MOOSE Seminar Keynote

“State of the practice in European embedded software engineering”

Rini van Solingen

Moose Project Manager
Principal Consultant in SW Management, LogicaCMG
Professor in Quality Management & Engineering, Drenthe University

Rini.van.Solingen@LogicaCMG.com

Agenda

- Introduction
- MOOSE web-repository
- Analysis of state-of-the-practice
- Strategies for increasing “deployability”
- Conclusions

http://www.mooseproject.org/
Introduction

Embedded SW Market

• Product is sold, not software
• Dominant hardware restrictions (memory, timing)
• Strongly based on previous products
• Increasing amount of software
• Application of software engineering technology

http://www.mooseproject.org/
Embedded Software Market

Number of embedded systems per household

Challenges embedded sw market

- Faster development
- More functionality with increasing complexity
- Increasing quality and performance demands
- Financial pressure on product cost, mostly hardware only

- Increasing demands on business drivers
- Technological innovation as a solution
  - Hardware technology
  - Software engineering technology
  - SE technology: methods, techniques, tools, and processes
Innovation = New Technology + Usage

- Innovation includes usage
  - Technology development without usage is NOT innovation
  - Many research initiatives focus on technology development only
  - Technology adoption by industry often lasts long

- Two types of innovation
  - Initial innovation
    - First time development and industrial application
  - Evolutionary innovation
    - Continuous improvement from application experiences

- Moose facilitation through experience exchange
  - Sharing what worked and did not work in which situation and why

http://www.mooseproject.org/

MOOSE web-repository
MOOSE web-repository

- Project centred storage and exchange of experiences
- Project characterisation
  - Effort, persons, lead-time, business driver, etc.
- Product characterisation
  - Software, Hardware, Real-time criticality, Market
- Technology characterisation
  - SE technology used
  - Satisfaction with the technology for that project (scale)
  - Short textual reason for this satisfaction rating
- Project evaluation report
  - Optional attachment with detailed findings and experiences
Content of the web-repository

- Currently: 78 projects included
- Projects originate from voluntary submission
  - Registered users can enter project experiences
  - Evaluation board evaluates submission on completeness and reliability
  - Registered users can contact project owners
  - Unregistered users can only browse anonymous projects and technologies
- Analysis of the web-repository
  - Provides insight in state-of-the-practice
  - First idea on trends
  - Limited validity of findings and conclusions

http://www.mooseproject.org/

Analysis of the web-repository

What is the state-of-the-practice?
Project characterisation (1) [Business driver and Spread]

Main business driver

- Schedule: 80%
- Functionality: 40%
- Quality: 5%
- Budget: 10%
- Performance: 5%

80% of the projects are driven by schedule or functionality demands

Number of teams and sites

- 1 site: 70%
- 2 teams: 50%
- 3 or more teams: 10%

70% of the projects are undertaken on 1 site with max. 2 teams

Project characterisation (2) [Lead-time and Effort]

Lead-time (months)

- <3 months: 15%
- 4-6 months: 20%
- 7-12 months: 25%
- 13-24 months: 15%
- >24 months: 10%

60% of the projects have a duration of <1 year
80% of the projects have a duration of <2 years

Effort (person years)

- <1 person year: 5%
- 1-5 person years: 30%
- 6-10 person years: 20%
- 10-50 person years: 20%
- >50 person years: 15%

50% of the projects cost less than 5 person years of effort
5% more than 50 person years of effort

http://www.mooseproject.org/
Product characterisation (1)
[Type and Real-time criticality]

Product Type
- Professional
- Consumer
- OEM

Real-time criticality
- 0% 10% 20% 30% 40% 50% 60%

50% of the products have hard real-time constraints
50% have soft or not

Product characterisation (2)
[SW Size and HW Size]

Size (KSLOC)
- <5
- 5-10
- 11-100
- 100-1000
- >1000
- Left blank

Size (HW memory limitations)
- <128 Kbytes
- 128-1024 Kbytes
- 1-16 Mbytes
- >16 Mbytes

60% of the products have more than 100,000 lines of software code
35% of the products have memory boundaries below 1 Mbyte
30% have over 16 Mbyte
Product characterisation (3) [SW and HW]

Technology Characterisation

What is the state-of-the-practice in SE technology application?
Which Requirements Engineering Methods are used?

- None used: 55%
- Proprietary method: 10%
- RequisitePro: 20%
- Problem Frame approach: 4%
- Interviews: 3%
- Use cases: 1%
- Other: 1%

55% of the products are developed without use of a RE method.

