Università della Svizzera italiana | Faculty of Informatics

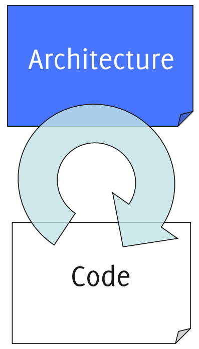
Software Architecture Modeling

Prof. Cesare Pautasso
http://www.pautasso.info
cesare.pautasso@usi.ch
@pautasso

Capturing the Architecture

Grady Booch

- Every system has an architecture
- Some architectures are manifest and visible, many others are not
- A system's (descriptive) architecture ultimately resides in its executable code
- Before a system is built, its (prescriptive) architecture should be made explicit
- A system's architecture may be **visualized and represented using models** that are somehow related to the code (existing or yet to be written)

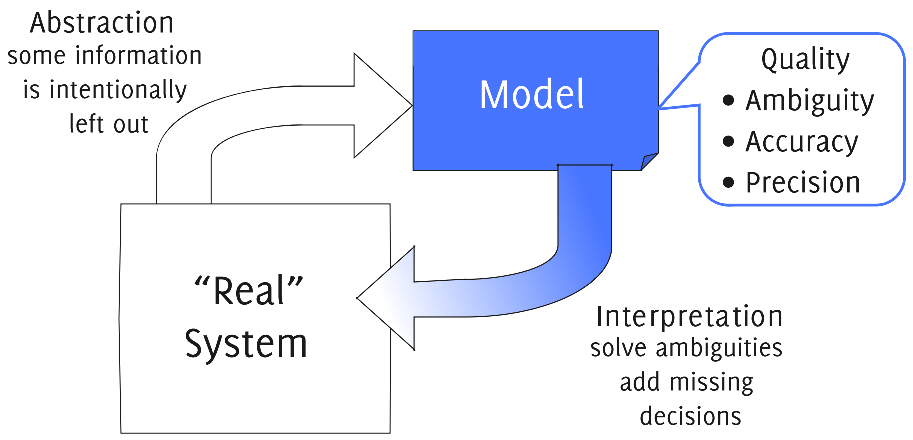


What is modeling?

Richard Taylor

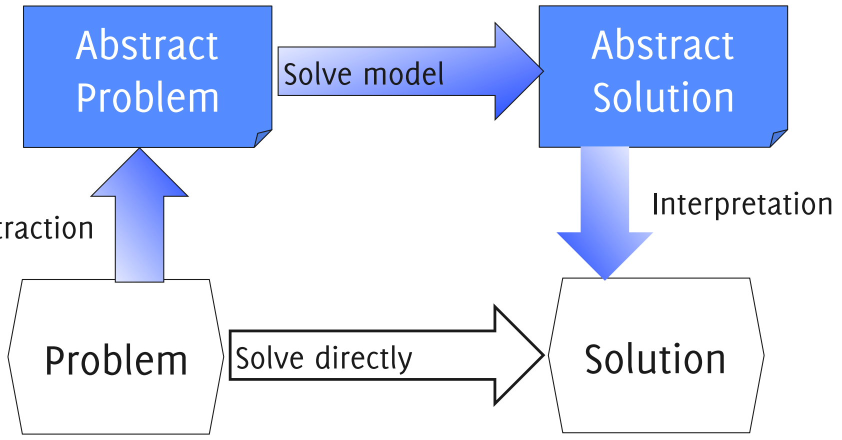
- An architectural **model** is an artifact that captures some or all of the design decisions that comprise a system's architecture.
- Architectural **modeling** is the reification and documentation of those design decisions.

Abstraction and Interpretation



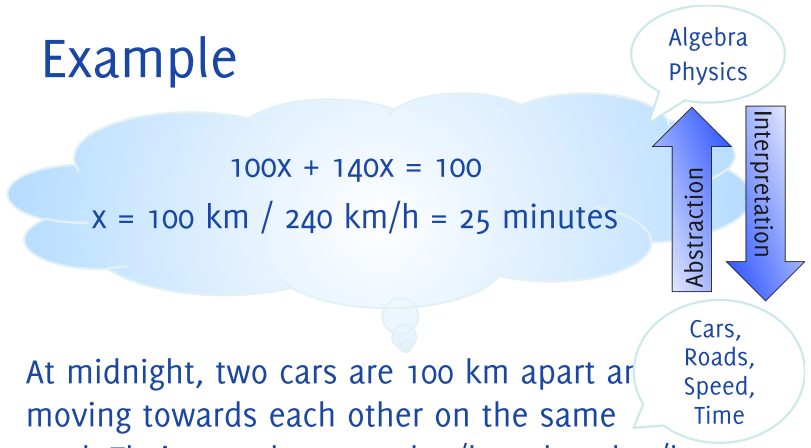
- The architecture models only some interesting aspects of a system.

Solving Problems with Models



- Abstract models help to find solutions to difficult engineering problems.

Example



At midnight, two cars are 100 km apart ar
moving towards each other on the same
road. Their speeds are 100km/h and 140km/h.
When will they pass each other?

