Empirical Evidence on Modeling in Software Development

Dr. Michel R.V. Chaudron
Associate Professor
Leiden Institute of Advanced Computer Science
chaudron@liacs.nl

Outline of talk
- Introduction
- Why care about of modeling:
  - Does modeling actually help improve software development?
  - How to do quality assurance in model-based software development?
    - State-of-the-art
- Conclusions & Future Directions
Introduction: Research Interest

- Effectiveness of software modelling
  - Quality of Modeling
    When is a model good (enough)
  - Analysis of Non-Functional Quality Properties of architectures

- What are the pay-offs of investing in early design/architecture?
  Fewer defects?
  Cheaper maintenance? ...

- Many modeling approaches around;
  focus on UML in custom software development

Software Engineering team at Leiden

Werner Heijstek
Models in Global SE

Hafeez Osman
Updating UML

Dave Stik
Morten
Teaching SW Design and UML in maintenance

Ana Fernandes
Modeling and Quality

Pieter Kwantes
Domain Specific Modeling

Ramin Etemadi
Optimizing Sw. Arch. Design

Bilal Karasneh
Reasoning about Sw. Arch. Design

Peng Ye

Javier Hernandez
Modeling Sw-Product Lines

Dr. Michel R.V. Chaudron
chaudron@liacs.nl
Why you should care about Modeling

Developer Benefits
- Better Understanding Problem Domain
- Improved Communication
- Better Risk management
- More Accurate Estimating

Process Benefits
- Improved requirements
- Improved Design Compliance
- Improved Design Quality
- Shared System Model

Product Benefits
- Fewer Defects
- Reduced rework
- More efficient Testing
- More efficient Maintenance
- Reduced testing effort
- Reduced maintenance effort

Improve Quality

Improved Productivity

Empirical Research

... is a way of gaining knowledge by means of observation or experience.

Theory, SE body of knowledge

Practice: SE projects

People, Methods, Artefacts

Hypothesis

Observation

Prediction/Test

Validation

Methods: Experiment, Case study, Survey, ...

Both quantitative and qualitative
Empirical Life-cycle

Initial Idea → Exploratory Interviews → Survey → Method Development

Experiment → Industrial Case Studies

Characteristics of Case Studies

<table>
<thead>
<tr>
<th>Case study</th>
<th>No. of classes</th>
<th>No. of person-years spent on modeling</th>
<th>No. of team members</th>
<th>CMM level (estimated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>734</td>
<td>15</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>168</td>
<td>20</td>
<td>20</td>
<td>2-3</td>
</tr>
<tr>
<td>C</td>
<td>108</td>
<td>20</td>
<td>10</td>
<td>2-3</td>
</tr>
<tr>
<td>D</td>
<td>716</td>
<td>Unknown</td>
<td>Unknown</td>
<td>1</td>
</tr>
<tr>
<td>E</td>
<td>443</td>
<td>10</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>F</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>G</td>
<td>75</td>
<td>10</td>
<td>Unknown</td>
<td>1</td>
</tr>
<tr>
<td>H</td>
<td>478</td>
<td>Unknown</td>
<td>Unknown</td>
<td>1</td>
</tr>
<tr>
<td>I</td>
<td>705</td>
<td>12</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>J</td>
<td>51</td>
<td>12</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>K</td>
<td>14</td>
<td>1.5</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>L</td>
<td>46</td>
<td>1.5</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>M</td>
<td>73</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>N</td>
<td>359</td>
<td>5</td>
<td>5</td>
<td>1-2</td>
</tr>
</tbody>
</table>

Your project here? Mail me: chaudron@liacs.nl
Industrial use of UML diagrams

<table>
<thead>
<tr>
<th>Diagram Type</th>
<th>Usage Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class Diagram</td>
<td></td>
</tr>
<tr>
<td>Use Case Diagram</td>
<td></td>
</tr>
<tr>
<td>Sequence Diagram</td>
<td></td>
</tr>
<tr>
<td>Use Case Narrative</td>
<td></td>
</tr>
<tr>
<td>Activity Diagram</td>
<td></td>
</tr>
<tr>
<td>Statechart Diagram</td>
<td></td>
</tr>
<tr>
<td>Collaboration Diagram</td>
<td></td>
</tr>
</tbody>
</table>

