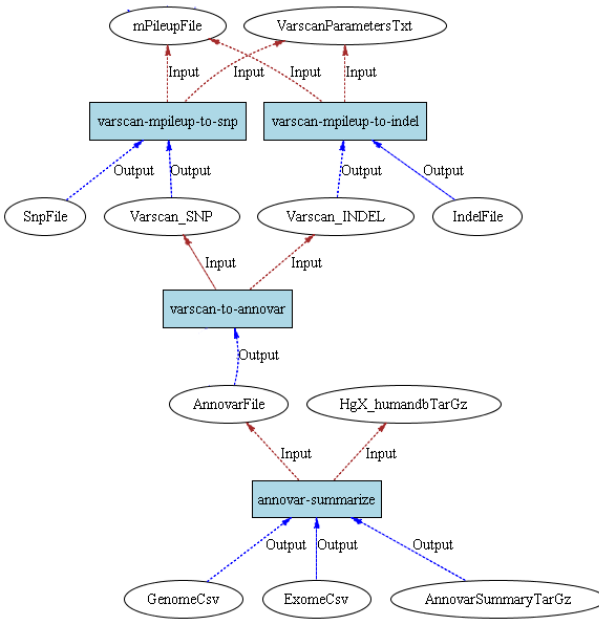

Figure 4*: Workflow experiment: Abstract graph (oval=data, box =process)*

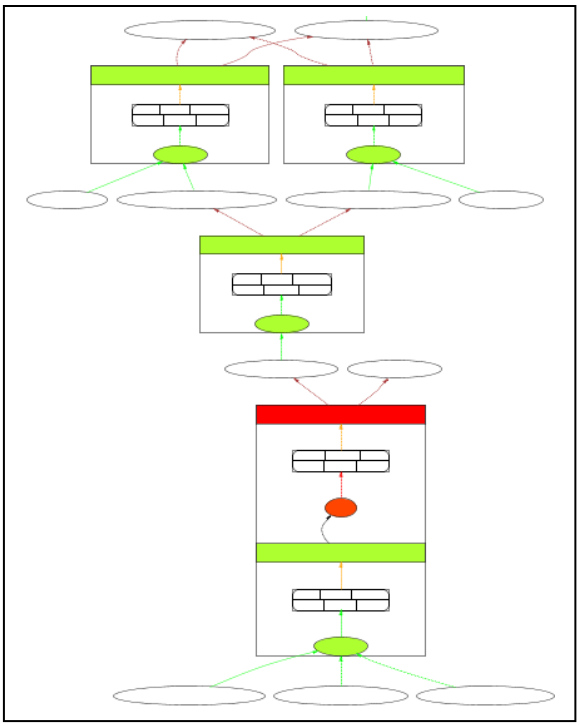

Figure 5*: Workflow experiment: Concrete Graph. Colors indicate status (red=failed, green=successful), and shape indicates type of component (input, processes, events and results)*

In addition, detailed inputs and outputs are linked to their processes, and timestamp information is shown for the different execution statuses of a job (queued, running, completed, or error).