Model Quality

- Ambiguity
  - A model is ambiguous if it leads to more than one interpretation
  - Incomplete models can be ambiguous: different people will fill in the gaps in different ways.
- Accuracy
  - A model is accurate if it is correct, conforms to fact, or deviates from correctness within acceptable limits.
- Precision
  - A model is precise if it is sharply exact or delimited.

Model Quality - Advice

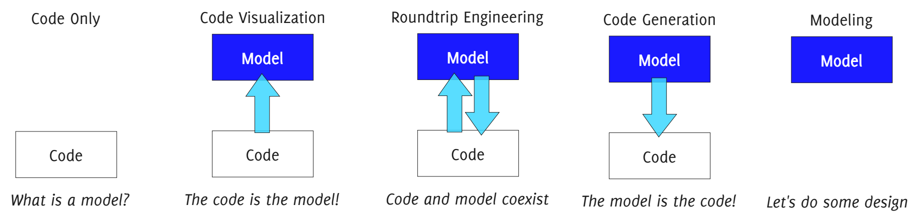
- Make sure your architecture is accurate
  (a wrong, inconsistent or conflicting architectural decision is a recipe for disaster)
- Sometimes you can even make it complete
  (but it will be more expensive, so only do it for critical aspects of the system)
- Precision helps, but avoid over-specifying and over-designing the architecture, especially if the architecture is inaccurate, adding details will not fix it. (developers may be trusted to add missing details)

Why Modeling?

- Record decisions
  - Document the architecture and its rationale
  - Which decision? Why the decision?
- Communicate decisions
  - Visualize the architecture
  - Different roles have different perspectives
- Evaluate decisions
  - What is a good model?
  - What is a good decision?
  - Detect problems early
- Evolve decisions
  - Give constraints and a clear path to change the architecture
- Generate artifacts
  - Drive the development from very precise models

Model-Driven Architecture

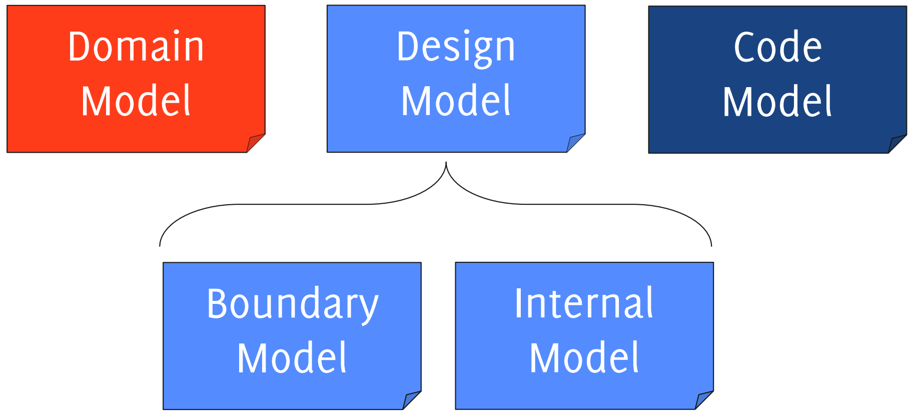
- MDA promotes modeling as the main software design and development activity
- The design is organized around a set of models and model transformations to move within and between different abstraction layers (e.g., code generation, roundtrip engineering, reverse engineering)



What to model?

- Static Architecture:
  - Structural Decomposition
  - Interfaces
  - Components
  - Connectors
  - Mapping to Code Artifacts
- Dynamic Architecture:
  - Behavior
  - Deployment
  - Mapping to Hardware
- Design Process:
  - Rationale
  - Stylistic Constraints
  - Dependencies
  - Team Organization
  - Legal Constraints
- Quality:
  - Attributes
  - Trade-offs
  - Testing (Q&A) Plan

Canonical Models



Domain Model

- Refutable truths about the real-world
- Outside your control
- Your system will be evaluated against it

Code Model

- Complete set of design commitments
- Represent the executable, ready-to-run "source code"

Boundary and Internal Design Models

- The visible interfaces of the system architecture...
- ..refined with details about the implementation

		Design Model	
Fairbanks	**Domain Model**	**Boundary**	**Internals**
Bosch		System context	Component Design
D Souza	Business Concept	Blackbox	Whitebox
Jackson	Domain	Domain+Machine	Machine
RUP	Business Modeling	Requirements	Analysis & Design

Example Domain Model

- Music songs are organized in albums
- The same song can be authored by many artists
- Listening to each song costs 0.99 CHF, but short samples can be heard for free
- Songs can be downloaded and also live streamed
- Songs are stored in files of standard MP3 format
- Files contain embedded metadata and watermarks
- A music player can carry 10’000+ albums

References

- Michael Jackson, Problem Frames: Analyzing and structuring software development problems, Addison-Wesley, 2001
- Richard N. Taylor, Nenad Medvidovic, Eric M. Dashofy, Software Architecture: Foundations, Theory and Practice, John-Wiley, January 2009
- Philippe Kruchten, Architectural Blueprints—The “4+1” View Model of Software Architecture, IEEE Software 12 (6). November 1995, pp. 42-50
- Scott W. Ambler, Agile Modeling (http://www.agilemodeling.com/)
- I. Asimov, The Relativity of Wrong, The Skeptical Inquirer, Fall 1989, Vol. 14, No. 1, Pp. 35-44