A Knowledge Discovery Service System for Provenance Exploration

Qing Liu[a], Ji Zhang[b], James Forbes[a], Kai Xu[c], Dinesh Nair[d]

[a] *CSIRO ICT Centre, Australia*
[b] *The University of Southern Queensland, Australia*
[c] *Middlesex University, UK*
[d] *GLiNTECH, Australia*

Abstract

Scientific research has moved from an isolated environment into a collaborated culture due to the data explosion and the experiment complexity. Several scientific workflow systems have been developed to support scientists to conduct the scientific analysis and create knowledge. They provide the ability to automatically record provenance, the process that led to a particular data, which can be used by other scientists for better understanding, re-using and verifying a data product to generate new knowledge. However, most of the existing scientific workflow systems provide poor query capability for users to effectively find relevant provenance information. It is not a trivial task due to the complexity of the workflow and the size of the provenance graphs. In this work, we propose a system architecture for provenance exploration in which querying, navigation and visualization methods are combined together. It enables effective and efficient knowledge discovery among various provenance information generated by different workflow runs.

Keywords: system architecture, workflow, knowledge discovery, provenance

1. Introduction

We are experiencing data explosion over the past 10 years. In the bioinformatics area, with the success of human genome sequencing, there are huge amounts of genomic data available in public sources. On the other hand, the emergence of Web technologies has seen a significant number of available bioinformatics

Email addresses: `Q.Liu@csiro.au` (Qing Liu), `Ji.Zhang@usq.edu.au` (Ji Zhang), `James.Forbes@csiro.au` (James Forbes), `kxkevin@gmail.com` (Kai Xu), `dinesh.nair@glintech.com.au` (Dinesh Nair)

web resources. According to the latest survey, there are 1078 biological databases (Galperin, 2007) and over 1200 bioinformatics tools (Brazas et al., 2008) publicly available online. Scientific research has shifted from its pure lab environment to a paradigm that Web-based analysis methods and data resources could be used to achieve the same objectives more efficiently and effectively at a large scale. Complex computation process to analyze data is now becoming possible. Analysis of these data using computational methods, the so called *in-silico* experiments, is becoming an integral part of modern biological studies.

There are many scientific workflow systems have been developed to support scientists to conduct the scientific analysis and create knowledge. All the scientific workflow systems have in common that they provide graphical user interface to let scientists compose workflow / analysis and support workflow execution. In this context, provenance means the data and process dependencies introduced during workflow runs and the meta-data of the workflow such as the workflow description, the licence applied, the annotation attached to the workflow etc.

Most of the workflow systems provide the capability to automatically record provenance which can be used by other scientists for better understanding, re-using and verifying a data product to generate new knowledge.

Taverna (Oinn et al., 2004) workflow system has been widely used by the bioinformatics community. The user can plug in the ”LogBook” which tracks all the provenance information during the workflow execution. Then the workflow process can be shared using *myExperiment* platform. However, the query ability of myExperiment is very limited.

The Wings/Pegasus (Kim et al., 2006; Deelman et al., 2005; Gil et al., 2007) system introduces the notion of reusable workflow template that is instantiated into a workflow instance, containing execution details.

Karma provenance management (Simmhan et al., 2008) provides a means to collect workflow, process and data provenance from data-driven scientific workflows. It relies on a notion of nested workflows, which allows provenance to be grouped according to its depth. It uses an incremental, building-block method to construct provenance queries based on the its provenance model provided by the Karma service.

Trident (Barga et al., 2008) is a scientific workflow workbench built on top of the commercial workflow enactment engine Windows Workflow. It has been mainly applied and demonstrated in the field of oceanography

Kepler (Altintas et al., 2004) and Vistrails (Callahan et al., 2006) records the workflow evolution in the workflow specification made by the user. Vistrails plugin for ParaView incorporates the provenance management capabilities of VisTrails into ParaView. All of the actions a user performs while building and modifying a pipeline in ParaView are captured by the plugin. This allows navigation of all of the pipeline versions that have previously been explored.

All the above workflow systems have developed their own provenance models to capture provenance information. The Open Provenance Model (Moreau et al., 2008) is an approach that consists of controlled vocabulary, serialization formats and APIs (Application Programming Interfaces) that allow provenance from individual systems to be expressed, connected in a coherent fashion, and queried seamlessly. However, identifying equivalent OPM features among workflow runs of different scientific workflow systems is often a difficult task (Cruz et al., 2009).

To fully facilitate scientific collaboration, it is essential for users to discover the relevant provenance information effectively and efficiently. Although Vistrials is able to capture the workflow evolutions made by users, most of the existing workflow systems do not provide methods to explore the relationships among various workflows. The provenance information is not fully utilized due to the poor query ability. This is not a trivial task. Given the large amount of provenance information generated by different analysis and the complexity of the provenance graph, retrieving the relationships between multiple workflows has very high computational complexity and is time consuming. The technical challenge lies in how to effectively use, manage and present the provenance information to the user.

In this work, we propose a service system architecture for provenance exploration in which querying, navigation and visualization methods are combined together. It enables effective and efficient knowledge discovery based on the semantic context of the workflow and the similarity among various provenance information generated by different workflow runs. In particular, the system can help users to locate relevant information via exploration and advanced query methods. The goal is achieved by (1) integrating provenance recording functions tightly with workflow construction and execution, (2) developing an graphical user interface for provenance navigation, and (3) proposing strong query ability and presenting the query results in an informative visual fashion.

