How to Select a Suitable Tool for a Software Development Project: Three Case Studies and Lessons Learned

Mark Kibanov\(^1,2\), Dominik J. Erdmann\(^1\), Martin Atzmüller\(^1,2\)

\(^1\)Knowledge and Data Engineering Group, Department of Electrical Engineering and Computer Science, University of Kassel

\(^2\)Research Center L3S, Hannover

Modellierung von Vorgehensmodellen – Paradigmen, Sprachen, Tools
Aachen, 2013-02-27
1 Introduction

2 Related Work or Why We Reinvent the Wheel

3 Framework

4 Case Studies

5 Conclusions
Agenda

1. Introduction
2. Related Work or Why We Reinvent the Wheel
3. Framework
4. Case Studies
5. Conclusions
Why?

- Every team needs at some point to choose a tool, framework or program.
Introduction

Why?

- Every team needs at some point to choose a tool, framework or program.
- And the team needs some system (objectivity) \( \rightarrow \) decision framework or method.
Why?

- Every team needs at some point to choose a tool, framework or programm.
- And the team needs some system (objectivity) → decision framework or method.
- Or: you just need to explain to the manager or customer why you have chosen current tool.
Introduction

Why?

- Every team needs at some point to choose a tool, framework or programm.
- And the team needs some system (objectivity) → decision framework or method.
- Or: you just need to explain to the manager or customer why you have chosen current tool.

What?

- Framework to choose a suitable tool
- Based on AHP
- Proved by three case studies
Agenda

1. Introduction
2. Related Work or Why We Reinvent the Wheel
3. Framework
4. Case Studies
5. Conclusions
Existing Models

- Analytic Hierarchy Process (AHP)
Existing Models

- Analytic Hierarchy Process (AHP)
- Cost-Benefit Analysis (CBA)
Existing Models

- Analytic Hierarchy Process (AHP)
- Cost-Benefit Analysis (CBA)
- Capability Maturity Model (CMM)
Existing Models

- Analytic Hierarchy Process (AHP)
- Cost-Benefit Analysis (CBA)
- Capability Maturity Model (CMM)
- Capability Maturity Model Integration (CMMI)
Existing Models

- Analytic Hierarchy Process (AHP)
- Cost-Benefit Analysis (CBA)
- Capability Maturity Model (CMM)
- Capability Maturity Model Integration (CMMI)
- Control Objectives Information and Related Technology (COBIT)
Agenda

1. Introduction

2. Related Work or Why We Reinvent the Wheel

3. Framework

4. Case Studies

5. Conclusions
Proposed Framework

Yet Another Model?

No! This is the framework for the rest of us:

- Small teams
- Low complexity
- But still powerful and flexible
Yet Another Model?

No! This is the framework for the rest of us
Proposed Framework

Yet Another Model?

No! This is the framework for the rest of us:

- Small teams
- Low complexity
- But still powerful and flexible
3-Step Framework

- Identify the software according to minimal and desired requirements
3-Step Framework

- Identify the software according to minimal and desired requirements
- Quantify the requirements
3-Step Framework

- Identify the software according to minimal and desired requirements
- Quantify the requirements
- Evaluate the software according to the requirements
Step 1: Minimal and Desired Requirements

3-Step Framework

1. Project stakeholders define (functional and technical) requirements
Step 1: Minimal and Desired Requirements

3-Step Framework

Step 1

1. Project stakeholders define (functional and technical) requirements
2. Define the software type
Step 1: Minimal and Desired Requirements

3-Step Framework

1. Project stakeholders define (functional and technical) requirements
2. Define the software type
3. Select the software to evaluate (according to minimal requirements)
Step 2: Quantification of Requirements

