Proposed Disposition: Resolved

OMG Issue No: 10896

Title: Behavioral Features

Source:

NIST (Mr. Conrad Bock, conrad.bock@nist.gov conradb@cme.nist.gov)

Summary:

In Secton 16.6.1 (Behavioral Features), first paragraph is about a number of things other than behavioral features, and much of it is incorrect or uses incorrect terminology. Behavioral features only declare capabilities or services, not resources. They aren't the "program" that implements the service (called Behavior in UML). Behavioral features can be used in OCL that defines a derivation of a property, but the behavioral feature isn't directly related to the derived property. Operations include the parameters (including return value). A "method" in UML is the behavior that implements the operation on a particular class. Responsibility in UML is only a standard stereotype of a usage dependency. It isn't a well-developed part of UML class modeling. Qualified associations are more accurately described as a special kind of ternary relation. An abstract can have operations and methods like any other class. Abstract classes cannot have direct instances. Interfaces specify features of classes, including operation features. They aren't interfaces of operations themselves.

Resolution:

Replace text as described below.

Revised Text:

The following is the current contents of this paragraph:

> UML allows the specification of behavioral features, which declare capabilities or resources. One use of behavioral features is to calculate property values. Behavioral features can be used in the OCL that derives properties. Facilities of UML supporting programs include **operations**, which describe the parameters of methods; **static operations**, which are operations attached to a class like static attributes; **interface classes**, which specify among other things operation features; **qualified associations**, which are a special kind of ternary relation; and **active classes**, which are classes each instance of which controls its own thread of execution control.

Replace this paragraph with the following:

> UML allows the specification of behavioral features, which declare capabilities and dynamic aspects of the system. OCL can be used to restrict derived properties. Facilities of UML that can be used to describe application programs include **operations**, which describe the parameters of methods; **static operations**, which have a shared implementation for all subclasses; **interface classes**, which specify an interface to a set of attributes and operations that could be implemented by one or more classes; and **active classes**, which are classes that have a separate thread of execution control for each instance.

Proposed Disposition: Resolved