

## Provenance For Service Reliability

This document details Use Cases relating to the use of Provenance for determining and tracking service reliability. The reliability of a service can be defined by many factors including:

- how long the service has been continuously available
- how quick the service responds
- the time it takes for the service to return a result for a particular input
- how fast a service is in comparison to other services offering the same functionality

### Use Case 1: Alternative Service Selection

Goal: The ability to manually or automatically select an alternative service to use if a service fails to respond during any part of a workflow execution.

The enactment system alerts the user that a service has failed in some way, it may be unresponsive or return an error, and requires the user to either select an alternative service to use or allow the system to select a service.

#### Requirements

##### UI

- A graphical representation of a failed activity in the workflow
- Ability to interact with a workflow during enactment when a service failure is reported
- Ability to check the health of an activity before enactment
- A view of alternative activities and the ability to select one.
- The ability to put alternative activities into a workflow

##### Provenance Model

- Identify the workflow model used for a given workflow run
- Resolve data references to the actual data from a workflow run
- Record error information pertaining to the failed service including when it was down and why

### Use Case 2: Provenance data as cache

Goal: A service fails and no alternatives are available so the enactment engine selects the results from a previous run of this activity with identical inputs.

##### UI

- A graphical representation of a failed activity in the workflow
- Ability to interact with a workflow during enactment when a service failure is reported
- Show the user that there are no alternatives or that alternative services are unavailable
- Show the user that a previous workflow used this service (or an alternative) with identical inputs
- Show the results for this service for identical inputs
- The ability to tell the enactment engine to automatically select alternative services based on provenance
- The ability to tell the enactment engine to stall a workflow run until a service is available

##### Provenance Model

- Record error information pertaining to the failed service including when it was down and why
- Resolve data references to the actual data from a workflow run
- Record inputs and output data

Record that the results from this service in this workflow run are from provenance data

#### Use Case 3: Provenance data during workflow design

When designing a workflow the user wants the system to flag that other workflows have used this service before and to have the ability to see the health statistics and inputs/outputs. The user wants the choice of using this provenance data.

UI

Display a list of workflows which have used this activity along with a simple traffic light style view of the health.

Allow the display of more in depth health statistics for a service

Show the results for a selected use of this service.

Allow a service to be flagged as using provenance data instead of live results

#### Provenance Model

Record the results from a service along with the inputs and when it was run

Record if a service was available

#### Use Case 4: Display workflow differences

Post execution the user wants to see the differences between the current run and previous runs of the same workflow. The user wants to graphically visualise which services have the same results as another run and which do not. The differences can be tabulated from many workflow runs and inputs, outputs and differences highlighted.

UI

Highlight services which have different outputs from a previous run

Have the ability to display the results from previous workflow runs

Tabular representation of results

Compare more than two runs and display in a 3D table view.

The ability to jump from one difference to the next graphically

The ability to compare two or more workflow runs

Show the number of times a workflow has been run

#### Provenance Model

Record the inputs and outputs from a workflow run

Dereference the actual output/input values from a workflow

The ability to compare workflow runs

Compare outputs from a service from different runs of the same workflow

#### Use Case 5: Smart re-runs of a workflow

During design the user does not want to re-run an entire workflow but wants to run from a selected point. If the workflow has a previous execution then this data can be used, if not then the system can discover data to be re-used from provenance and do a partial re-run of the workflow, injecting the provenance data when necessary.

UI

Ability to select a point in a workflow to execute from

Ability to tell the enactor to only run new services added to a workflow

Remember what parts of a workflow have been run previously

#### Provenance Model

Record the inputs and outputs from a workflow run  
Query to discover provenance which uses the same inputs  
Dereference the output from a service to the actual data