Compared with the existing systems, the distinct contributions of our system are in the following aspects:

- **System architecture**: We apply a system level approach to fully support provenance exploration; All the services and their interactions are designed to provide capability of efficient and effective provenance knowledge discovery;

- **Knowledge-based service recommendation**: The construction of *in-silico* experiment / analysis process is critical for the scientific discovery. The system provides a recommendation feature to users to improve those. The *FlowRecommender* (Zhang et al., 2009) algorithm is developed to enable knowledge discovery from the historical provenance information for assisting workflow construction;

- **Provenance exploration**: The users are able to navigate the published workflows based on the semantic context by interactive graphical interface. The workflows are presented in an abstract view using visualization techniques and can be zoomed in for further details. It helps users to quickly identify the key information in a visual manner;

- **Provenance query**: We emphasize on how new query methods can better serve users' information need. In particular, we develop the two novel query methods, *Keyword Query* and *Graph Query*, by extracting knowledge from the historical provenance information. The visual graphical interface can help users to construct a complex query and return the desired information efficiently and effectively;

For easy presentation, in this work the terms ”workflow”, ”*in-silico* experiment” and ”analysis process” are used interchangeably. The paper is organized as following: We present the high level system architecture in Section 2; Section 3 introduces the system key components, their interactions and the algorithms developed to support knowledge discovery using provenance information; An user scenario is presented in Section 4 and followed by Conclusion.

2. System Architecture

A system level approach is applied to fully support knowledge discovery based on provenance information.

2.1. Overview

The system provides four types of functionality to users: workflow construction, workflow publish, provenance query and provenance exploration. Figure 1 shows the n-tier architecture of the system with all the services as building blocks.

- Front Layer: Four front services as part of the above four functionalities are presented for users interacting with the system. All the requests received from users are passed to the management layer;

- Management Layer: It includes seven building services: workflow modification service, workflow enactment engine, provenance record service, provenance retrieve service, knowledge discovery service, knowledge retrieve service and provenance visualization service. For each service, there are modules developed to perform specific functions such as workflow recommendation, pattern extraction and indexing etc. to support knowledge discovery

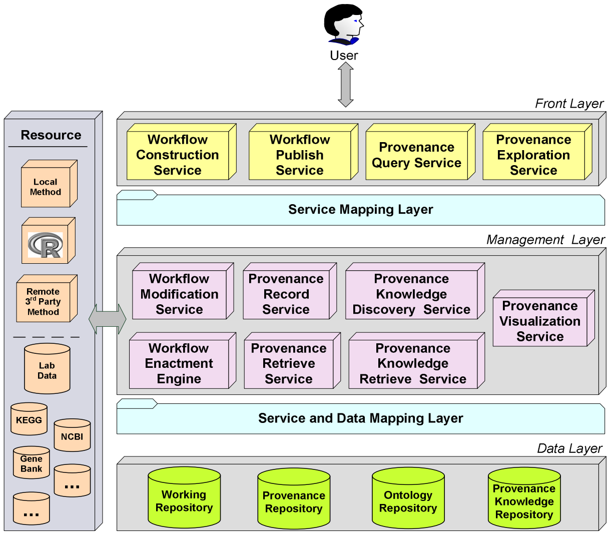

User
Front Layer
Workflow Construction Service
Workflow Publish Service
Provenance Query Service
Provenance Exploration Service
Service Mapping Layer
Management Layer
Workflow Modification Service
Provenance Record Service
Provenance Knowledge Discovery Service
Provenance Visualization Service
Workflow Enactment Engine
Provenance Retrieve Service
Provenance Knowledge Retrieve Service
Service and Data Mapping Layer
Data Layer
Working Repository
Provenance Repository
Ontology Repository
Provenance Knowledge Repository
Resource
Local Method
Remote 3rd Party Method
Lab Data
KEGG
NCBI
Gene Bank
...
...

Figure 1: Provenance System Architecture

using provenance inforamtion. The detailed service interactions and the algorithms developed will be introduced in Section 3.

- Data Layer: Four data repositories are designed: working repository, provenance repository, ontology repository and provenance knowledge repository.
  - All the workflows executed but not published are stored in the working repository.
  - Provenance repository records all the published workflow information which are ready to be shared.
  - All the knowledge extracted from historical provenance information are saved in the knowledge repository for the purpose of query, visualization and workflow recommendation.
  - The ontologies, that represents the semantic context of scientific domain information for the purpose of *in-silico* experiment / workflow classification, are put in the ontology repository. It has hierarchical structure. All the none leaf nodes are *virtual nodes* which do not have workflows associated with them directly. All the leaf nodes are *solid nodes* which could have workflows linked to. The circles represent the workflow instances. The hierarchical structure represents the semantic context of a particular workflow instance. Figure 2 shows an example in which ”Workflows” and ”Gene Analysis” are virtual nodes and ”Human”, ”Rat” and ”Protein-Protein Interaction” are solid nodes.

The semantic context of workflow $w1$ and $w2$ is they are about the gene analysis process for human. By classifying the workflow provenance based on their semantic context, the system is able to support scientists to discover the required provenance information more efficiently.

- Resource: All the analysis methods and data which are used for conducting the *in-silico* experiment are registered in the Resource (see Fig. 1). All the methods are implemented as Web services. This provides the system with flexibility such as on-demand system re-configuration and an interactive workflow construction environment. Furthermore, the biologists with little computing knowledge are able to not only use the local methods but also access the publicly available Web services and data sources to analyze the data. There are also some assisting services which provides input and output syntactic mapping between analysis methods.

