

20 InformationFlows

20.1 Information Flows

20.1.1 Summary

The InformationFlows package supports exchange of information between system entities at high levels of abstraction. InformationFlows may be useful during top-down model development, representing aspects of models not yet fully specified, and for recording less detailed, heuristic representations of more complex model areas. In these ways, InformationFlows can help to clarify and document overall understanding of the intent of large or complicated models.

InformationFlows describe circulation of information through a system in a general manner. They do not specify the nature of the information, mechanisms by which it is conveyed, sequences of exchange, or any control conditions. During more detailed modeling, representation and realization links may be added to specify which model elements implement an InformationFlow and to show how information is conveyed. Similarly, InformationItems can be used to represent the information that flows along InformationFlows even before details of their realization have been designed.

The contents of the InformationFlows package are shown in Figure 20.1.

20.1.2 Abstract Syntax

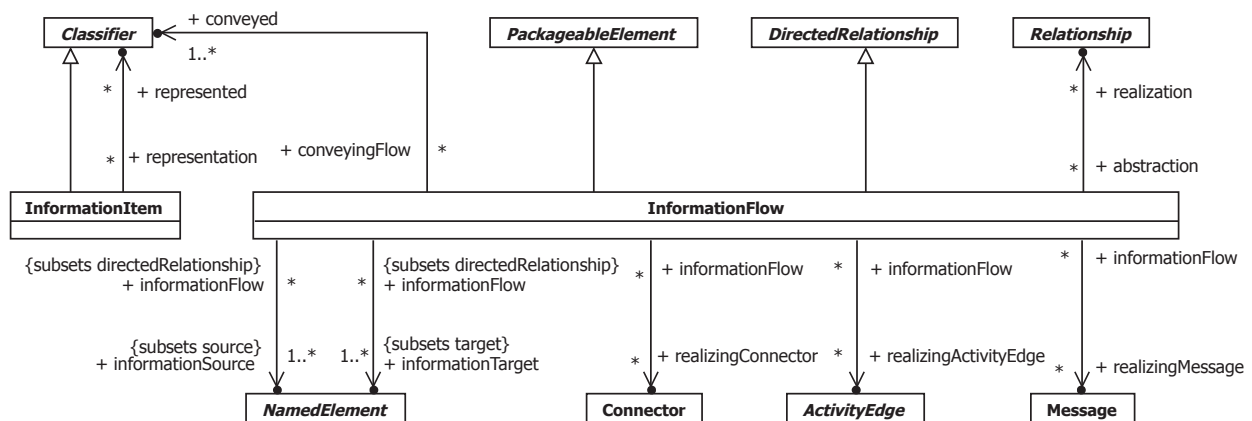


Figure 20.1 Information Flows

20.1.3 Semantics

InformationFlows require some kind of “information channel” for unidirectional transmission of information items from sources to targets. They specify the information channel’s realizations, if any, and identify the information that flows along them. Information moving along the information channel may be represented by abstract InformationItems and by concrete Classifiers.

The sources and targets of an InformationFlow designate sets of objects that can send (sources) or receive (targets) conveyed InformationItems or Classifiers. In Figure 20.1, sources and targets are shown as NamedElements. In practice, a constraint on InformationFlow requires that sources and targets must be one of the following types: Actor, Node, UseCase, Artifact, Class, Component, Port, Property, Interface, Package, ActivityNode, ActivityPartition, Behavior, and InstanceSpecification. Furthermore, when a source or target is an InstanceSpecification, it cannot be a link (i.e., the InstanceSpecification may not be classified by a Relationship).

An InformationFlow’s sources and targets represent all potential instances typed or contained (i.e., owned) by them. For example, if a source or target is a