Using UML 2.0 in Real-Time Development

Kirsten Berkenkötter

University of Bremen
Introduction

Weaknesses of UML 1.4

New Features of UML 2.0

Weaknesses Reviewed

Conclusion
Introduction

} UML 1.4
  common standard
  basics are easy to learn
  useful for communication, specification, documentation

} structural modeling
  class diagrams
  object diagrams
  component diagrams
  deployment diagrams
Introduction

} behavioral modeling
  use case diagrams
  collaboration diagrams and sequence diagrams
  statechart diagrams
  activity diagrams

} real-time development
  hardware-software mutual dependencies
  high reliability
  often safety-critical background
  high demands on a modeling language
} Introduction
} Weaknesses of UML 1.4
} New Features of UML 2.0
} Weaknesses Reviewed
} Conclusion
General Weaknesses of UML 1.4

} UML specification
    informal
    sometimes ambiguous

} metamodel
    4-layer metamodeling approach not followed

} usability
    overwhelming number of diagrams and elements
General Weaknesses of UML 1.4

- diagrams and views
  - inconsistent
  - no mapping between different diagrams in a model

- composition of models
  - no hierarchy
  - insufficient for large models
Real-Time Dependent Weaknesses of UML 1.4

- hardware-software interdependencies
  - no sufficient support

- timing constraints
  - no syntax and semantics for timing purposes

- communication
  - no communication structures like ports, connectors, protocols
Introduction

Weaknesses of UML 1.4

New Features of UML 2.0

Weaknesses Reviewed

Conclusion
New Features of UML 2.0

} general

different specification documents
division of language core and modeling elements
4-layer metamodeling approach realized

} profiles

deriving new elements from metamodel
introducing new terminology, new syntax, new semantics and constraints, and further information like transformation rules
New Features of UML 2.0

} structural modeling

hierarchical modeling in composite structure diagrams
communication structures with ports and connectors
components as software components with internal structure and interfaces to the outside
New Features of UML 2.0
New Features of UML 2.0
New Features of UML 2.0

} behavioral modeling

fine-grained action model
independently defined activity diagrams with new features like interruptible regions and loops
simple time model and timing diagrams
interactions with new features like alternatives, options, breaks, loops, critical regions
behavioral and protocol state machines
New Features of UML 2.0
New Features of UML 2.0

sd payDrink

: paymentController : machineController

alt
[coinValue == 50]

coin(50) →

| fiftyCent() |

| else |

coin(1) →

| oneEuro() |
New Features of UML 2.0

} structural modeling
  class diagrams
  object diagrams
  package diagrams
  component diagrams
  composite structure diagrams
  deployment diagrams

} behavioral modeling
  use case diagrams
  sequence diagrams
  communication diagrams
  activity diagrams
  interaction overview diagrams
  statechart diagrams
  timing diagrams
Introduction

- Weaknesses of UML 1.4
- New Features of UML 2.0
- Weaknesses Reviewed
- Conclusion
Review

General Weaknesses

} UML specification
   more fine-grained
   still informal

} metamodel
   4-layer metamodelling approach realized
   extension mechanism based on metamodel (profiles)
Review

} usability
   even more diagrams and elements added
      even worse

} diagrams and views
   no change

} composition of models
   improved by hierarchical modeling with composite structure diagrams
Review

Real-Time Dependent Weaknesses

- hardware-software interdependencies
- deployment diagrams still insufficient
- modeling of hardware as components with ports as access points possible
Review

} timing constraints
  timing model lacks important features:
  ▪ no discrete and dense time
  ▪ no timing zones
  ▪ no system time
  ▪ no synchronization between clocks

} communication
  ports and connectors
  protocol state machines
Introduction

Weaknesses of UML 1.4

New Features of UML 2.0

Weaknesses Reviewed

Conclusion
Conclusion

} improvements
  - hierarchical modeling
  - communication structures
  - profiles

} problems
  - informal specification ➔ no formal reasoning
  - usability

璀 profiles best solution for real-time development