Styles of Modeling

- Sketch
- Communication
- Recipe for construction
Modelling style and model purpose

Styles of using UML
- as a sketch – thinking tool/understanding
- for communicating system design
- as a blueprint – guide the implementation work
- as a implementation (MDA) - code generation

Uses of Software Models

+ Understanding
+ Analyzing / Predicting
+ Communicating
+ Guiding
UML in practice

Communication

More effort $\Rightarrow$ More expensive

Recipe for construction

Modeling & Documentation in Agile Development

- Agile principles:
  
  *working software over comprehensive documentation*

Survey under 75+ agile developers

Tending towards:

we need a bit more

Modeling is compatible with agile development
Does use of UML improve software quality?

A large number of developers indicated the use of UML improves **understandability** and **modularity**.

**Economic Model for Cost of Quality**

Cost of SQ = Achievement Cost + Non-conformance Cost
Cost of Modeling

Focus on effort

“Is there a correlation between class-count and the effort spent in modeling?”

Sources:
- Experiment
- Set of industrial case studies

Empirical Data

- Experiment
  - 106 MSc students in (TU/e)
    - Organized in 35 teams
  - Task: Model Car Navigation System
  - Duration: 6 weeks assignment
  - Three treatments were applied:
    1. No modeling conventions
    2. With modeling conventions
    3. Tool-supported modeling conventions
There is no significant correlation between class-count and the effort spent in modeling.

→ class-count is not a good measure for the effort spent on modeling?
Impact of MDD on Effort Distribution

as % of total project effort

Does Quality of Modeling Matter?
An Industrial Case Study

Focus on detail in a UML Model

Low detail

High detail
How is Level of Detail distributed in a diagram?

Case 1

Case 2

Relation between UML-LoD and Code Quality

Select ‘defects’ in defect DB

Find classes in source code that were repaired for solving this defects

Find corresponding classes in UML models

Determine LoD for CD and SD

Dr. Michel R.V. Chaudron, IPA Spring Days, April 2012, Garderen

Leiden University. The university to discover.
Level of Detail for Sequence Diagrams is significantly (negatively) correlated with defect density. More detailed model => fewer defects.

- **Quality Assurance for UML Modelling**
  - How to define Quality for modeling?
  - How to assess quality?
  - How to check conformance in code?
Quality Models

Existing models

- Decomposition of characteristics
  - Bottom level: metrics
- Differences in
  - Relations between characteristics
  - Vocabulary

Boehm’s Quality Model

How to Define UML Model Quality?
The quality attributes of a UML model is determined by its purpose...

<table>
<thead>
<tr>
<th>Primary Use</th>
<th>Purpose</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintenance</td>
<td>Modification</td>
<td>Complexity, Balance</td>
</tr>
<tr>
<td></td>
<td>Testing</td>
<td>Modularity, Communicateness, Correspondence</td>
</tr>
<tr>
<td></td>
<td>Comprehension</td>
<td>Self-descriptiveness, Conciseness</td>
</tr>
<tr>
<td></td>
<td>Communication</td>
<td>Precision, Esthetics</td>
</tr>
<tr>
<td></td>
<td>Analysis</td>
<td>Detailedness, Consistency, Completeness</td>
</tr>
<tr>
<td>Development</td>
<td>Prediction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Implementation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Code Generation</td>
<td></td>
</tr>
</tbody>
</table>
How to measure goodness of UML model?