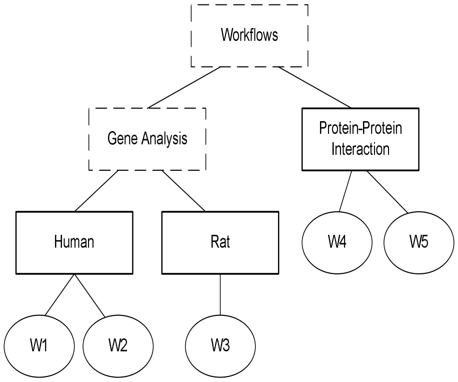


Figure 2: An Example of Ontology

2.2. Data Model

Since RDF and OWL provide a natural way to model provenance graphs, an OWL ontology with an underlying RDF store is designed to describe the data model for the provenance service system. It also provides flexibility to extend and associate the model with other ontologies if the need arises. Figure 3 shows the designed data model which consists of three main parts for capturing provenance information: Workflow Composition, Workflow Modification and Workflow Execution.

The Workflow Composition of the data model captures the attributes, components and structure of each Workflow. This constitutes the information stored in the Working Repository and Provenance Repository for users to be able to publish and retrieve workflows.

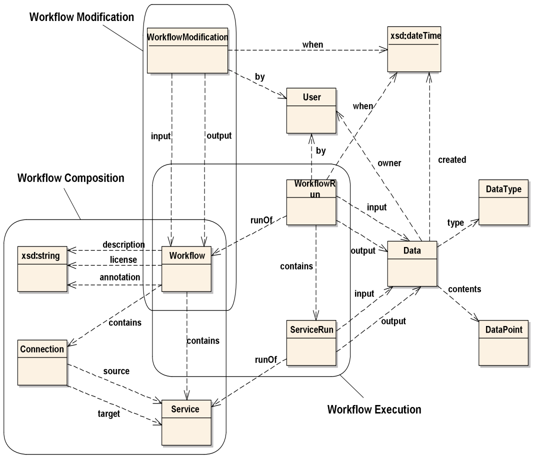


Figure 3: Data Model

The Workflow Modification of the data model captures the history of workflow modifications. This provenance information provides the acknowledgement and recognition of a previous contribution to intellectual property. It also helps to identify the relationships between the two workflows if the one is generated from the another.

The Workflow Execution of the data model describes the run of Workflows and captures the input, output and intermediate data produced during the run. This provides data provenance information describing by whom, when and how data was manipulated and produced.

We do not focus on the internal structure of the provenance model. The focus of this work is to mining the relationships among the workflow provenance to support knowledge discovery.

3. System Key Components, Interactions and Algorithms

We aim to promote the collaboration culture by showcasing the features required for data analysis and sharing. In the section, we will present the system key components, their interactions and algorithms developed to enable provenance exploration in which querying, navigation and visualization methods are combined together. Specifically, we will present (1) the workflow construction method using the service recommendation module to improve the efficiency and effectiveness of constructing analysis process; (2) the provenance navigation method for effective exploration and (3) provenance query methods in helping users to locate their desired information quickly.

3.1. Workflow Construction Service

The system promotes the knowledge sharing by re-using and/or modifying the existing published workflows in addition to constructing the workflows from scratch.

The *in-silico* experiment is conducted using the graphical user interface, executed by the workflow engine and stored in the Working Repository. The domain ontology, which classifies the workflows into the experiment groups for easy exploration, is provided to guide workflow description. There are three ways to compose the analysis process.

- **Re-using**: The users are able to re-run the workflows published in the system. This involves querying the existing workflows using Provenance Query Service. The executed workflow is recorded as an instance of the workflow defined;
- **Evolution**: The search results partially satisfy the user's demand and the user can revise the workflow to meet his/her own interest using Workflow Modification Service. In the provenance knowledge repository, this is represented as the workflow evolution which is captured by workflow modification in the data model designed. This provenance information facilitate the acknowledgement and recognition of previous contribution to intellectural property which is essential for information sharing.
- **Creation**: The user can construct a total new analysis process if he/she can not find any information useful. In the provenance repository, this will be recorded as a new workflow.

Figure 4 shows the interactions between the services. Given the large amount of services available, to create the workflow more efficiently, the workflow recommendation module, *FlowRecommender* algorithm (Zhang et al., 2009), is developed to provide step-wise recommendations.

The algorithm extracts patterns from the historical provenance information. Here the patterns represents the candidate methods which can be potentially used to extend/complete the partial workflows under construction. They can be easily found from the nodes in the provenance that have appeared in the workflows but do not only appear in the start position of the workflows. We utilize the measure of confidence to measure the strength of correlation between a node and its upstream sub-paths in the workflows. If the confidence of a node v given an upstream sub-path p in the workflow exceeds a given confidence threshold, then we call p as a pattern of v.