Which Requirements Engineering Tools are used?

- MsWord: 50%
- No tool used: 20%
- RequisitePro: 10%
- MsExcel: 5%
- Other: 5%

50% of the projects use MsWord/Excel for RE.
20% use dedicated RE tools.
Which Design methods are used?

<table>
<thead>
<tr>
<th>Design Method</th>
<th>0%</th>
<th>5%</th>
<th>10%</th>
<th>15%</th>
<th>20%</th>
<th>25%</th>
<th>30%</th>
<th>35%</th>
</tr>
</thead>
<tbody>
<tr>
<td>None used</td>
<td>30%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UML</td>
<td>25%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proprietary method</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SA/SD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Object Oriented</td>
<td>40%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design Tooling</td>
<td>0%</td>
<td>5%</td>
<td>10%</td>
<td>15%</td>
<td>20%</td>
<td>25%</td>
<td>30%</td>
<td>35%</td>
</tr>
<tr>
<td>No tool used</td>
<td>25%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MsVisio/Visual studio</td>
<td>20%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rational Rose</td>
<td>15%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MsWord</td>
<td>10%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rational Rose RT</td>
<td>5%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prosa</td>
<td>5%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

30% of the products is designed without use of a method
25% by use of UML

Which Design tools are used?

<table>
<thead>
<tr>
<th>Design Tooling</th>
<th>0%</th>
<th>5%</th>
<th>10%</th>
<th>15%</th>
<th>20%</th>
<th>25%</th>
<th>30%</th>
</tr>
</thead>
<tbody>
<tr>
<td>No tool used</td>
<td>25%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MsVisio/Visual studio</td>
<td>20%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rational Rose</td>
<td>15%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MsWord</td>
<td>10%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rational Rose RT</td>
<td>5%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prosa</td>
<td>5%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

25% of the products is designed without use of a tool
20% by a generic drawing tool
Which coding languages are used?

<table>
<thead>
<tr>
<th>Programming Language</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>20%</td>
</tr>
<tr>
<td>C++</td>
<td>55%</td>
</tr>
<tr>
<td>Assembler</td>
<td>15%</td>
</tr>
<tr>
<td>Java</td>
<td>10%</td>
</tr>
<tr>
<td>Other</td>
<td>0%</td>
</tr>
</tbody>
</table>

55% of the products is made in C/C++
20% in Assembler
15% in Java

Which Test Tools are used?

<table>
<thead>
<tr>
<th>Test Tooling</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>No tool used/manual testing</td>
<td>30%</td>
</tr>
<tr>
<td>Proprietary</td>
<td>10%</td>
</tr>
<tr>
<td>QAC/C++</td>
<td>5%</td>
</tr>
<tr>
<td>Hitex Debugger</td>
<td>5%</td>
</tr>
<tr>
<td>PurifyPlus</td>
<td>5%</td>
</tr>
<tr>
<td>GNU tooling</td>
<td>10%</td>
</tr>
<tr>
<td>TLCS Debugger</td>
<td>5%</td>
</tr>
<tr>
<td>Hardware test benches</td>
<td>5%</td>
</tr>
<tr>
<td>ConTest/TestFrame</td>
<td>5%</td>
</tr>
<tr>
<td>Other</td>
<td>0%</td>
</tr>
</tbody>
</table>

30% of the products is manually tested
Which Configuration Management tools are used?

Configuration Management Tooling

- CMSynergy
- ClearCase
- PVCS
- No tool used/manual CM
- VisualSourceSafe
- CVS
- Proprietary tool
- Other

90% of the products is developed using a CM tool

Which PR/CR handling tools are used?

PR/CR handling Tooling

- No tool used/manual
- ChangeSynergy
- ClearQuest
- MsWord
- Proprietary tools
- Lotus Notes
- ClearDDTS
- MsExcel
- Other

40% of the projects use a PR/CR tool
20% have no tool
20% use a generic tool
Which Management processes are installed

Management process

ISO900x
CMM level 2
Proprietary
CMM level 3
None used
Other

0% 5% 10% 15% 20% 25% 30%

35% of the projects use CMM for process management

http://www.mooseproject.org/

---

Which Engineering process is used?