**Design Heuristics**
- Coupling, Cohesion, absence of anti-patterns

**Completeness**
- Interaction of classes must be described in SD
- Methods of CD must be called in SD

**Consistency**
- In many cases it can not be automatically determined whether a flaw is an incompleteness or an inconsistency

**Correspondence**
- Dependencies in the implementation are allowed only if they occur in the design

---

**Design Heuristics for UML**

- Metrics based on Chidamber & Kemerer OO metrics

  *Maintainability* relates to *Coupling, Cohesion, ...*

---

Highest coupling = 7

Highest coupling = 4
Automated Model Checking and Visualization

UML model

UML Analysis Tool

Quality Metrics/Rules
- Completeness
- Consistency

Visualization of model + metrics

Screenshot of the MetricView tool

Quality Metrics
- Completeness
- Consistency
- ... extensible

Quality = absence of ‘bad things’ such as omissions, spaghetti

= violation

= ok
Distribution of Coupling

System of several hundred classes

MetricView Tool

MetaView to visualize the (hidden) inter-diagram relations

Example: Which classes contribute to a certain use case?

http://www.youtube.com/watch?v=G3HJ QR9EG4
MetricView Tool
http://www.youtube.com/watch?v=G3HJ_QR9EG4

The values of metrics are visualized on class diagrams using colors.
Example: Coupling-Between-Objects (CBO)

We will analyse your UML model: mail to Chaudron@liacs.nl

3D visualization of design measurements
Your cases welcome...

http://www.youtube.com/watch?v=G3HJ_QR9EG4
Findings from State of the Practice

UML is used in different ways for different purposes
- Incompleteness
  - Modelers focus on complex and critical parts
- Disproportion
  - Different parts of the system are modeled different in level of quality
- Inconsistency
  - Conflicting information in different views of one model

Quality Assurance for modeling should address these

Industrial Experience with MetricView as Quality Assurance Tool

- Based on 15+ industrial projects

- If there are weak spots in the design, then these are indicated as ‘suspects’ by MetricView

- MetricView indicates many weak spots that do not require improvement according to project architects (false positives?)
  \[\Rightarrow\] prioritization is needed

- The later MetricView is applied, the fewer ‘weak spots’ are removed from the design \[\Rightarrow\] process issue
## Comparing “MDD with code generation” & “Modeling + Programming”

<table>
<thead>
<tr>
<th>MDD with code generation</th>
<th>Modeling with Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requires:</td>
<td></td>
</tr>
<tr>
<td>- stable domain</td>
<td>Off the shelves tooling</td>
</tr>
<tr>
<td>- very good understanding of domain</td>
<td>Standard language (staff hiring &amp; turnover)</td>
</tr>
<tr>
<td>- Multiple projects</td>
<td>Divergence of model and code</td>
</tr>
<tr>
<td>- Don’t introduce in critical projects</td>
<td></td>
</tr>
</tbody>
</table>

Needs dedicated techniques for Quality Assurance, estimation

---

## Future Directions: Modelling and V&V

- Automated Reasoning about Designs
  - Diagnosis & Prioritization
  - Severity Assessment of defects

- Embedding Correctness in
  - Domain Specific Languages
  - Code-generation / Model Transformations
**Concluding Remarks**

- Modeling is becoming more and more common in software development. There are payoffs for modeling in:
  - improved code quality
  - improved communication
- Empirical studies enhance alignment of research and practice
- Practical Quality Assurance for UML should use low hanging fruit:
  i) Naming and layout-conventions, ii) Reviews & Inspections
  iii) Versioning, iv) Metrics

---

**contact**

**Dr. Michel R.V. Chaudron**  
Associate Professor  
Leiden Institute of Advanced Computer Science  
chaudron@liacs.nl  
www.liacs.nl/~chaudron

If you have UML models you would like to assess?  
If you want advice on Quality Assurance for modeling?  
If you want to sponsor our research?
How often do design issues lead to problems?

Incompleteness of models was considered most often to lead to implementation problems.

Developers apply more detail on *critical* and *complex* parts of a system.
Which factor most often drives developers to non-correspondence?

