The Provenance FAQ

Authors:	Sheng Jiang, Luc Moreau, Paul Groth, Simon Miles, Steve Munroe, and Victor Tan
Reviewers:	All project partners
Identifier:	The Provenance FAQ
Type:	Internal
Version:	1
Version:	November 29, 2006
Status:	Confidential

Abstract

The Provenance FAQ is a selection of frequently asked questions and responses relating to the provenance store and its use. It operates as a repository of knowledge that the project partners can consult in the case of misunderstandings over some of the more complex aspects of the provenance architecture.

The PROVENANCE project receives research funding from the European Commission's Sixth Framework Programme

Members of the PROVENANCE consortium:

IBM United Kingdom Limited	United Kingdom
University of Southampton	United Kingdom
University of Wales, Cardiff	United Kingdom
Deutsches Zentrum fur Luft- und Raumfahrt s.V.	Germany
Universitat Politecnica de Catalunya	Spain
Magyar Tudomanyos Akademia Szamitastechnikai es Automatizalasi Kutato Intezet	Hungary

The PROVENANCE project receives research funding from the European Commission's Sixth Framework Programme

Provenance FAQ

This page lists common questions from partners about Provenance.

How to pass time stamp information?

Time information should be passed within actor state p-assertions. Best practice is to use timestamp schema.

How to document an actor name?

Actor names should be passed as an actor state. Best practice is to useActor profile schema.

How to document an actor type?

Actor type should be passed as an actor state. Best practice is to use Actor profile schema.

How to add an annotation to a data item?

The provenance system is not an generic annotation system, but restricted annotations can be supported in the form of actor states.Best practice is to use Annotation schema.

How to use interaction context and exposed metadata?

A complete description given in the Linking 101 document.

How are p-assertions about the same interactions, generated by different actors to be linked? Isn't there a need to pass around local p-assertion ID's?

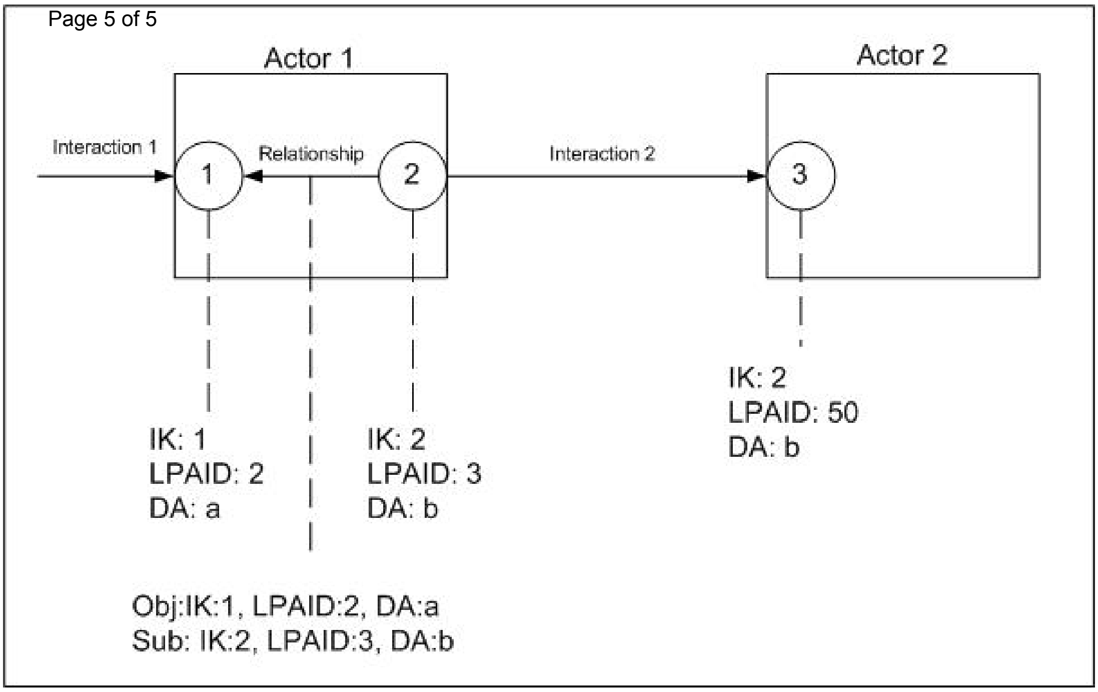
It is true that each actor recording a p-assertion about the same interaction will generate a different local p-assertion id (LPAID), this is correct. There is however, no need to either:

- a) pass these LPAID's around or,
- b) make any explicit relationship p-assertions between interaction p-assertions of different actors for the same interaction. (indeed, the architecture makes it clear that relationship p-assertions are only for connecting incoming and outgoing interactions **within** a given actor not **between** actors).

In order to link up p-assertions from the same interaction but recorded by separate actors 2 things need to be matched up:

- 1. The Interaction Key
- 2. The Data Accessor.

The figure below should make this clearer. I will explain the figure.



Two actors are shown (actor 1 and actor 2). Actor 1 has received a message (marked as interaction 1) from another (unseen) actor and records an interaction p-assertion (1) about it with interaction key (IK) = 1, a local p-assertion id (LPAID) = 2 and a data accessor (DA) = a. Actor 1 then sends another message to actor 2, marked as interaction 2 and records an interaction p-assertion stating that for this interaction IK=2, LPAID=3, DA = b. Actor 1 also records a relationship p-assertion stating that interaction 2 was caused by interaction 1.The relationship p-assertion states that the object of the relation is contained within interaction 1 and is identified by the data accessor (DA) = a. The relationship p-assertions also states that its subject is contained within interaction 2 and is identified with the DA = b.

Now, Actor 2, upon receiving the message sent to it from actor 1 records an interaction p-assertion (3) with the following characteristics (IK=2, LPAID = 50, DA = b) notice that both the IK and the DA are the same as that recorded for this interaction by actor 1. Only the LPAID is different, since this is **local** to actor 2.

Now, when a tool (either a querying tool or a visualisation tool) tries to link interaction p-assertion 3 back to interaction p-assertion 2 it cannot use the LPAID's of the two p-assertions since they are different. It must therefore, examine both the interaction key (thus identifying the interaction both assertions relate to) and **also** the data accessor (these are passed in the p-header). By doing this both p-assertions are linked to the same interaction and thus the interaction is identified intensionally, i.e. identified without the use of a unique id --- by the IK and the DA.

How do I use anonymous endpoints?

Due to the range of network technologies currently in wide-spread use (e.g., NAT, DHCP, firewalls), many deployments cannot assign a meaningful global uri to a given endpoint. To allow these ``anonymous" endpoints to initiate message exchange patterns and receive replies, WS-Addressing defines the following well-known uri for use by endpoints that cannot have a stable, resolvable uri: http://www.w3.org/2005/02/addressing/role/anonymous.

-- SteveMunroe - 25 Aug 2006