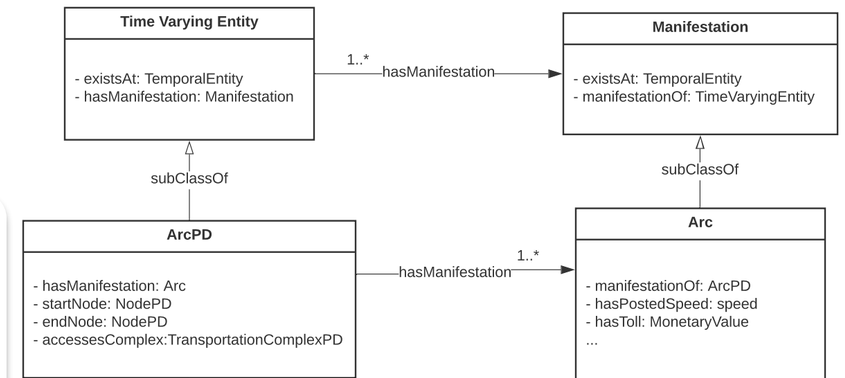
RepresentingChange: Arc Example

Megan Katsumi | May 14, 2020

*The **TimeVaryingEntity** class represents the class of all objects that are subject to change. Instances of this class demonstrate only invariant properties. Variant properties are demonstrated by way of the related manifestations.*

*The **ArcPD** class represents the class of arcs in a transportation network. Each arc in the real world corresponds to one instance of the ArcPD class. This class demonstrates only invariant properties: start node, end node, and the transportation complex (e.g. road segment) that the arc corresponds to. Variant properties are demonstrated by way of the related arc-manifestations.*



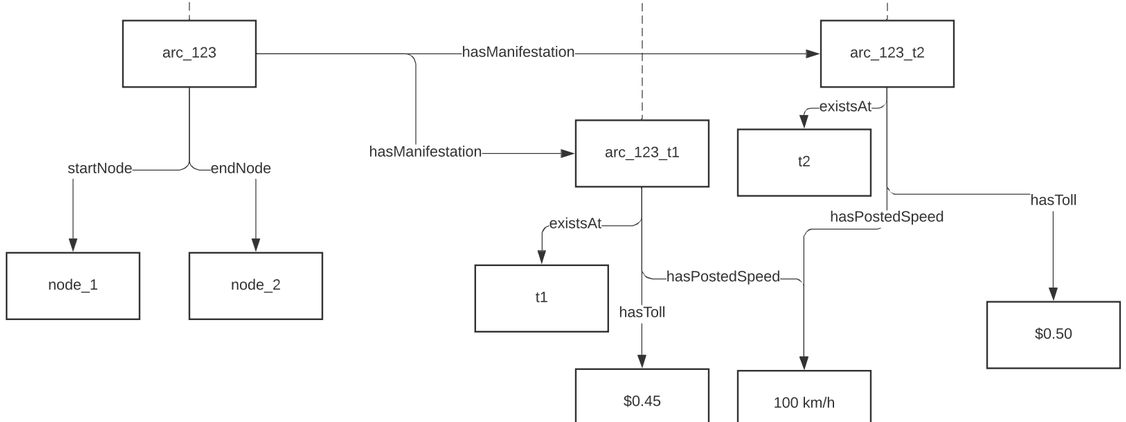
*The **Manifestation** class represents the class of all objects that are subject to change, as they exist at some point in time. Instances of this class demonstrate invariant properties, and any changes to property values are captured by new instances. A manifestation's identity is captured by way of its relationship to a single time varying entity.*

*The **Arc** class represents the class of arcs in a transportation network, as they exist throughout various points in time. Instances of this class demonstrate variant properties of arcs, such as speed limits or tolls. Any changes to these variant property values are captured by new instances.*
This allows for the representation of a different set of constraints on these properties that would not be possible if capturing an arc as a single instance. For example, an arc (Time Varying Entity) may have a number of different speed limits over its lifetime, but at any one point in time, an arc (Manifestation) should have just one speed limit.

Class-Level

Instance-Level

*The **arc_123** object is an instance of **ArcPD** that represents an individual arc in the transportation network. For as long as this arc exists, it will start at node **node_1** and end at **node_2**. Changes to the arc's other properties can be captured through its manifestations, **arc_123_t1, arc_123_t2,...***



*The **arc_123_t1** object is an instance of an **Arc** that represents **arc_123** at some point in time. For as long as this arc-manifestation exists (the interval or instant specified by t1), **arc_123** has a toll of $0.45 and a posted speed of 100 km/h.*
*The **arc_123_t2** object is an instance of an **Arc** that represents **arc_123** at some (implicitly later) point in time. For as long as this arc-manifestation exists (the interval or instant specified by t2), **arc_123** has a toll of $0.50 and a posted speed of 100 km/h.*

Representing Change

In order to represent changing attributes of an object, without losing information about its past values/relationships, we define two classes to describe such objects.

In the case of an arc in a transportation network (as in the example here), this results in two Arc classes: one class to capture the Arc and its invariant attributes, and another class to capture the variant attributes. Any arc will be represented with a single instance of the invariant class, and at least one (but likely many) instances of the variant class.

In the example depicted here, the invariant class is labelled "ArcPD" where -"PD" is short for "perdurant" (note: this is an artefact of the ontological philosophy behind this modelling approach, there is no need to preserve this style of labelling going forward). The variant class is simply labelled "Arc".

Note that the example properties have been simplified for the purposes of the example.