Among other factors, developers considered **incomplete model** more often drives them to **non-correspondence**.

Does strictness of implementing a model differ for different modeling constructs?

Developers favor to implement **dependency** and **inheritance** relations more strictly than other constructs.
Which classes are modeled?

Complexity and Coupling is higher for classes that are modeled than not modeled

Does Quality Matter? Experiment

- Subjects are given UML models
- Ask questions about UML model
  - Two types
    - Which implementation matches the diagrams?
    - How do you interpret these diagrams?
  - Answer from the perspective of the developer.
- Multiple-choice test (15 Questions)
  - 4 options + 1 option
    - “There’s something wrong, I can’t give an answer”
  - + Background questions
    (possible confounding factors: training, experience, …)
Example Question

It is your task to implement class ATM given these two UML diagrams?

A)  
```
getCardInserted()
  c.requestPIN();
  dosomething;
  a.open()
```

B)  
```
getCardInserted()
  c.requestPIN();
  dosomething;
  a.lock()
```

C)  
```
getCardInserted()
  c.requestPIN();
  dosomething;
  a.acknowledge()
```

D)  
```
getCardInserted()
  c.requestPIN();
  dosomething;
  a.validate()
```

E)  
```
Ring the bell!
Something is wrong!
```

64% don't see the defect and just implement

Opinions on Value of Modelling

- Agile
- MDA
- "Loose UML"
- Formal Methods

100% completeness
Maintenance paradox

- Maintainers of software would like better documentation

- Even if UML models are created during design

  Too detailed models become outdated

  Better abstraction in reverse- / round trip-engineering is needed in the setting of incremental & iterative development

Practical QA for UML modeling

Naming and layout-conventions

*The Elements of UML2.0 Style*,
Scott Ambler, Cambridge Univ Press, 2005

Reviews & Inspections

Guidelines by e.g. Shull et.al., Biffl,

Version Management

Many tools around (e.g. CVS, SVN, ...)

Metrics

SDMetrics  http://www.sdmetrics.com/
Investigations into effectiveness of software modeling

MetricView contributors

- Johan Muskens
- Christian Lange
- Martijn Wijns
- Dennis van Opzeeland
- Ariadi Nugroho
- Werner Heijstek
- Maurice Vermeer
- Alex Telea
- Robin van den Broek

Publications

- **Journal Publications**
  - Supporting task-oriented modeling using interactive UML views. [PDF](#)
  - In Practice: UML Software Architecture and Design Description.

- **Conference Publications**
  - A Visualization Framework for Task-Oriented Modeling using UML.
  - An Experimental Investigation of UML Modeling Conventions.
  - Effects of Defects in UML Models – An Experimental Investigation.
  - Improving the Quality of UML Models in Practice.
  - Visual Exploration of Combined Architectural and Metric Information.
  - Combining Metrics Data and the Structure of UML Models using GIS Visualisation Approaches.
  - An Exploratory Study on the Industrial Use of UML: Improving Control over Design Quality.
  - Investigations in Applying Metrics to Multi-View Architecture Models.
  - Konsistenz und Vollständigkeit industrieller UML Modelle.

- **Workshop Publications**
  - Towards Task-Oriented Modeling using UML.
  - Managing Model Quality in UML-based Software development.
  - Quantitative Techniques for the Assessment of Correspondence between UML Designs and Implementations.
Statistics about Design Doc’s in general

Typical Word document mixes:
- Text and diagrams
  500-800 words / diagram

- Formal UML & inventions
  40-50% of diagrams is UML
  60-50% of diagrams is not UML

Text is used for:
- Design decisions / Rationale
- Elaboration/explanation
- Linking to context

‘Design’ Tooling must support: models, sketches & text

Dr. Michel R.V. Chaudron, IPA Spring Days, April 2012, Garderen

Uri Dekel, James D. Herbsleb: Notation and representation in collaborative object-oriented design: an observational study. OOPSLA 2007: 261-280