Semantik der UML 2.0
What is UML

The *Unified Model Language* is an industry standard language for

- visualizing,
- specifying,
- constructing, and
- documenting

software-intensive systems. It can be used with many processes, throughout the development lifecycle, and across different implementation technologies.
Why we build models

In order to

- understand complex systems,
- minimize engineering risk,
- communicate with stakeholders,
- drive the implementation.
Characteristics of useful models

A models should be

- abstract,
- understandable,
- accurate,
- predictive,
- inexpensive.

Most software models of the past failed on one or more of these aspects.
UML History

near completion

March’03  UML 1.5
May’01  UML 1.4
June’99  UML 1.3
OMG acceptance, Nov’97  UML 1.1
UML partners, Jan’97  UML 1.0
June’96  UML 0.9
OOPSLA’95  Unified Method 0.8
Booch method  OMT  OOSE  other methods

UML partners, Jan’97
UML 1.x: What went wrong?

- MDD potential of models not fully exploited
- Inadequate modeling capabilities
- Too complex
- Inadequate semantics definition
- No diagram interchange capability
- No fully alignment with MOF
Formal RFP requirements

1. Infrastructure - UML internals
   more precise conceptual base for better MDA support

2. Superstructure - user-level features
   new capabilities for large-scale software systems
   consolidation of existing features

3. OCL - constraint language
   full conceptual alignment with UML

4. Diagram interchange standard
   for exchanging graphic information
Superstructure requirements

The language has been restructured and modularized.

There have been significant changes to the UML metamodel.
Two main specifications

UML 2.0

UML 2.0 Superstructure

UML 2.0 Infrastructure

OCL

Diagram
Inter−change
**Language Structure**

A core language plus a set of optional “sub-languages”

<table>
<thead>
<tr>
<th>MOF</th>
<th>Profiles</th>
<th>OCL</th>
<th>State machines</th>
<th>Structured classes and components</th>
<th>Activities</th>
<th>Interactions</th>
<th>Actions</th>
<th>Flows</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Basic UML
(Classes, Basic behavior, Internal structure, Use cases, . . .)

UML Infrastructure
UML state machines

Semantik der UML WS04/05 11 20. Juli 2004
UML class diagrams

{self.boss->isEmpty() or
self.employer = self.boss.employer}
UML activities

![UML diagram showing activities]

- Modify Proposal
- Review Proposal
- Notify of Modification
- Publish Proposal
- Notify of Rejection
UML interactions

sd CriticalRegion

:Emergency  :Operator  :Caller  :Callee

par

call(100)
call(100)
call(101)
call(101)
call(911)

critical

call(911)
call(911)
call(911)

Semantik der UML WS04/05 14 20. Juli 2004
UML actions
UML flows

Customer <-> Company

Product <<flow>>

Employee

Wage <<flow>>
Goal of the seminar

Study of

- foundations,
- concepts,
- syntax and semantics, and
- implementation

of the sublanguages of basic UML put in context.