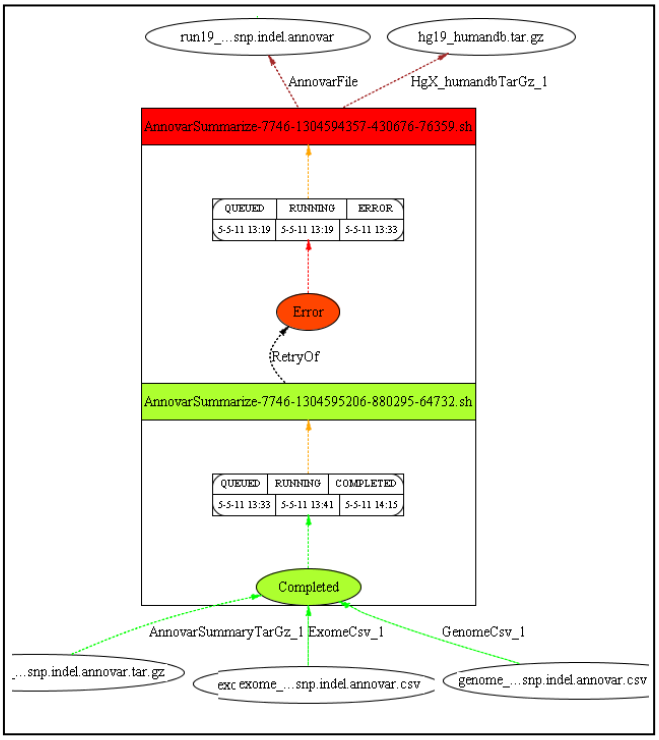

Figure 6*: Workflow experiment: Zoom on a concrete graph.*

The third function of the eBioCrawler generates additional graphical representation(s) used in the web interface to represent the experiment execution visually and to facilitate user interaction with its components. Figures 4, 5, and 6 illustrate the generated graphical representation of an experiment.

The graphics are generated with third-party software graphviz [18], which consists of producing a) the GIF image corresponding to the experiment graph, and b) the map file defining the coordinates of the graph components. These two files are processed by the eBioCrawler and merged within an HTML output with clickable items.

The HTML format represents a high-level view of an experiment designed and implemented to suit the needs of an e-BioInfra administrator. The view includes on-mouse-over highlighted links and clickable components:

- Input/output artifacts link to the exact remote location of the files (e.g. HTTP, LFN, and SRM);
- Succeeded jobs (highlighted in green) link to the standard output of each specific process;
- Failed jobs (highlighted in red) link to the standard error of each failing process;
- Events for each process are added to the graph with their exact time of occurrence
- Links between the input/output artifacts and their corresponding processes are properly labeled.

This representation facilitates the analysis of the provenance data by a user; it is particularly valuable for tracing causes of errors.

C. *Browsing interface*

This component implements a generic provenance query interface illustrated in Figure 7, and a web-based interface (workflow dashboard). Through them the end-user can access the information about the performed experiments, interpret the results, and trace potential sources of failure. Both interfaces use the provenance repository as a back-end to query and search metadata about the experiments and to provide the abstract representation of the experiment next to its run-time execution in a graphical view.

In addition, the workflow dashboard combines the power of the database querying functionalities and the rich graphical representation of experiments generated by the eBioCrawler to implement the following functionalities:

- Full search and exploration facility using the data repository. The search is based on a combination of keywords (e.g. scientist, status, name, coverage, date);
- High-level information such as summary of experiment, its final status, execution time, and generated results.
- Statistical information based on the status of experiments, the executed application, or the user;
- Pre-defined queries that could be formulated by the scientist (e.g. most recent successful or failing experiments; most longest or shortest experiments);

- Aids to trace the sources of failure by following the graphical representation of the experiment execution and its clickable components, using the graphical HTML/GIF/MAP formats described in section IV.

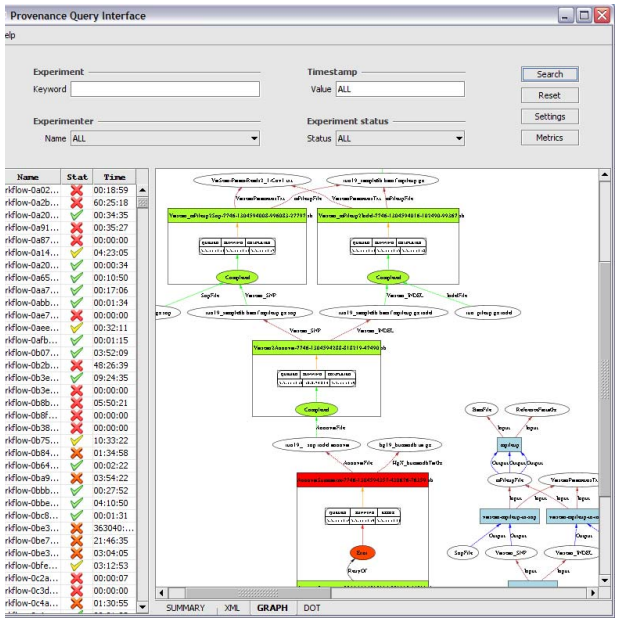

Figure 7*: Provenance Query Interface*

V. EVALUATION

To validate the implementation and explore the potential of our approach in real situations, we ran the eBioCrawler on the real logs of the e-BioInfra server containing data for 3194 experiments/workflows executed on the Dutch Grid in the period from December 2010 to June 2011. The workflows were executed by 10 users and covers in total 104 data analysis programs implementing variations of a smaller number of applications in medical imaging and DNA sequencing.

