Model-Driven Development: Concepts, Trends and Standardization

Stephen J. Mellor
Freeter
## A Brief History of Modeling

<table>
<thead>
<tr>
<th>Year</th>
<th>Description</th>
<th>Authors/Editors</th>
</tr>
</thead>
<tbody>
<tr>
<td>1979</td>
<td>Structured Design</td>
<td>Yourdon and Constantine</td>
</tr>
<tr>
<td>1981</td>
<td>Structured Analysis</td>
<td>De Marco</td>
</tr>
<tr>
<td>1985</td>
<td>Structured Devpt/RT</td>
<td>Ward and Mellor</td>
</tr>
<tr>
<td>1988</td>
<td>OOA</td>
<td>Shlaer and Mellor</td>
</tr>
<tr>
<td>1988</td>
<td>OO Design</td>
<td>Booch</td>
</tr>
<tr>
<td>1992</td>
<td>OMT</td>
<td>Rumbaugh et al</td>
</tr>
<tr>
<td>1997</td>
<td>UML 1.1</td>
<td>Three Amigos</td>
</tr>
<tr>
<td>2002</td>
<td>Executable UML</td>
<td>Mellor and Balcer</td>
</tr>
<tr>
<td>2004</td>
<td>UML 2.0</td>
<td>Cast of thousands</td>
</tr>
</tbody>
</table>
Differing Expectations

Modeling formalism should mirror the implementation.

Model should mirror my mind

Modeling formalism should mirror the knowledge we’re capturing.
UML Design Decisions

- Can’t satisfy all constituencies
- Define each diagram
- Do *not* define how they fit together
Executable Translatable UML
Customer X

- Customer X has been using xtUML and BridgePoint on pilot projects.
- They measured memory requirements, processor speed etc. and calculated the cost differential between generated and hand coding.
- They used Logiscope to get a measure of code in terms of Maintainability, Testability, Stability, Changeability and Analyzability.
# Results from Pilot Projects

*(results from projects where high performance was essential)*

|--------------------|-------------------------------------------------------------------------------------------------------------------------|
| Prop. language     | - 0.8% higher data memory cost  
                    - 5.2% lower program memory cost  
                    - worst case 6.4% higher cpu cost |
| C++                | - no additional data memory cost (0%)  
                    - no additional program memory cost (~0%)  
                    - cpu cost not verified yet |
| Firmware (C)       | - no additional data memory cost (0%)  
                    - 2.4% higher program memory cost  
                    - 0.7% lower cpu cost |

*No time spent on optimizing the generated code*
## Logiscope Code Quality Results

<table>
<thead>
<tr>
<th>Functions possible to grade (% of total)</th>
<th>Legacy (51%)</th>
<th>Generated (86%)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Maintainability</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Excellent</td>
<td>58%</td>
<td>Excellent</td>
</tr>
<tr>
<td>Good</td>
<td>37%</td>
<td>Good</td>
</tr>
<tr>
<td><strong>Testability</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Excellent</td>
<td>86%</td>
<td>Excellent</td>
</tr>
<tr>
<td>Good</td>
<td>10%</td>
<td>Good</td>
</tr>
<tr>
<td><strong>Stability</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Excellent</td>
<td>65%</td>
<td>Excellent</td>
</tr>
<tr>
<td>Good</td>
<td>18%</td>
<td>Good</td>
</tr>
<tr>
<td><strong>Changeability</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Excellent</td>
<td>57%</td>
<td>Excellent</td>
</tr>
<tr>
<td>Good</td>
<td>36%</td>
<td>Good</td>
</tr>
<tr>
<td><strong>Analyzability</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Excellent</td>
<td>56%</td>
<td>Excellent</td>
</tr>
<tr>
<td>Good</td>
<td>18%</td>
<td>Good</td>
</tr>
</tbody>
</table>

- Legacy application has 394 functions
- 193 of these not possible to grade
- Application has 215 functions
- 29 of these not possible to grade
Ricoh

- Printers, office equipment
- C
- Unknown RTOS
- Renesas and Mitsubishi
- 128k
Tellabs

- Communications switching (internet/telephony)
- C++
- pSOS, Solaris
- PowerPC (over 1000 controllers)
- >1 MB per controller card
Saab

- Military systems
- Ada
- Classified
- Classified
- Classified
Bioanalytical Systems

- Laboratory equipment
- C
- No RTOS
- 8051
- 32k ROM + 6kb RAM
Ericsson

- Cellular telephony, base stations, wireless
- C, C++
- Various
- ARM
- Various
St Jude Medical

- Defibrillator, pacemaker
- Assembly, C, C++
- No RTOS
- Zilog Z8 series
- Unknown
Delphi

- Heating Ventilation Air Conditioning
- C
- No RTOS
- Unknown
- <64k
Visa

- Credit Card Authorization
- C++
- Solaris
- Sparc
- 4GB

Well over 10,000 Trx/Sec (actual number proprietary)
Orange (nee France Telecom)

- Network Applications
- C++
- Solaris
- Sparc
- 4GB and up
UML Tools Use Different Subsets of UML

Meaningful interchange between tools is difficult.
Executable UML Foundation

The Executable UML Foundation defines:

- An executable subset
- A definition of the execution semantics of that subset
- A base semantics

Semantic Backplane

Model Builder → Model Verifier → Model Tester → Model Analyzer → Model Compiler

Other tools
When will Execution be Commonplace?

1985: “In three years time…”
1987: “In three years time…”
1989: “In three years time…”
1991: “In three years time…”
1993: “In three years time…”
1995: “In three years time…”
1997: “In three years time…”
1999: “In three years time…”
2001: “In three years time…”
2003: “In three years time…”
2005: “In three years time…”
Why now?

Knowledge
Individuals
Projects
Companies

Market Usage
Sketchers
Blueprinters
Executable Modelers

System Complexity
Programs
Systems
Systems of Systems

Standards
Methods
UML
Interchange
A prediction for modeling

- UML
  - xtUML
  - Actions
- Mappings with text
- Mappings with metamodels (QVT)
- Executable UML Foundation
- Other standards
- DSLs
Separating notation enables *domain-specific languages*, graphical languages specific to a particular domain:

- VCR controls
- Fax machines
- Chemical plant
- Train control

UML would be used for domains with no pre-existing standard language, or for software
Questions?

StephenMellor@StephenMellor.com