



The importance of *Modeling*

Unified Modeling Language
as a standard language to analyze,
design and document software
intensive solutions

Why we model



A model is a simplification of reality. We build models so we can better understand the system we are developing

Aims achieved throughout modeling

- Model help us to visualize a system as it is or as we want it to be
- Models permit us to specify the structure or behavior of a system
- Models gives us a template that guides us in constructing a system
- Models document the decision we have made

Introducing the UML (1/2)

UML is a Language

A modeling language is a language whose vocabulary and rules focus on the conceptual and physical representation of a system.

UML is a Language for Visualizing

Some things are best modeled textually; others are best modeled graphically. In all interesting systems, there are structures that transcend what can be represented in a programming language.

UML is a Language for Specifying

Specifying means building models that are precise, unambiguous and complete. The UML addresses the specification of all the important analysis, design and implementation decisions.

The **UML** is a language for

- *Visualizing*
- *Specifying*
- *Constructing*
- *Documenting*

the artifact of a software-intensive system.

UML is a Language for Constructing

The UML is not a visual programming language, but its models can be directly connected to a variety of programming languages (Java, C++, ...)



Introducing the UML (2/2)

UML is a Language for Documenting

A software solution is made up by all sort of artifacts in addition to raw executable code. These artifacts include:

The **UML** is a language for

- *Visualizing*
- *Specifying*
- *Constructing*
- *Documenting*

the artifact of a software-intensive system.

- Requirements
- Architecture
- Design
- Source code
- Project Plans
- Tests
- Prototypes
- Releases



Benefits from UML

The **UML** is not limited to modeling software. It is expressive enough to model non software system, such as workflow in the legal system, the structure and behavior of a patient healthcare system and the design of hardware.