3-Step Framework

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eigenschaft</td>
<td>Gewicht</td>
<td></td>
</tr>
<tr>
<td>Allgemeine Eigenschaften</td>
<td></td>
<td>20</td>
</tr>
<tr>
<td>A</td>
<td>Software-Qualität</td>
<td>12</td>
</tr>
<tr>
<td>A.1</td>
<td>Seriosität</td>
<td>3</td>
</tr>
<tr>
<td>A.2</td>
<td>Benutzerfreundlichkeit</td>
<td>2</td>
</tr>
<tr>
<td>A.3</td>
<td>Dokumentation</td>
<td>2</td>
</tr>
<tr>
<td>A.4</td>
<td>Support</td>
<td>1</td>
</tr>
<tr>
<td>A.5</td>
<td>Fehlerfreiheit</td>
<td>2</td>
</tr>
<tr>
<td>A.6</td>
<td>Firmeninternes Know-how</td>
<td>2</td>
</tr>
<tr>
<td>A.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>Software-Merkmale</td>
<td>8</td>
</tr>
<tr>
<td>B.1</td>
<td>Atomic Commits</td>
<td>3</td>
</tr>
<tr>
<td>B.2</td>
<td>File Renames</td>
<td>2</td>
</tr>
<tr>
<td>B.3</td>
<td>Merge Tracking</td>
<td>3</td>
</tr>
</tbody>
</table>

Figure: Example of quantification of requirements by team member (MES GUI case study)
Step 2: Quantification of Requirements

3-Step Framework

\[ w(r_i) = \frac{a(r_i)}{\sum_{k=1}^{n} a(r_k)} , \]
### Step 2: Quantification of Requirements

#### 3-Step Framework

**Figure:** Example of normalized requirements weight (MES GUI case study)

<table>
<thead>
<tr>
<th></th>
<th>Eigenschaft</th>
<th>Gewicht</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Eigenschaft</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Allgemeine Eigenschaften</td>
<td>20</td>
<td>36,36</td>
</tr>
<tr>
<td>3</td>
<td>A Software-Qualität</td>
<td>12</td>
<td>21,82</td>
</tr>
<tr>
<td>4</td>
<td>A.1 Seriosität</td>
<td>3</td>
<td>5,45</td>
</tr>
<tr>
<td>5</td>
<td>A.2 Benutzerfreundlichkeit</td>
<td>2</td>
<td>3,64</td>
</tr>
<tr>
<td>6</td>
<td>A.3 Dokumentation</td>
<td>2</td>
<td>3,64</td>
</tr>
<tr>
<td>7</td>
<td>A.4 Support</td>
<td>1</td>
<td>1,82</td>
</tr>
<tr>
<td>8</td>
<td>A.5 Fehlerfreiheit</td>
<td>2</td>
<td>3,64</td>
</tr>
<tr>
<td>9</td>
<td>A.6 Firmeninternes Know-how</td>
<td>2</td>
<td>3,64</td>
</tr>
<tr>
<td>10</td>
<td>A.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>B Software-Merkmale</td>
<td>8</td>
<td>14,55</td>
</tr>
<tr>
<td>12</td>
<td>B.1 Atomic Commits</td>
<td>3</td>
<td>5,45</td>
</tr>
<tr>
<td>13</td>
<td>B.2 File Renames</td>
<td>2</td>
<td>3,64</td>
</tr>
<tr>
<td>14</td>
<td>B.3 Merge Tracking</td>
<td>3</td>
<td>5,45</td>
</tr>
</tbody>
</table>
Figure: Put requirements from different team members together
Step 3: Evaluation of Software

3-Step Framework

Step 3

1. Define the scenarios

Input scenarios in decision analysis spreadsheet: The requirements are rated as follows:

1.0 = Alternative fully satisfies business requirement or decision criterion.
0.5 = Alternative partially satisfies business requirement or decision criterion.
0.0 = Unknown or Null/Balanced (the alternative neither satisfies nor dissatisfies business requirement or decision criterion).
-0.5 = Alternative partially dissatisfies business requirement or decision criterion.
-1.0 = Alternative fully dissatisfies business requirement or decision criterion.
Step 3: Evaluation of Software

