UML Tutorial

What is UML 2?

UML 2 advances the successful UML specification, and is quickly becoming the accepted standard for specifying, documenting and visualizing software systems. The Unified Modeling Language (UML) is also used for the modeling of non-software systems, and is extensively implemented in most industry sectors including finance, military and engineering.

If you are new to the Unified Modeling Language, our Introduction to UML is a recommended starting point.

UML is divided into two general sets and includes fourteen basic diagram types:

1. Structural Modeling Diagrams

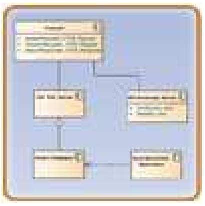
Structure diagrams define the static architecture of a model. They are used to model the 'things' that make up a model - the classes, objects, interfaces and physical components. In addition, they are used to model the relationships and dependencies between elements.

1. Package Diagrams

Package diagrams are used to divide the model into logical containers, or 'packages', and describe the interactions between them at a high level.

Learn More

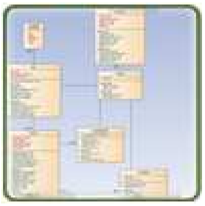
2. Component Diagrams



Component diagrams are used to model higher level or more complex structures, usually built up from one or more classes, and providing a well defined interface.

Learn More

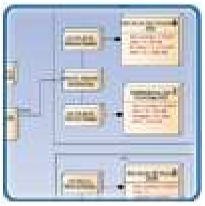
3. Class or Structural Diagrams



Class or Structural diagrams define the basic building blocks of a model: the types, classes and general materials used to construct a full model.

Learn More

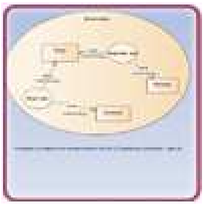
4. Deployment Diagrams



Deployment diagrams Deployment diagrams show the physical disposition of significant artifacts within a real-world setting.

Learn More

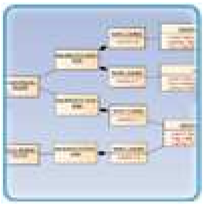
5. Composite Structure Diagrams



Composite Structure diagrams provide a means of layering an element's structure and focusing on inner detail, construction and relationships.

Learn More

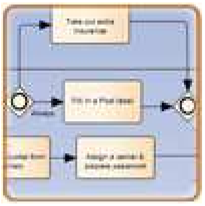
6. Object Diagrams



Object diagrams show how instances of structural elements are related and used at run-time.

Learn More

7. Profile Diagrams



Profile diagrams provide a visual way of defining light-weight extensions to the UML specification. UML Profiles are often used to define a group of constructs with domain-specific or platform-specific properties and constraints, which extend the underlying UML elements.

Learn More

2. Behavioral Modeling Diagrams

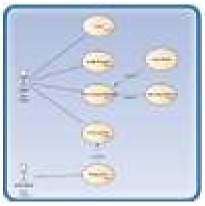
Behavior diagrams capture the varieties of interaction and instantaneous states within a model as it 'executes' over time; tracking how the system will act in a real-world environment, and observing the effects of an operation or event, including its results.

8. Use Case Diagrams

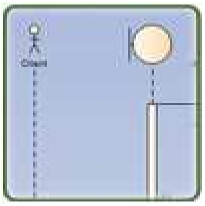
Use Case diagrams are used to model user/system interactions. They define behavior, requirements and constraints in the form of scripts or scenarios.

9. Sequence Diagrams

Sequence diagrams are closely related to communication diagrams and show the sequence of messages passed



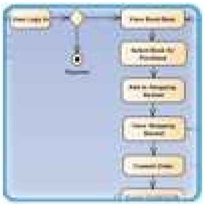
Learn More



between objects using a vertical timeline.

Learn More

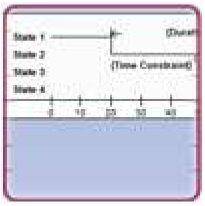
10. Activity Diagrams



Activity diagrams have a wide number of uses, from defining basic program flow, to capturing the decision points and actions within any generalized process.

Learn More

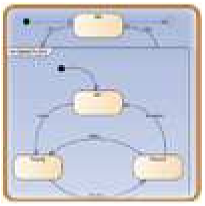
11. Timing Diagrams



Timing diagrams fuse sequence and state diagrams to provide a view of an object's state over time, and messages which modify that state.

Learn More

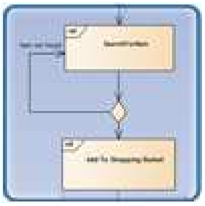
12. State Machine Diagrams



State Machine diagrams are essential to understanding the instant to instant condition, or "run state" of a model when it executes.

Learn More

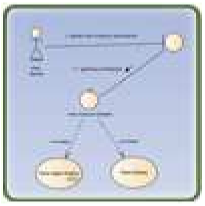
13. Interaction Overview Diagrams



Interaction Overview diagrams fuse activity and sequence diagrams to allow interaction fragments to be easily combined with decision points and flows.

Learn More

14. Communication Diagrams



Communication diagrams show the network, and sequence, of messages or communications between objects at run-time, during a collaboration instance.

Learn More

stay in touch

Products
Enterprise Architect
Pro Cloud Server
MDG Integration Eclipse, VS
MDG Technology UPDM, SysML
MDG Link DOORS

UML at a Glance
UML
UML Tools
PHP UML Modeling
Business Process Modeling
Model Driven Architecture

Solutions
Corporate
Government
Small/Medium Enterprise
IT Professionals
Trainers

Resources
UML 2.0 Tutorial
MDG Technologies
Corporate Resources
Developer Resources
Media Resources

Support
Online Manual
User Fo
Report
Feature
Compare Editions