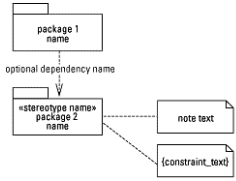


GENERAL-PURPOSE CONCEPTS

Can be used on various diagram types

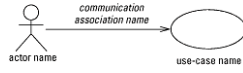
Package, dependency, note



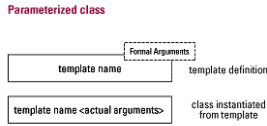
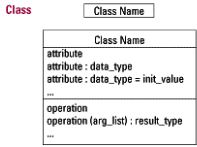
USE-CASE DIAGRAM

Shows the system's use cases and which actors interact with them

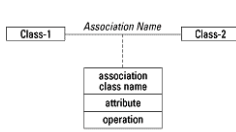
Actor, use case, and association



CLASS DIAGRAM Shows the existence of classes and their relationships in the logical view of a system



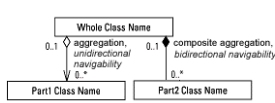
Association classes



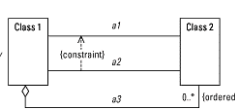
Role names and derived associations



Aggregation, navigability, and multiplicity



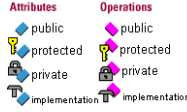
Constraints



Visibility and properties

Class	
-	private attribute
#	protected attribute
/-	private derived attribute
+S	class public attribute
+	public operation
#	protected operation
-	private operation
+S	class public operation

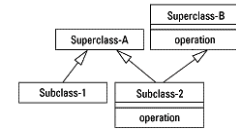
Optional visibility icons



Qualified association



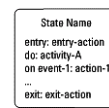
Generalization/specialization



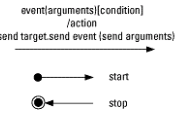
STATE-TRANSITION DIAGRAM

Shows the state space of a given context, the events that cause a transition from one state to another, and the actions that result

State icon

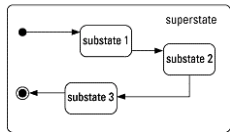


State transitions



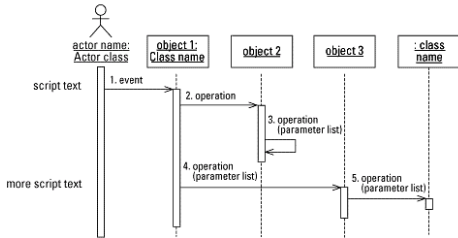
History

Nesting

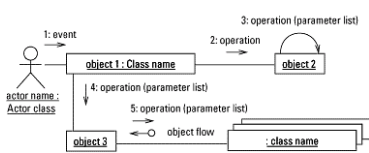


INTERACTION DIAGRAMS Show objects in the system and how they interact

Sequence diagram

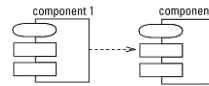


Collaboration diagram



COMPONENT DIAGRAM

Shows the dependencies between software components



DEPLOYMENT DIAGRAM

Shows the configuration of runtime processing elements