In the current prototype implementation the eBioCrawler, the data repository and e-BioInfra services run on separate servers. For each workflow, around 12 log and description files from the e-BioInfra had to be fetched and processed by eBioCrawler. The provenance repository was populated from scratch several times to get an impression of its performance under varied network and services conditions.

The total data collection took in average around 4.5 hours, ranging from 250ms (smallest experiment) to 15s (largest experiment) per workflow. This includes data collection, linking process, storage in the repository, calculation of summary statistics and generation of the graphical representation. The amount of metadata collected per workflow is expressed by the number of processes, input/output artifacts, their corresponding annotations, and the links between the different components of the experiment. The generated provenance database for all 3194 experiments has 188 MB in size, which is reflected by the total number of the experiments it stores.

To have an impression of the potential of this approach, we used the knowledge base to obtain statistics about the activity performed via the e-BioInfra gateway. TABLE II. summarizes statistics at the experiment or workflow level, including the total number of performed and failed experiments, number of generated outputs (result files), applications, execution CPU time, and experiment failure ratio (number of workflows that did not complete successful / total number of workflows). Similar statistics are provided per user performing scientific experiments.

TABLE II. EBIOINFRA WORFLOW EXPERIMENTS SUMMARIES

User	**# exp.**	**# failure**	***# outputs***	***# Applicat.***	***exec. time cpu (days)***	***failure ratio***
User 1	894	178	*763*	*3*	*284*	*20%*
User 2	696	371	*6040*	*7*	*1857*	*53%*
User 3	522	286	9873	*55*	*1949*	*55%.*
User 4	485	254	*18476*	*21*	*911*	*52%*
User 5	180	85	*2212*	*17*	*67*	*47%*
User 6	143	57	*6325*	*17*	*004*	*40%*
User 7	127	89	*203*	*8*	*097*	*70%*
User 8	97	69	422	*3*	*181*	*71%*
User 9	23	18	329	*3*	*012*	*78%.*
User 10	27	17	*155*	*4*	*020*	*63%*
Total	**3194**	**1424**	***44799***	***104***	***5382***	***45 %***

TABLE III. illustrates summary statistics for grid jobs for all experiments, including the total number of completed, failed, and retried jobs; queuing and execution time (in days), and the job failure ratio. Similarly, statistics are also computed per user. The statistics presented here were obtained in a straightforward manner by issuing direct SQL queries against the knowledge base.

TABLE III. E-BIOINFRA JOB STATISTICS

User	**# jobs**	***# completed***	***# failed***	**# retried**	***queue time***	***exec time***	***failure ratio***
User 1	1,209	*769*	*440*	256	*1*	*283*	*36.39%*
User 2	7,728	*3,562*	*4,166*	3,346	*172*	*1,685*	*53.91%*
User 3	11,978	2,568	*9,410*	5,009	*195*	*1,753*	*78.56%*
User 4	9,820	*2,121*	*7,699*	3,168	*119*	*792*	*78.40%*
User 5	1,551	*788*	*763*	522	*7*	*60*	*49.19%*
User 6	850	*428*	*422*	68	*1*	*002*	*49.65%*
User 7	793	*173*	*620*	509	*2*	*095*	*48.18%*
User 8	1,074	290	*784*	642	*5*	*177*	*63.00%*
User 9	969	282	*687*	391	*6*	*006*	*70.90%*
User 10	262	*059*	*203*	159	*2*	*019*	*66.73%*
Total	**36234**	***11040***	***25,194***	**14070**	***509***	***4873***	***69.53%***

