Candidate Workflows for Provenance Challenge 3

Satya S. Sahoo[1], Yogesh Simmhan[2], Roger Barga[2]
[1]Kno.e.sis Center (Wright State University), [2] Microsoft Research

Date: 10/22/2008

Introduction

We propose two candidate workflows from the Neptune and Pan-STARRS eScience projects at Microsoft Research for the provenance challenge. In addition to our candidate workflows, we also propose the following three aspects of provenance to be evaluated as part of the challenge:

1. **Provenance Collection**: Previous provenance challenges have focused on provenance collection, but primarily in the context of a scientific workflow. We would like to extend the notion of provenance collection beyond scientific workflows to include scenarios that often do not involve orchestration by a workflow engine. For example, the oceanography scenario (Neptune Project) requires the collection of provenance information describing the sensors collect the readings for input to a chart visualization workflow.

   We believe that this form of provenance collection is also a critical part of the overall provenance collection framework and needs to be explored by the community.

2. **Provenance Representation**: Representation and interoperability of provenance has also been explored by previous provenance challenges. Our candidate workflows involve complex control flows such as the Pan-STARRS workflows include double-nesting, loop structures and conditional branching. Ability to track provenance of data within collections is another interesting aspect.

3. **Provenance Analysis**: Unlike previous challenge we also propose the performance evaluation of provenance queries over large datasets in this challenge. The results from this challenge can be used to create a "**Provenance Performance Benchmark**" for the community with associated provenance dataset and list of template queries.

In the next section, we describe the two eScience scenarios and their associated workflows for this provenance challenge.

I. Oceanography Scenario (Neptune Project)

Description:

The Neptune project, led by the University of Washington (http://www.neptune.washington.edu/), is an ongoing initiative to create network of instruments widely distributed across, above, and below the seafloor in the northeast Pacific Ocean. We consider a simulated scenario, illustrated in Figure 1.1, involving collection of data by ocean buoys (containing a temperature sensor and an ocean current sensor) which is then sent as input to a scientific workflow for creation of a visualization chart as output.

The temperature sensor and the ocean current sensor, contained in an ocean buoy, collect ocean temperature and the ocean current velocity readings respectively. The readings are collated by a process called "DataCollectionProcess", the collated readings in a form of "NcFile" file type, along with a set of parameters, are given as input to the "HyperCubeSchemaGeneratorActivity" process. This process is part of a scientific workflow that is used in the Neptune project to generate visualization charts. The scientific workflow involves the sequential execution of four processes along with a set of parameters and data values as input.

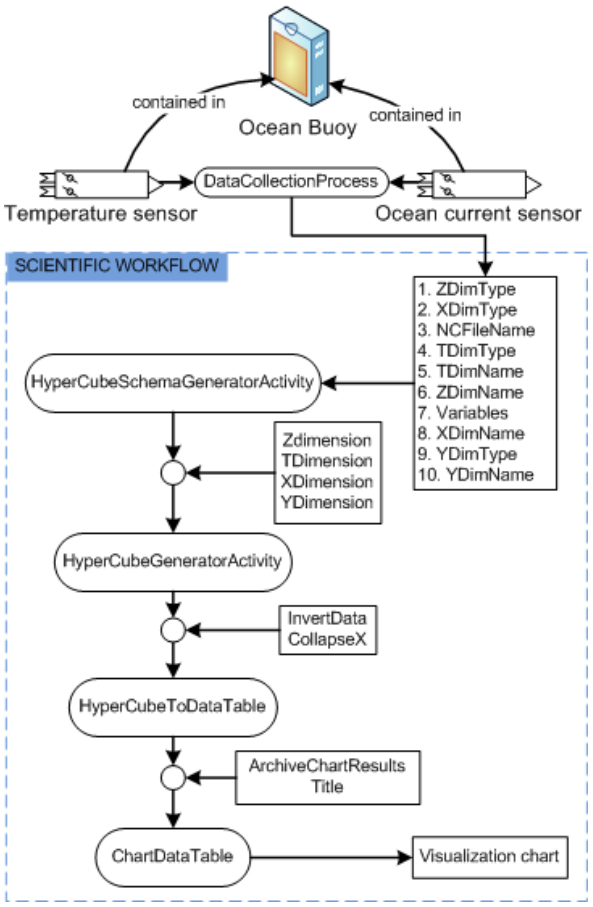

Figure 1.1: A simulated oceanography scenario from Neptune Project with data from sensors used to create chart visualizations

Novelty of this scenario: The novelty of this oceanography scenario is the inclusion of provenance details beyond the scientific workflow view such as the details regarding the two types of sensors and the ocean buoy containing them. Provenance researchers have long realized the need to include detailed provenance represented in expressive format beyond provenance information generated as part of a scientific workflow. Hence, through our oceanography scenario we propose provenance challenge to explore the issues involved in effectively collecting, representing provenance information that are not part of a scientific workflow. We note that this provenance information needs to be integrated with the provenance generated by scientific workflows.

