

# Succeeding with the Unified Modeling Language (UML)

By Sinan Si Alhir (August 7, 1998)

Updated August 7, 1998

---

## Abstract

The Unified Modeling Language (UML) is a modeling language for specifying, visualizing, constructing, and documenting the artifacts of a system-intensive process. It was originally conceived by Rational Software Corporation and three of the most prominent methodologists in the information systems and technology industry, Grady Booch, James Rumbaugh, and Ivar Jacobson (the Three Amigos). The language has gained significant industry support from various organizations via the UML Partners Consortium and has been submitted to and approved by the Object Management Group (OMG) as a standard (November 17, 1997).

This paper elaborates on *succeeding* with the UML.

---

## Contents

- [Introduction](#)
  - [Mechanisms](#)
  - [Problems, Solutions, and Problem Solving](#)
  - [Problems and Solutions](#)
  - [Problem Solving](#)
  - [Conclusion](#)
  - [References](#)
- 

## Introduction

Engineering practices atop of the foundation of the Unified Modeling Language (UML) enable the UML to be applied successfully. However, it is these fundamental and critical practices that are often overlooked. Such practices are *identified* and *very briefly elaborated* herein.

[Return to Contents](#)

## Mechanisms

Mechanisms are practices for approaching modeling and diagramming that enable the creation of more precise and communicable models.

- *Perspectives (conceptualization, specification, and realization)*

- Enable clear viewpoints to be associated with diagrams.
- Are used to heighten the effectiveness of the communication.
- *Dichotomies* (*type–instance*, *specification–realization*, and *static–dynamic*)
  - Enable something to be viewed from multiple perspectives.
  - Are used to discover inconsistencies within models.
- *Layers* or *levels of abstraction* (*system level*, *subsystem levels*, *class levels*, and *method level*)
  - Enable attention and concentration to be focused regarding a subject (problem or solution).
  - Are used for organizing all of the diagrams pertaining to a single model into a coherent body of knowledge.
- *Extension mechanisms* (*stereotypes*, *properties*, *tagged values*, and *constraints*)
  - Enable the UML to be customized and extended.
  - Are used to ensure that the UML will evolve rather than be redefined to meet changing needs and requirements.

[Return to Contents](#)

## Problems, Solutions, and Problem Solving

The UML enables and promotes (but does *not* require nor mandate) a *use–case–driven*, *architecture–centric*, *iterative*, and *incremental* process that is *object oriented* and *component based*.

- *Use cases* are used to manage and provide focus for a problem–solving effort.
- *Architecture* is used to manage complexity and maintain integrity and focus as a solution to a problem evolves.
- *Iterations* and *increments* are used to repeatedly apply a process to evolve a solution to a problem.

[Return to Contents](#)

## Problems and Solutions

Within a problem–solving process, knowledge regarding a problem and solution is captured, organized around decisions, and depicted using the UML so that it can be communicated and leveraged. When deciding what diagram to use for communicating, consider the question or questions the communication is addressing, and what diagram or set of diagrams most effectively communicate the response. This decision centers on what dimensions of a model are to be emphasized in the response. Furthermore, each diagram type emphasizes different dimensions of a model.

[Return to Contents](#)

## Problem Solving

Within a problem–solving process, it is the underlying method that *suggests* how knowledge is utilized to realize a solution to a problem. This includes suggesting which *diagrams* to use and the *perspective* and the *level of abstraction* used to render and interpret these diagrams. Methods should be considered as suggestions and recommendations that organize and facilitate the problem–solving process rather than being considered rigid and inflexible rules that restrict the art of problem solving.

[Return to Contents](#)

## Conclusion

The **UML** is *not* a process, but enables processes. It is fundamentally critical that the *appropriate* process be utilized to solve a problem. And an appropriate process that utilizes the best engineering practices will provide the *potential* for success.

The success of the **UML** will be measured by its appropriate use on successful projects. The **UML** does not guarantee success, but enables fundamental engineering practices that are critical for success.

[Return to Contents](#)

---

## References

[Alhir]

Sinan Si Alhir. "**UML** in a Nutshell : A Desktop Quick Reference". O'Reilly Associates, Inc., 1998.

[Return to Contents](#)

---

Email [salhir@earthlink.net](mailto:salhir@earthlink.net)  
Web Site [home.earthlink.net/~salhir](http://home.earthlink.net/~salhir)  
Copyright © 1998 Sinan Si Alhir. All rights reserved.