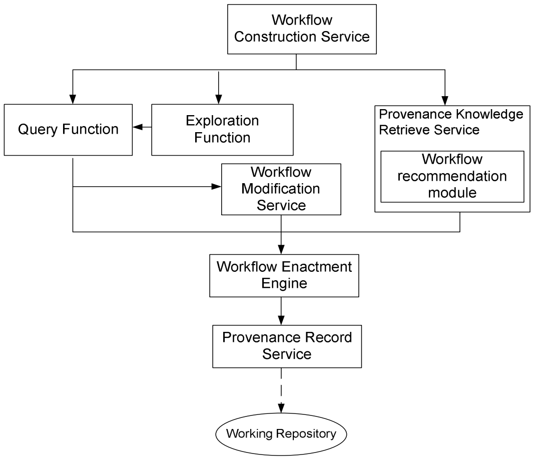

Workflow Construction Service
Query Function
Exploration Function
Provenance Knowledge Retrieve Service
Workflow recommendation module
Workflow Modification Service
Workflow Enactment Engine
Provenance Record Service
Working Repository

Figure 4: Workflow Construction Service

The patterns of the candidate nodes for the provenance are registered in the pattern table. The pattern table is a 3-dimensional table, where the first column in the table are the candidate nodes, the second column contains the corresponding pattern of each of the candidate nodes and the final column contains the value of confidence of each node with respect to its pattern. The pattern table is pre-constructed and make readily before workflow recommendation is performed. It is stored in Provenance Knowledge Repository. For details on pattern extraction and registration in the system, please refer to (Zhang et al., 2009).

During the construction phase, FlowRecommender tries to match the current workflow under construction against the pattern. The analysis method is recommended once such matching is successful.

3.2. *Workflow Publish Service*

One of important features of the system is real time publishing and sharing. After the user executed and published the workflow designed, the system triggers the knowledge discovery service in which pattern extraction and workflow similarity calculation will be conducted. The feature-graph index will also be updated to support provenance query. Figure 6 represents the service interactions discussed above.

3.3. *Provenance Exploration Service*

The system provides exploration feature for users to browse the published workflows based on their semantic context in the system. Two novel browsing methods have been developed to enable OLAP-like exploration of the results.

Building Ontology based on the Workflow Semantic Context: The system classifies the workflows into groups based on the semantic context of the workflows gathered from users during the workflow construction. This semantic description represents the biological study performed. Further sub-classification is possible depending on the application. This function points users directly into the area of the similar study. It greatly improves the efficiency and effectiveness of the search.

Visualizing by Structural Similarity: After directing the users into the specific workflow group, the system visualizes the relationships among the workflows with the same semantic context in an abstract view based on their structural similarities. The *similar* workflows are ”sitting” closer than those dis-similar ones. To enable this function, there are three steps involved for visualization based on the structural similarity:

- Similarity Calculation Method: Since each workflow is regarded as a graph, we apply the graph similarity method to measure the workflow similarity. In particular, the workflow similarity is defined by the maximum common induced subgraph (MCIS). Computing MCIS for unlabeled general graphs is shown to be NP-complete (Bunke and Shearer, 1998). However, we only consider workflows which are labeled directed graphs so the computation is reasonably fast.

  To serve for our specific purpose, the similarity degree of the two non-empty graphs G_1 and G_2 is modified and defined as

  $$d(G_1, G_2) = |mcs(G_1, G_2)|/max(|G_1|, |G_2|) \tag{1}$$

  where $mcs(G_1, G_2)$ is the maximal common subgraph of two graphs G_1 and G_2; $|G_x|$ represents the number of vertices in graph G_x. The pair-wise workflow similarities are stored in the workflow similarity metrics which locates in the Knowledge Repository.

- Weighted-graph construction: A weighted graph is constructed based on the similarity metrics. Each vertex represents a workflow and an edge represents there is a similarity relationship between the two workflows. The weight of the edge is defined by Equation (1). The updated weighted graph is also stored in the Knowledge Repository.

  A threshold θ is set up to control the number of edges in the weighted graph to be visualized for Provenance Visualization Service. Only if $d(G_1, G_2) > \theta$, an edge is constructed between G_1 and G_2 in the weighted graph. Without any threshold, the presented visualization may not be useful for users to understand the workflow relationships. For example, it is possible that most of the workflows start with a loadData service. It means they share the similarities but it is meaningless in terms of their semantic context if this relationship is visualized during exploration or query process. The relationship between the two graphs is only visualized when their similarity is above the threshold defined. This threshold needs to be setup / adjusted based on the domain experience. Since our framework is a generic framework, it is also possible to give different thresholds to different workflow groups defined in the ontology hierarchy. The goal is to present the workflow relationships which are meaningful for the domain users.

- Visualization: We employ NicheWorks by (Wills, 1997) to visualize the weighted graph constructed above. NicheWorks is a visualization tool for the investigation of very large graphs. In NicheWorks, there are four options for displaying the graph using the state vector:

  - Deleted - treat the data point as if it were not present
  - Normal - show the data
  - Highlighted - show the data so it will stand out against normal data
  - Focused - show as much detail as possible on the data

  For exploration purpose, all the workflows / vertex in the weighted graph are visualized. The ”deleted” state could be used for ”hiding” the workflow / vertex from visualization if needed. For example, if the workflow does not satisfy the query result, we can set ”deleted” state for the corresponding workflow and it ”disappears” from the screen by applying NicheWorks. Only the workflows which match the query criteria are presented to users.

Algorithm 1 shows a sketch of the whole visualization process which includes the above three components: (1) similarity calculation (line 2 - 6); (2) weighted graph construction (line 7 -13) and (3) visualization (line 14). Since the workflow similarity metrics is constructed based on the ontology hierarchy (which is also true for weighted graph), the visualization presented by NicheWorks is just for one group of workflows which belong to a solid node in the ontology. The Visualization Service builds virtual node to link all the groups together based on the ontology hierarchy and distributes the groups evenly in the space with the top vertex in the ontology hierarchy as a virtual node in the centre (line 16).