The following example queries can be executed over the provenance information collected for the oceanography scenario:
Example Query 1: If an ocean buoy 'X' is found to be damaged through contamination with sea water, all visualization charts generated using data from the two sensors within this ocean buoy should be discarded.
Example Query 2: Output the provenance information for a specific data value, "HyperCubeX", in form of a structure such as a graph that can be used easily sent as input to a visualization tool for provenance visualization.

II. Astronomy Scenario (Pan-STARRS Project)

Description:

The Panoramic Survey Telescope & Rapid Response System (Pan-STARRS) will perform a detailed survey of the visible universe and build a time series of astronomical detections to track moving objects. Microsoft

Research is working with University of Hawaii and Johns Hopkins University to create the infrastructure to manage large data generated by Pan-STARRS (~30TB/year) [*Simmhan et. al, 2008*]. We propose two workflows from the Pan-STARRS project for the provenance challenge.

1. Load Workflow

This workflow loads a batch of CSV (comma separated value) files representing telescope detections into a single staging database for that batch. The workflow does pre-validation of the CSV files, creates an empty relational database, and loops through each file loading it into a unique table in the database. One or more pre- and post- validations are performed on each file and the entire batch after the load.

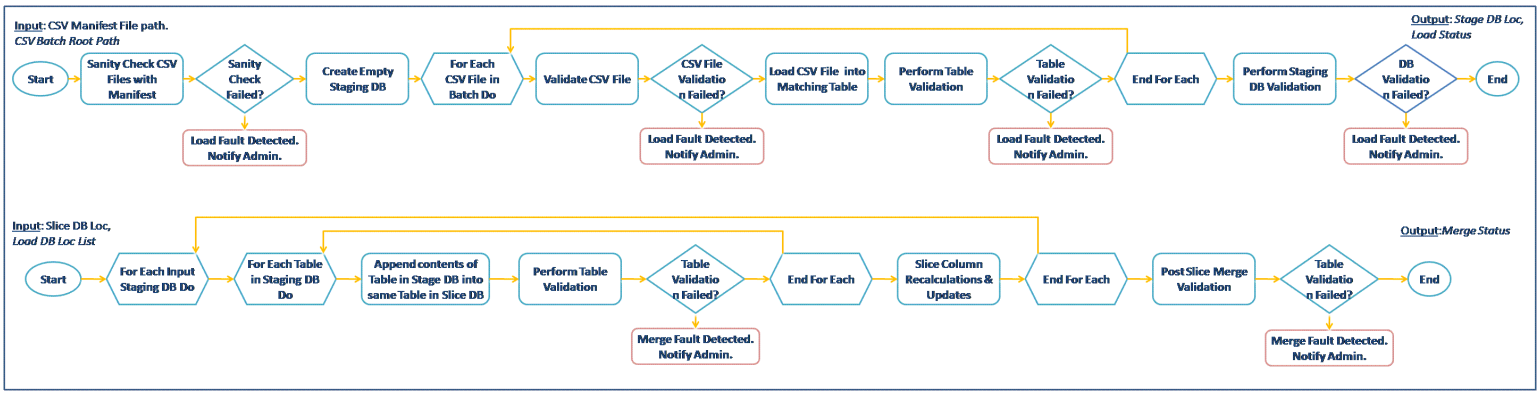

Figure 2.1: Pan-STARRS workflows (a) Load Workflow loads data files in a staging database with a set of validation steps, (b) Merge Workflow appends new data from several staging databases to existing slice database.

Novelty of this workflow: This workflow incorporates two control structures besides sequential dataflow – a loop structure over the set of files, and an if-else decision structure after each validation step to decide if we should proceed with the next step in the workflow.

2. Merge Workflow

This workflow takes a set of staging databases created by the load workflow and incrementally merges it with an existing slice database. Both these databases have the same schema. The merge loops through each input staging database and through each table in the input database and appends that table with an existing table with the same name in the slice database. Post-merge validations and updates to computed-columns are performed. The input to this workflow is a set of staging databases and a slice database; the output entities are the merged slice database and merge status messages (success, failed, errors).

Novelty of this workflow: The staging database acts as a collection of tables that can be seen, with one unit for passing around but are peering inside into individual tables when merging. This workflow also uses double-nesting.

[*Simmhan et. al, 2008*] *GrayWulf: Scalable Software Architecture for Data Intensive Computing.* Yogesh Simmhan, Maria Nieto-Santisteban, Roger Barga, Tamas Budavari, Laszlo Dobos, Nolan Li, Michael Shipway, Alexander S. Szalay, Ani Thakar, Jan Vandenberg, Alainna Wonders, Sue Werner, Richard Wilton, Dan Fay, Michael Thomassy, Catharine van Ingen, Jim Heasley, Conrad Holdberg. Hawaii International Conference on System Sciences (HICSS), 2008..