Engineering process

Incremental
Waterfall
Proprietary
Time-boxing
Parallel development
No engineering process
Other

0% 10% 20% 30% 40% 50%

50% of the products is developed by an incremental engineering process

http://www.mooseproject.org/
Overview over technologies and SE

<table>
<thead>
<tr>
<th>SE Technology</th>
<th>Value</th>
<th>Most Used</th>
<th>Most Satisfied</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management process</td>
<td>★★★★</td>
<td>ISO900x</td>
<td>CMM level 2</td>
</tr>
<tr>
<td>Engineering process</td>
<td>★★★★</td>
<td>Incremental</td>
<td>Incremental</td>
</tr>
<tr>
<td>RE method</td>
<td>★★★★</td>
<td>No method</td>
<td>Interviews / Use-cases</td>
</tr>
<tr>
<td>Design method</td>
<td>★★★★</td>
<td>No method</td>
<td>UML-RT</td>
</tr>
<tr>
<td>Programming language</td>
<td>★★★★</td>
<td>C/C++</td>
<td>C/C++</td>
</tr>
<tr>
<td>RE tooling</td>
<td>★★★★</td>
<td>MsWord</td>
<td>RequisitePro</td>
</tr>
<tr>
<td>Design tooling</td>
<td>★★★★</td>
<td>No tool</td>
<td>Rose RT</td>
</tr>
<tr>
<td>Test tooling</td>
<td>★★★★</td>
<td>No tool</td>
<td>HW tools / ConTest</td>
</tr>
<tr>
<td>CM tooling</td>
<td>★★★★</td>
<td>CMSynergy</td>
<td>CMSynergy</td>
</tr>
<tr>
<td>PR/CR tooling</td>
<td>★★★★</td>
<td>No tool</td>
<td>ChangeSynergy</td>
</tr>
</tbody>
</table>

Summary state-of-the-practice

- Product characterisation
  - Most products contain more than 100,000 lines of software code
  - 70% of products have memory boundaries above 1 Mbyte
  - Half of the products have limited or no real-time constraints
  - 30% of the products have MsWindows OS and are Intel/PC-based

- Project characterisation
  - Most projects are driven by schedule or functionality, have a duration of <2 years, and are undertaken on 1 geographic location
  - Half of the projects cost less than 5 person years of effort

- Technology characterisation
  - Half of the products are built without a RE method using MsWord/Excel
  - 30% of the products is designed without a method, 25% by use of UML
  - Half of the products are programmed in C/C++
  - Almost all products are developed using a configuration management tool
Opportunities for innovation

- Real-time specific tools score well, but are hardly used. Maturisation of RT specific tools might be interesting
- No method or tool support common for Requirements Engineering. RE seems most promising improvement area for embedded systems
- CMM most actual used and appreciated, so adoption of CMM-I seems opportunity
- Introduction of embedded test tooling, integrated with HW test tooling potential improvement area
- Integration of technologies continues to be large opportunity, however, highly context dependent. Default integrated tool set based on most used technologies could be interesting for industry

Reasons for not using innovative technologies

- Legacy in technologies is leading
- Time-pressure in project does not leave time for new things
- Immaturity of new technologies and complexity (learning investment) is too high
- Risks for new technology introduction are too high
- Benefits of technologies are not clear upfront, guarantees are not given
- Experiences or measurements are hardly available
- Sentiment
- Deployment is major challenge in industry
Increasing “Deployability”

- Maturity assessment of technologies
- Impact specification of technologies
- Interfacing for technology chains
- Measurement and exchange of experiences
- Variations of technologies to application domains
- Increased collaboration between technology providers and users
- Paradigm shift in SW engineering research:
  – from revolution to evolution
  – from introduction to maturisation
  – from development to evaluation

Conclusions
**State-of-the-practice**

- Large gap between available and industry used technologies
- Industry acts conservative towards SE technologies
  - Often no methods used
  - Mainly generic tools used
  - Proven technology is used at low risk
  - Pragmatic approaches
- Industry is often not able to make rigorous changes
  - So: **Minimal changes but with maximal results**
    - Not a revolution strategy towards innovation
    - Evolution strategy towards innovation
    - However, this is not supported by most technologies

**MOOSE web-repository**

- Sharing experiences among practitioners
- Finding projects that are similar to own situation
- Finding proposals for new/innovative technologies
- Getting in touch with other projects directly
- Support for minimal change maximum effect strategy
- Web-repository in public domain and maintained on open-source concepts
- Feasibility of the web-repository depends on continuous addition of new project experiences
- Joint benefit from joint effort
Thank you for your attention

Rini.van.Solingen@LogicaCMG.com

http://www.mooseproject.org/