Figure 8 shows an example of exploration. In area UI-500, all the black vertices represent the workflows. All the blank vertices (eg. Workflow, Gene Analysis) represent the virtual nodes which are in the ontology hierarchy. The node

Algorithm 1 Structural Similarity Visualization

Input:

the current node S in the ontology hierarchy and a threshold θ

Output:

visualized workflow relationship

Description:

```
for each leaf node $S_{leaf}$ of $S$ do
    retrieve all the workflows $G_{leaf}$ which belong to $S_{leaf}$;
    for each workflow $G_x$ ($G_x \in G_{leaf}$) do
        calculate $d(G_x, G_y)$ if not exists ($G_y \in G_{leaf}$);
    end for
    generate similarity matrix for $G_{leaf}$;
    initialize graph $G_w$ and add all the workflows in $G_{leaf}$ as vertex to $G_w$;
    for each pair $G_x$ and $G_y$ do
        if $d(G_x, G_y) \geq \theta$ then
            draw an edge between $G_x$ and $G_y$;
            set weight of the edge as $d(G_x, G_y)$;
        end if
    end for
    pass $G_w$ to NicheWork for visualization;
end for
Distribute the above visualized graph evenly in the space with the root of ontology in the centre; link the root to the above visualized graph through the virtual nodes of ontology.
```

in the central, Workflow, represents the top level of the hierarchy. Its sub-group, gene analysis, links to the central node. The gene analysis group (a virtual node) includes human and rat two groups. This hierarchy structure is also reflected in the ”workflows” section (area UI-300). When users select the next level of ontology, ”Gene Analysis” for example, only the workflows belonging to ”Gene Analysis” are visualized accordingly.

By applying the above algorithm, the evolutional workflows have high possibility that they are presented as a cluster. The workflows which are not generated from evolution are also presented closer if they are similar. Therefore, it is easy for users to have a high level understanding of the differences among the workflows.

After a specific workflow (vertex) is selected, the Provenance Retrieve Service queries the Provenance Repository by workflow identification to get detailed workflow and presents it in area UI-500.

3.4. Provenance Query Service

The system supports two types of query methods: keyword query and graph query for users to identify desired information quickly. The output of the query is all the workflows which satisfy the query criteria and is presented using OLAP-like result browsing as described in Section 3.3.

Keyword Query: An inverted index is developed to provide efficient keyword query answering. It is based on the contents extracted from the workflow specification which includes the description, license information and the annotation attached to the workflow. The inverted index is stored in the Knowledge Repository.

When users send keyword query, the query algorithm retrieves all the workflows which contain the keyword extracted from query using the inverted index. All the workflows which are part of the query result are visualized using the methods described above.

Graph Query: The system also supports graph query. It is designed for users who want to form a more complex queries to improve the effectiveness of the query results. Users could construct graph query in area UI-400 by dragging and dropping the analysis methods from area UI-200. The process is similar to the workflow construction phase except no service recommendation provided. Compared with keyword query, graph query provides users an opportunity to specify the dependence relationship between the methods.

This concept has been discussed in the visTrial system (Freire et al., 2006) as query by example. However, the dependence relationship that users can expresse is limited to the direct link (see Figure 5(a) for example). Depends on users' knowledge, it is usual that the input methods and/or parameters associated with the methods are not linked directly. To address this issue, the system supports users to construct an approximate query workflow by creating two virtual services (see Figure 9 area UI-200):

- The first virtual service represents *one* and only one service which is not clear for users. We call this as *any service* represented by ”?”;
- The second virtual service, *any service path* represented by ”*”, represents more than one un-known services which are linked together;

By constructing these two virtual services, the users are able to form query in which the input methods can have grandparent-child relationship (see Figure 5(b, c)). Therefore, the system gives uses more query power to identify the information quickly.

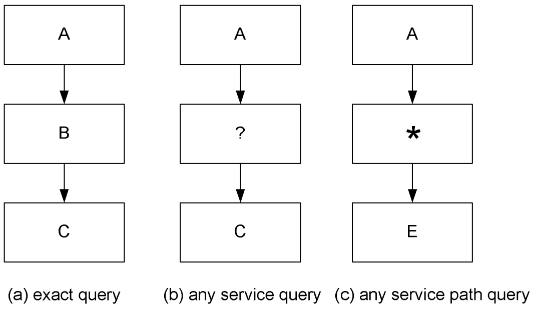


Figure 5: Virtual Services

A feature-graph matrix index is constructed for the purpose of answering graph query efficiently. The frequent sub-structures are selected as features from the existing workflows / graphs published in the system. Each row of the matrix corresponds to a feature being indexed while each column of the matrix corresponds to a workflow / graph. The original framework is proposed by (Yan et al., 2005) and is modified to suit our particular needs.

Given an exact query graph, the index filters as many workflows / graphs as possible to minimize the query time. Please refer to (Yan et al., 2005) for detail query method.

For an approximate query graph which involves virtual query service, the query algorithms first breaks the approximate query at virtual query service into two exact sub-query graphs. The two sub-query graphs then apply the query framework proposed by (Yan et al., 2005) respectively to filter the workflows /graphs that do not contain the sub-query graph. Only the workflows which contain both sub-query graphs are the candidate workflows. Then any traditional sub-graph isomorphism algorithm could be applied to prune the false positives.

Same as keyword query, the query result is visualized by applying visualization methods presented in Section 3.3.