VI. DISCUSSION

Respective to performance, we found that the response time of the eBioCrawler (up to 15s per workflow) is acceptable for batch processing, but less tolerable in an interactive setting. Database access during the data crawling process has performed well for the volume of data and transactions so far (3194 workflows, 36234 jobs, 44799 outputs). However, if performance limitations are faced, we could either improve the hardware resources of MySQL, or consider another DBMS when we reach a performance bottleneck in the future. Note that the size of the database will remain rather small because it only contains pointers to the real files. Nevertheless, it is required to ensure the persistency of links to data files, which is not straightforward when a flat file system is used and users have complete freedom to relocate or delete files, which is the case in the current set-up.

Regarding the provenance information correctness, we identified difficulties related to data representation and consistency. Firstly, some of the log files to be processed were missing (e.g. the standard output of MOTEUR execution) or mal-formatted (e.g. XML description files and date/time format). This has led to incomplete data representation in some of the experiments. While debugging, it turned out that these experiments were abandoned or stopped by the user, and therefore the logging information was left at an incomplete state. Secondly, for some complex workflows involving a large combination of input and output data it was not possible to properly link all the output files to the exact processes generating them. In these cases some of the generated graphs contained lose components, but they are still correct otherwise.

Our approach is potentially valuable to a wide audience, from researchers to system administrators. Because of the wide and distributed scale of the resources, "the grid" often becomes a black box to users: a) jobs may get scheduled arbitrarily on distributed resources, b) log files will be generated on the remote nodes selected by the job scheduler, c) the user may lose access to the remote nodes where those files reside, and d) the log files may get cleaned without prior notification. In addition, with the increased volume of data, resources, methods and collaborators in e-Science endeavors, it becomes increasingly difficult to perform repeatable, scalable, and traceable experiments. Our approach collects the distributed information in a structured way that can be queried for various goals like verifying results, operational statistics, tracing errors, customized reporting, etc. This greatly increases e-BioInfra usability, with potential impact in the scientific and operational dimensions.

By tracing the whole history of the resources up to the current state, it is possible to confirm or provide evidence of the scientific workflows. Analysis and comparative mechanisms, scientific opinions or interpretations, and the results of various kinds of examinations may provide further ways to improve (or debug) the actual scientific experiment (workflow).

As an example of operational impact, let's focus on the final status of an experiment, which is further enhanced to also cover *partial* success or failure. Such distinction in status is motivated by the fact that 1) an experiment that completed successfully may include jobs that were re-submitted after failure; similarly, 2) an experiment that failed may include jobs that succeeded. Experiments that succeeded after re-trial (of jobs) could help, for example, in identifying "friendship" between the hardware resources and the type of jobs, i.e., some hardware resources are better suited for a type of jobs and software. Sources of failure within experiments achieving partial success can be more easily traced, debugged, and solved; those experiments could be re-submitted after being adapted or corrected. In addition, output results generated by successful jobs (within the failed experiments) and stored on the grid can also be traced and easily removed from the permanent storage. Finally, the analysis of the provenance data and resources involved in an experiment may present new insights about the e-BioInfra usage and potential optimization, for example, for data retention strategies, assistance for workflow debugging, etc.

VII. CONCLUSIONS AND FUTURE WORK

In this paper we described how a knowledge base containing provenance of experiments performed with the e-BioInfra has been achieved. The proposed solution implements an OPM-compliant data model using a relational database management system, and provides the necessary set of tools to gather, store, link and access the knowledge base. The rich and complex information sources available within the e-BioInfra platform provided excellent material to test and validate our approach. These sources include logs and descriptions files for few thousands of scientific experiments performed using MOTEUR/DIANE workflow management service between December 2010 and June 2011. Provenance data for the e-BioInfra is collected automatically from the various components and transformed into a knowledge base repository that can be accessed for analysis of the experiment provenance in various forms (e.g. GUI for provenance graphs). Specific software tools are available to better explore the information in the knowledge base, such as the workflow dashboard presented here. Each tool can be tailored to the specific needs of the application domain and the type of users.