Who

System architect

Information System

Business modeler

Banking & Finance

Project manager

Telecommunications

Team leader

Transportation

Workflow designer

Defense/Aerospace

Data modeler

Retail

Application modeler

Medical electronic

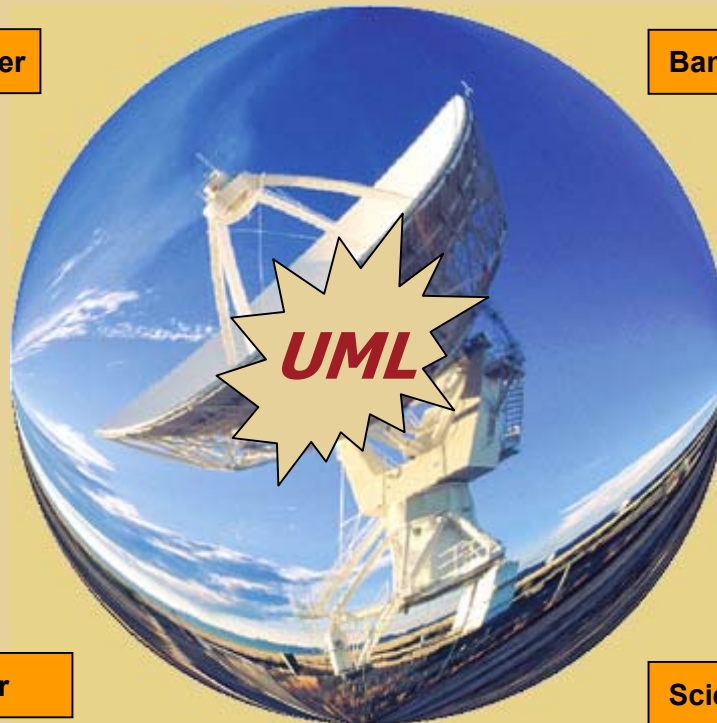
Process modeler

Scientific

System integrator

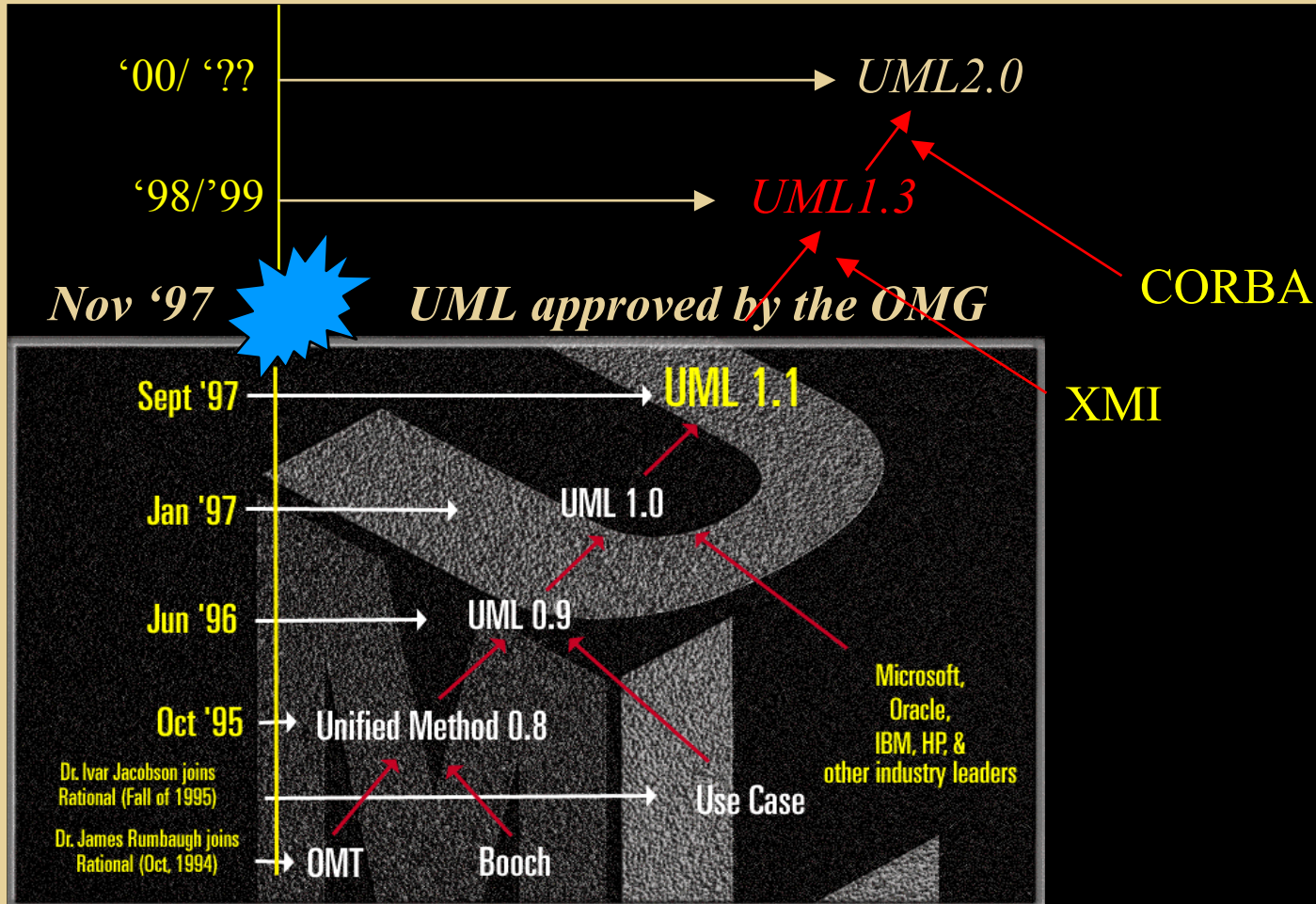
Distributed Web-based services

Where

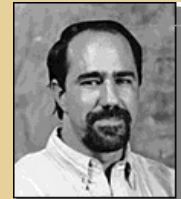


UML history

“The three amigos”



Grady Booch



Ivar Jacobson



James Rumbaugh



Available UML tools

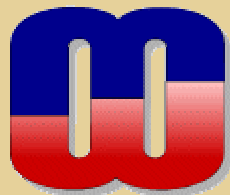


XUML
PRODUCTIVITY
PRAGSOFT CORPORATION
EXECUTABLE



Datasheets
▶ Real-time Modeler

Visual Object Modelers



MetaEdit+



VisualAge Smalltalk

UML Server



- ▶ Floating License Server for UMLStudio
- ▶ Repository Server for UMLStudio
- ▶ Automatic Version Control
- ▶ Visual Access Control



Copyright © 2000 PragSoft Corporation

cost effective error free software
SIMULATION
iUML



GDPRO 5.0

Rational the e-development company
Rational Rose 2000

contacts / shop / search

What next...



UML is a *language*, its is *not* a *method* nor a *developing tool*.

After the formalization of the solution, each developer can decide to implement the solution using any available tool: *JDeveloper, JBuilder, Visual Cafè, Jamba, Visual C++*, etc...

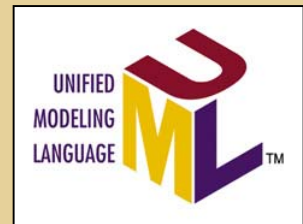
Integrated tools allow to design and implement within the same suite.

The entire picture

PwC Methods



Language



Project programming

Project planning

Analysing

Project delivery

Defining

Designing

Testing

Building & Implementing

Implementation tools



Design tools