3.5. Service Interaction

In this sub-section, we present how the services are interacted with each other in the proposed architecture to achieve the methods described above.

Figure 6 shows the interaction among different modules embeded in the Knowledge Discovery Service. All the modules are designed for the methods of: (1) workflow pattern extraction; (2) weighted graph construction and (3) inverted index and feature-graph matrix index construction. These modules are triggered when users push the ”publish” button for workflow publishing.

Figure 7 shows the interaction among different services and modules embeded in the Knowledge Retrieve Service. All the services and modules are designed for

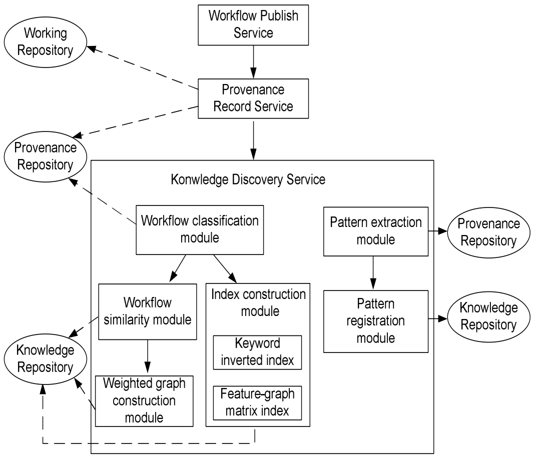

Workflow Publish Service
Working Repository
Provenance Record Service
Provenance Repository
Konwledge Discovery Service
Workflow classification module
Pattern extraction module
Provenance Repository
Workflow similarity module
Index construction module
Keyword inverted index
Feature-graph matrix index
Pattern registration module
Knowledge Repository
Knowledge Repository
Weighted graph construction module

Figure 6: Workflow Publish Service

the purpose of: (1) workflow recommendation; (2) exploration through visualization and (3) answering keyword query and graph query. The services and modules are triggered when query is received.

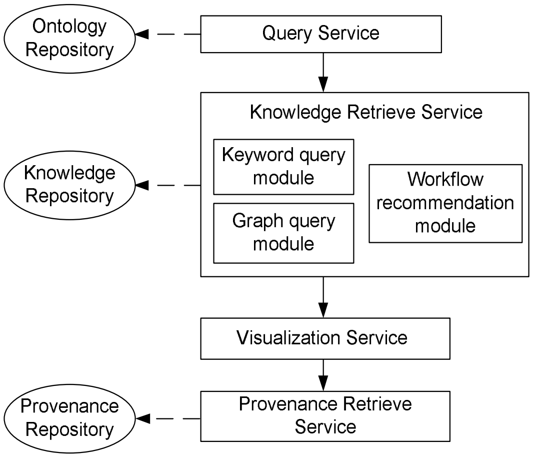


Figure 7: Provenance Query Service

4. Use Scenario

In this section, we first introduce our interface design and then explain the provenance exploration experience by our user group.

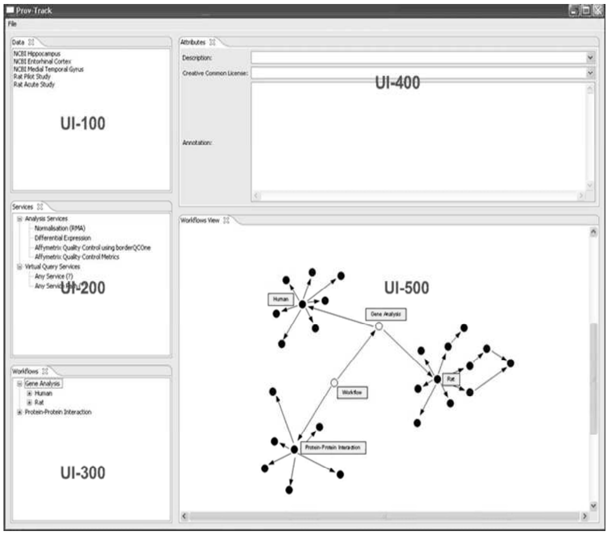


Figure 8: An Example of User Interface

4.1. User Interface Design

The user interface is designed based on the structure principle and simplicity principle. Figure 8 shows an example of an user interface which includes 5 function areas. All the resources which are available from the system are classified into 3 categories and presented on the left side of the interface:

- UI-100: the data source;
- UI-200: the analysis methods which are used to construct a workflow and presented as web services;
- UI-300: the workflows published within the system which could be shared and re-used by other users;

The right side of the user interface is a working area for users to construct/execute workflow and issue query and exploration. It includes:

- UI-400: It supports three functions:
  - describing attributes of a workflow when users are constructing a new workflow;
  - describing keyword query;
  - constructing graph query;

- UI-500: it supports four functions:
  - constructing a new workflow;
  - presenting workflow execution result;
  - presenting query result;
  - presenting exploration result;

The interaction between user interface and system functions is described in details in Section 3.

4.2. Use Case

Since our current user group is the biologists and bioinformaticians who work on the neurodegenerative disease, we use affymetrix microarray study of Alzheimer's disease as a case to present the system. In biological research, by studying the functions of the genes involved, biologists can have a better understanding of the underlying biology of the disease which can lead to early detection and prevention.

In our case, several types of molecular biology studies are available within the system: microarray analysis, protein-protein interaction, DNA sequencing and genome-wide association studies etc. One of the biology questions our user would ask is: *"find the top K differentially expressed genes in brain hippocampus area"*.

