PVMv2 Specification

Thomas Bytheway
Lucian Carata

August 31, 2018

1 Introduction

2 Abstract Types

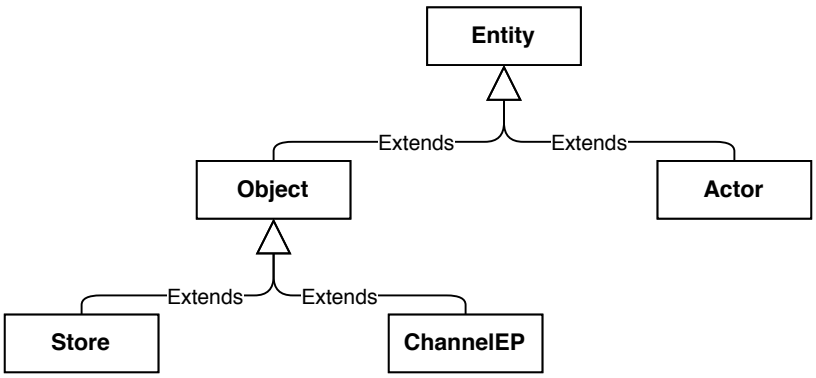

Figure 1: A diagram of the basic abstract types in PVMv2

Entity

Actor Active entities that perform actions. They act upon objects, or other actors. Generally processing elements like unix processes.

Object Passive entities that are acted upon by actors. Generally data carrying entities.

Store Object type that stores data internally. Versions from first write to last write, using EditSessions.

ChannelEP Object type that data flows through without internal storage. Does not version. May participate in connections.

3 Graph Schema

3.1 Nodes

3.1.1 Entity Nodes

3.1.2 Non-canonical Names

Path Names

Network Addresses

3.2 Relations

INF Indicates that the source has potentially imparted data or control to the destination.

NAMED Indicates that the source has been mentioned by the destination non-canonical name.

4 Mappings

4.1 Concrete Type Declaration

4.2 Verbs

Declare Forces the creation of entity if it does not already exist.

Sink Called when as part of a mapped function, an actor sinks data to an entity as an atomic operation. Causes Files to version, creating a new File entity connected to the previous one by an INF relation. Creates an INF relation from actor to entity, attaching tag as the type property.

Source Called when as part of a mapped function, an actor sources data from an entity. Creates an INF relation from entity to actor, attaching tag as the type property.

Connect Called as part of a mapped function to declare that two conduit nodes are connected to each other and that data can flow between them. The dir argument indicates the direction of flow, either mono or bi. Mono indicates that data may only flow from the first conduit to the second, bi indicates that data may flow in both directions.

Mention

Unlink

Property