Editor: Richard H. Thayer ■ California State Univ., Sacramento ■ thayer@csus.edu
Technical Reviewer: Merlin Dorfman ■ Cisco Systems ■ dorfman@computer.org

Use Case Terminology

Geri Winters

actor: In UML, someone or something outside the system that interacts with the system.

alternate flow: The part of a use case that describes its alternative implementations. It's also used to describe error conditions, since errors can be considered a kind of alternative. *Also called alternate path.*

association: In UML, a relationship between an actor and a use case that indicates that the actor interacts with the system by means of the use case.

basic flow: The part of a use case that describes its most common implementation. The basic flow is written assuming that no errors or alternatives exist. *Also called basic path or happy day scenario.*

constraint: A semantic condition or restriction that describes a limitation or state. For example, a constraint might be a limitation on some data's range of values or on some behavior of the application, or it could be a description of a required state of the system at a particular point in time.

extend: In UML, a relationship from an extending use case to a base use case specifying how the behavior defined for the extending use case can be optionally inserted into the behavior defined for the base use case.

include: In UML, a relationship from a base use case to an included use case specifying how the behavior defined for the included use case can be inserted into the behavior defined for the base use case.

model: A semantically closed abstraction of a system or a complete description of a system from a particular perspective. Examples include use case, architecture, and domain models and code.

Object Management Group (OMG): An international standards organization that owns and maintains CORBA and UML standards.

postcondition: A constraint that must be true when a use case has ended.

precondition: A constraint that must be true when a use case is invoked.

relationship: A semantic connection between model elements. Examples include associations, dependencies, and generalizations. Relationships to use cases include **association**, **extend**, and **include**.

requirement: A condition or capability to which a system must conform. Requirements are either derived from user needs or stated in a contract, standard, specification, or other formally imposed document.

scenario: A description of a specific sequence of actions. In use case scenarios, specific persons or actor instances replace actors, and only one path is taken through the use case's possible basic and alternate flows. *Also called use case instance.*

system: A conceptual entity defined by its boundaries. Examples include companies, divisions, sets of software applications, components, machines, and devices.

Unified Modeling Language (UML): A graphical language for visualizing, specifying, constructing, and documenting an object-oriented software-intensive system's artifacts.

use case: In UML, a complete task of a system that provides a measurable result of value for an actor. More formally, a use case defines a set of use case instances or scenarios.

use case diagram: A UML diagram that shows actors, use cases, and their relationships.

use case model: A model that describes a system's functional requirements in terms of use cases.

use case specification: A document that describes a use case. A use case specification's fundamental parts are the use case name, brief description, precondition, basic flow, postcondition, and alternate flow.

Geri Winter is the president and CEO of Wyyzzk, Inc. Contact her at geri@wyyzzk.com.

0740-7459/05/$20.00 © 2005 IEEE