Firstly the user searches for existing analysis processes which are similar to the analysis process he/she wishes to perform. For this the Provenance Query Service is used and Figure 9 shows a screenshot of a graph query. When the user presses the 'Query' button, all the workflows which contain the query workflow are displayed and visualized in function area UI-500. Based on the semantic context the user want, several appropriate workflows are selected.

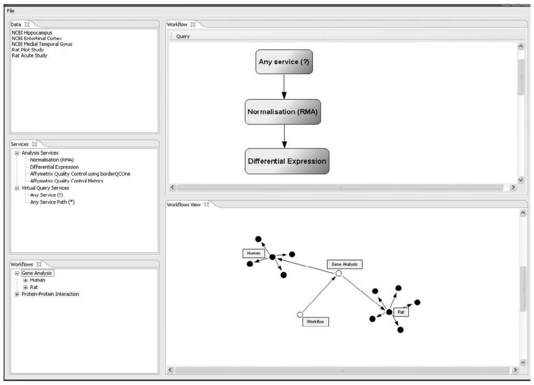
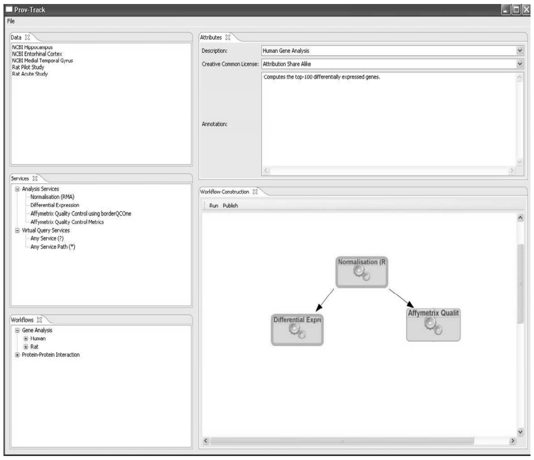


Figure 9: An Example of Graph Query

The user now modifies the workflow using the Workflow Construction Service which is shown in the screenshot of Figure 10. The attributes of the workflow are shown in function area UI-400 and the composition of the analysis services are shown in function area UI-500. At this point the user can change any of the attributes or analysis services. Additional analysis services can be selected from function area UI-200 or can be recommended by the Workflow recommendation service.



NCBI Hippocampus
NCBI Entorhinal Cortex
NCBI Medial Temporal Gyrus
Rat Pilot Study
Rat Acute Study

Description: Human Gene Analysis
Creative Common License: Attribution Share Alike
Annotation: Computes the top-100 differentially expressed genes.

- Analysis Services
  - Normalisation (RMA)
  - Differential Expression
  - Affymetrix Quality Control using borderQCOne
  - Affymetrix Quality Control Metrics
- Virtual Query Services
  - Any Service (?)
  - Any Service Path (*)

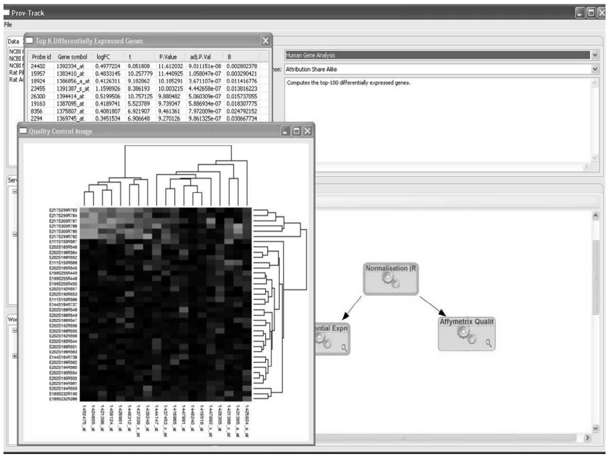
- Gene Analysis
  - Human
  - Rat
- Protein-Protein Interaction

Figure 10: An Example of Workflow Construction

When the user has modified the workflow they can then run it by pressing the ’Run’ button and the result of this is shown in the screenshot in Figure 11. If the output data that resulted from the execution of an analysis service is viewable, then the icon for the service includes a spy-glass as shown in function area UI-500. The output data can then be seen in a separate window as shown by the differentially expressed gene and quality control image windows in the screen shot.

In the next step of the use case scenario the user publishes the workflow, and execution data, by pressing the ’Publish’ button. This uses the Workflow Publish Service to store the workflow and execution information in the Provenance Repository.

The published workflow can now be discovered by the search services described above and Figure 8 show a screenshot resulting from the use of the Provenance Exploration Service. In function area UI-500 of the screenshot, all the workflows are arranged according to their hierarchical classification and the new human gene analysis workflow published by the user is also presented in the screenshot.



Probe id	Gene symbol	logFC	t	P.Value	adj.P.Val	B
24402	1392334_at	0.4977224	9.051808	11.612032	9.011151e-08	0.002802378
15957	1383410_at	0.4833145	10.257779	11.440925	1.058047e-07	0.003290421
18924	1386856_a_at	0.4126311	9.182862	10.185291	3.671107e-07	0.011416776
23455	1391387_s_at	1.1598926	8.386193	10.003215	4.442658e-07	0.013816223
26300	1394414_at	0.5199506	10.757125	9.880482	5.060309e-07	0.015737055
19163	1387095_at	0.4189741	5.523789	9.739347	5.886934e-07	0.018307775
8356	1375807_at	0.4081807	6.921907	9.461361	7.972009e-07	0.024792152
2294	1369745_at	0.3451534	6.906648	9.270126	9.861325e-07	0.030667734

Human Gene Analysis

License: Attribution Share Alike

Computes the top-100 differentially expressed genes.