In our initial evaluation, we demonstrated that the challenging data management strategies within the e-BioInfra could be improved with this approach by enabling users to:

- Distinguish between the different execution statuses of experiments, and therefore, have full information about their experiments and easily trace the source of errors.
- Gather the sources for error occurrence and explore them to improve the design of failing experiments and new ones.

- Deploy the statistical information provided by the workflow dashboard (or extracted directly from the knowledge base) to get a clear insight about the e-BioInfra usage and the potentials for optimization.

In the near future the solution presented here will be put into production at the new e-BioInfra release. This will require refinement of the user interfaces, and possibly also additional queries to obtain information of relevance to the biomedical researchers. We will also attempt to integrate parts of the eBioCrawler functionalities into the workflow service itself. Since this is an invasive operation, it will be performed in close collaboration with the MOTEUR project. Additional future work include (a) enhancing the knowledge base with support for Dublin Core metadata elements to better enhance the processes of documenting, reviewing, and publishing the experiment results; and (b) enhancing the data management toolbox with additional components to improve the search for system administrators (e-BioInfra support) and to address the needs of other groups in the area of life sciences.

ACKNOWLEDGMENT

The authors would like to thank the contribution of the former developer of the PLIER API (Victor Guevara), the other members of the e-bioscience group of the AMC who develop and maintain the e-BioInfra, and the AMC researchers who run grid workflows with the e-BioInfra to analyze biomedical data. This work was carried out in the context of the BiG Grid project, with financial support from the Netherlands Organization for Scientific Research (NWO).

REFERENCES

[1] J. Montagnat, B. Isnard, T. Glatard, K. Maheshawari and M. Blay Fornarino, “A data-driven workflow languange for grids based on array programming principles,” Proccedings of WORKS 20’09, Portland (Oregon/USA), November 2009.

[2] T. Glatard, J. Montagnat, D. Lingrand and X. Pennec, “Flexible and efficient workflow deployement of data-intensive applications on grids with MOTEUR,” J. High Performance Computing Applications, vol.22(3) pp.347-360, August 2008.

[3] J.T. Moscicki, “DIANE - Distributed Analysis Environment for GRID-enabled Simulation and Analysis of Physics Data,” NSS IEEE 2003, Portland (Oregon/USA), October 2003.

[4] C. Lim, S. Lu, A. Chebotko and F. Fotouhi, “Storing, reasoning, and querying OPM-compliant scientific workflow provenance using relational databases,” Future Generation Computer Systems, vol. 27 pp.781–789, October 2010.

[5] A. Chebotko, S. Lu, X. Fei and F. Fotouhi, "RDFProv: A Relational RDF Store for Querying and Managing Scientific Workflow Provenance", *Data & Knowledge Engineering (DKE)*, vol.69(8), pp. 836-865, August 2010 (Elsevier Science).

[6] L. Moreau, et al, “The Open Provenance Model Core Specification (v1.1*),”* Future Generation Computer Systems, vol.27(6) pp.743-756, June 2011.

[7] PLIER: http://twiki.ipaw.info/bin/view/OPM/Plier

[8] ebioInfra: http://www.ebioscience.amc.nl/

[9] S.D. Olabarriaga, T. Glatard and P.T. De Boer, “A virtual laboratory for medical image analysis,” IEEE Trans Inf Technol Biomedicine, vol.14(4) pp.979-85, April 2010.

[10] The Dutch Grid for e-science: www.biggrid.nl

[11] Dublin Core: http://www.dublincore.org/

[12] P. Missier, et al, “Seamless Provenance Representation and Use in Collaborative Science Scenarios (Abstract),” AGU Fall Meeting, San Francisco (CA/USA), Fall 2010.

[13] S. Bowers, T. McPhillips, S. Riddle, M.K. Anand and B. Ludäscher, “Kepler/pPOD: Scientific Workflow and Provenance Support for Assembling the Tree of Life,” IPAW’08, pp.70-77, 2008.

[14] J. Kim, E. Deelman, Y. Gil, G. Mehta and V. Ratnakar, “Provenance Trails in the Wings/Pegasus Workflow System,” Concurrency and Computation: Practice Experience, vol.20(3) pp.587-597, April 2008.

[15] L. Matyska, et al., “Job Tracking on a Grid – the Logging and Bookkeeping and Job Provenance Services,” CESNET Technical Report No 9/2007, 2007.

[16] D. Jordan, C. Russell, “Java Data Objects (1st ed.)” O'Reilly Media. pp. 384. ISBN 0596002769, 2003.

[17] G. King, C. Bauer, *“Java Persistence with Hibernate* (Second ed.), “ Manning Publications, pp. 880, ISBN 1932394885, November 2006.

[18] J. Ellson, E. Gansner, L. Koutsofios, S.C. North and G. Woodhull, “Graphviz – open source graph drawing tools,” LNCS 2002, vol.2265 pp.594-597, 2002. Web site: http://www.graphviz.org

[19] S. Shahand, M. Santcroos, Y. Mohammed, V. Korkhov, A. C. M. Luyf, A. van Kampen and S.D. Olabarriaga, “Front-ends to Biomedical Data Analysis on Grids,” Proceedings of the HealthGrid Conference, Bristol(UK), June 2011.

[20] D.R. Harvey and P. Hider, “Organising Knowledge in a Global Society” Wagga Wagga NSW: Charles Sturt University..

[21] B. Clifford, I. Foster, J.-S. Voeckler, M. Wilde1 and Y. Zhao1, “Tracking Provenance in a Virtual Data Grid” Concurrency and Computation: Practice Experience, vol.20(5) pp.565-575, April 2008

[22] D.A. Holland, M. Seltzer, U. Braun and K.-K. Muniswamy-Reddy, “ PASSing the provenance challenge” Concurrency and Computation: Practice and Experience, vol.20(5) pp.531-540, April 2008.

[23] J. Frew, D. Metzger and P. Slaughter, “Automatic capture and reconstruction of computational provenance,” Concurrency and Computation: Practice and Experience, Special Issue on the First Provenance Challenge, vol. 20 (5) pp.485-496, April 2008.

[24] Provenance Challenge: http://twiki.ipaw.info/bin/view/Challenge.

[25] E. Deelman, D. Gannon, M. Shields and I. Taylor, “Workflows and e-Science: An overview of workflow system features and capabilities”, Future Gen Comp Syst, vol. 25 (5) pp. 528-540, 2008.

[26] M.J. Ooms, "Provenance Management in Practice," Master's thesis, University of Twente, NL, 2009.

[27] I. Wassink, H. Rauwerda, P. van der Vet, T. Breit and A. Nijholt, “E-BioFlow: Different Perspectives on Scientific Workflows,” Communic Comput Information Science, v.13, pp. 243-257, 2008.

[28] Synthesized Tools for Archiving, Monitoring Performance and Enhanced DEbugging (STAMPEDE) Project: https://confluence.pegasus.isi.edu/display/stampede/Home

[29] P. Missier, J. Zhao, S. S. Sahoo, C. Goble and A. Sheth, “Janus: from Workflows to Semantic Provenance and Linked Open Data”, IPAW’ 10 Troy, NY June 15-16, 2010.

[30] L. Bavoil, S. P. Callahan, Patricia J. Crossno, J. Freire, C. E. Scheidegger, C. T. Silva and H. T. Voet al., “VisTrails: Enabling interactive multiple-view visualizations”, IEEE Visualization, pp. 135-142, Mineapolis , USA 2005.

[31] S.D. Olabarriaga, P.T. de Boer, K. Maheshwari, A. Belloum, J.G. Snel, A.J. Nederveen, M. Bouwhuis, "Virtual Lab for fMRI: Bridging the Usability Gap", 2nd IEEE Conference on e-Science and Grid Computing, Amsterdam, December 2006.

[32] Generic Application Service Wrapper (GASW): http://modalis.polytech.unice.fr/softwares/moteur/gasw

[33] T. Glatard and S.D. Olabarriaga. "User friendly management of workflow results: from provenance information to grid logical file names," 4th IEEE International Conference on e-Science (eScience'08*)*, Indianapolis, December 2008.