Figure 11: An Example of Workflow Output

5. Conclusion

In this work, we present a knowledge discovery service system for provenance exploration in which analysis process could be constructed, executed, published, queried and visualized. Workflow recommendation are developed for efficient workflow construction. We introduce keyword query, exact graph query and approximate graph query for users allocating desired information quickly. Further more, navigation and visualization method are proposed to represent the semantic context similarity and structural similarity relationships among the workflows respectively. Our experience with our user group demonstrates the effectiveness and efficiency of the our service system. In future, we will investigate more efficient methods for graph visualization.

Acknowledgment

The CSIRO Tasmanian ICT Centre is jointly funded by the Australian Government through the Intelligent Island Program and the CSIRO. The Intelligent Island Program is administered by the Tasmanian Department of Economic Development and Tourism.

References

Altintas, I., Berkley, C., Jaeger, E., Jones, M., Ludscher, B., Mock, S., 2004. Kepler: An extensible system for design and execution of scientific workflows. In: IN SSDBM. pp. 21–23.

Barga, R., Jackson, J., Araujo, N., Guo, D., Gautam, N., Simmhan, Y., 2008. The trident scientific workflow workbench. In: Proceedings of the 2008 Fourth IEEE International Conference on eScience. IEEE Computer Society, Washington, DC, USA, pp. 317–318.
URL http://portal.acm.org/citation.cfm?id=1488725.1488936

Brazas, M. D., Fox, J. A., Brown, T., McMillan, S., Ouellette, B. F. F., 2008. Keeping pace with the data: 2008 update on the bioinformatics links directory. Nucleic acids research 36.

Bunke, H., Shearer, K., 1998. A graph distance metric based on the maximal common subgraph. Pattern Recognition Letters 19 (3-4), 255–259.

Callahan, S. P., Freire, J., Santos, E., Scheidegger, C. E., Silva, C. T., Vo, H. T., 2006. Vistrails: Visualization meets data management. In: In ACM SIGMOD. ACM Press, pp. 745–747.

Cruz, S. M. S. d., Campos, M. L. M., Mattoso, M., 2009. Towards a taxonomy of provenance in scientific workflow management systems. In: Proceedings of the 2009 Congress on Services - I. IEEE Computer Society, Washington, DC, USA, pp. 259–266.
URL http://portal.acm.org/citation.cfm?id=1590963.1591561

Deelman, E., Singh, G., hui Su, M., Blythe, J., Gil, A., Kesselman, C., Mehta, G., Vahi, K., Berriman, G. B., Good, J., Laity, A., Jacob, J. C., Katz, D. S., 2005. Pegasus: a framework for mapping complex scientific workflows onto distributed systems. Scientific Programming Journal 13, 219–237.

Freire, J., Silva, C. T., Callahan, S. P., Santos, E., Scheidegger, C. E., Vo, H. T., 2006. Managing rapidly-evolving scientific workflows. In: International Provenance and Annotation Workshop (IPAW). LNCS 4145. Springer Verlag, pp. 10–18.

Galperin, M. Y., November 2007. The molecular biology database collection: 2008 update. Nucleic Acids Research.

Gil, Y., Ratnakar, V., Deelman, E., Mehta, G., Kim, J., 2007. Wings for pegasus: creating large-scale scientific applications using semantic representations of computational workflows. In: Proceedings of the 19th national conference on Innovative applications of artificial intelligence - Volume 2. AAAI Press, pp. 1767–1774.
URL http://portal.acm.org/citation.cfm?id=1620113.1620127

Kim, J., Gil, A., Ratnakar, V., 2006. Semantic metadata generation for large scientific workflows. In: In Proceedings of the Fifth International Semantic Web Conference. pp. 5–9.

Moreau, L., Plale, B., Miles, S., Goble, C., Paolo Missier, R. B., Simmhan, Y., Futrelle, J., McGrath, R., Myers, J., Paulson, P., Bowers, S., Ludaescher, B., Kwasnikowska, N., den Bussche, J. V., Ellkvist, T., Freire, J., Groth, P., 2008. The open provenance model (v1.01)s. Tech. rep.

Oinn, T., Addis, M., Ferris, J., Marvin, D., Carver, T., Pocock, M. R., Wipat, A., 2004. Taverna: A tool for the composition and enactment of bioinformatics workflows. Bioinformatics 20, 2004.

Simmhan, Y. L., Plale, B., Gannon, D., April 2008. Query capabilities of the karma provenance framework. Concurr. Comput. : Pract. Exper. 20, 441–451. URL http://portal.acm.org/citation.cfm?id=1350745.1350749

Wills, G. J., 1997. Nicheworks - interactive visualization of very large graphs. In: GD ’97: Proceedings of the 5th International Symposium on Graph Drawing. Springer-Verlag, London, UK, pp. 403–414.

Yan, X., Yu, P. S., Han, J., 2005. Substructure similarity search in graph databases. In: SIGMOD ’05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data. ACM, New York, NY, USA, pp. 766–777.

Zhang, J., Liu, Q., Xu, K., 2009. Flowrecommender: A workflow recommendation technique based on process provenance. In: The 8th Australasian Data Mining Conference (AusDM’09). Melbourne, Australia.