3-Step Framework

Step 3

1. Define the scenarios
2. Execute the scenarios
Step 3: Evaluation of Software

3-Step Framework

Step 3

1. Define the scenarios
2. Execute the scenarios
3. Input scenarios in decision analysis spreadsheet: The requirements are rated as follows:

1.0 = Alternative fully satisfies business requirement or decision criterion.
0.5 = Alternative partially satisfies business requirement or decision criterion.
0.0 = Unknown or Null/Balanced (the alternative neither satisfies nor dissatisfies business requirement or decision criterion).
-0.5 = Alternative partially dissatisfies business requirement or decision criterion.
-1.0 = Alternative fully dissatisfies business requirement or decision criterion.
Step 3: Evaluation of Software
3-Step Framework

Step 3

1. Define the scenarios
2. Execute the scenarios
3. Input scenarios in decision analysis spreadsheet: The requirements are rated as follows:
   - 1.0 = Alternative fully satisfies business requirement or decision criterion.
   - 0.5 = Alternative partially satisfies business requirement or decision criterion.
   - 0.0 = Unknown or Null/Balanced (the alternative neither satisfies nor dissatisfies business requirement or decision criterion).
   - -0.5 = Alternative partially dissatisfies business requirement or decision criterion.
   - -1.0 = Alternative fully dissatisfies business requirement or decision criterion.
### Step 3: Decision Analysis Spreadsheet

#### 3-Step Framework

<table>
<thead>
<tr>
<th>Eigenschaft</th>
<th>Gew</th>
<th>SVN</th>
<th>relat</th>
<th>Git</th>
<th>relat</th>
<th>Merk</th>
<th>relat</th>
<th>Synch</th>
<th>relat</th>
<th>Plast</th>
<th>relat</th>
<th>Baza</th>
<th>relat</th>
<th>MIS</th>
<th>relat</th>
<th>Perf</th>
<th>relat</th>
<th>Accu</th>
<th>relat</th>
<th>Pure</th>
<th>relat</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Software-Qualität</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A.1 Seriösität</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0.5</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>0.5</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>0.5</td>
<td>2</td>
</tr>
<tr>
<td>A.2 Benutzerfreundlichkeit</td>
<td>2</td>
<td>0.5</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>-0.5</td>
<td>-1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0.5</td>
<td>2</td>
<td>0.5</td>
</tr>
<tr>
<td>A.3 Dokumentation</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0.5</td>
<td>2</td>
<td>0.5</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0.5</td>
<td>2</td>
<td>0.5</td>
<td>2</td>
<td>0.5</td>
</tr>
<tr>
<td>A.4 Support</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>A.5 Fehlerfreiheit</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>B Software-Merkmale</td>
<td>10</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>B.1 Atomic Commits</td>
<td>5</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>B.2 File Renames</td>
<td>5</td>
<td>0.5</td>
<td>2.5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>B.3 Merge Tracking</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>C Projektumbgebung-integrierbarkeit</td>
<td>22</td>
<td>7</td>
<td>0.5</td>
<td>3.5</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>0.5</td>
<td>3.5</td>
<td>1</td>
<td>7</td>
<td>0.5</td>
<td>3.5</td>
<td>1</td>
</tr>
<tr>
<td>C.1 Eclipse Schnittstelle (Plug-in)</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>C.2 Jira-Schnittstelle (Plug-in)</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>C.3 Hudson-Schnittstelle (Plug-in)</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>C.4 Linux-Server-Kompatibilität</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>D Entwicklungsumgebung-Integrierbarkeit</td>
<td>35</td>
<td>7</td>
<td>0.5</td>
<td>3.5</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>0.5</td>
<td>3.5</td>
<td>1</td>
<td>7</td>
<td>0.5</td>
<td>3.5</td>
<td>1</td>
<td>7</td>
<td>0.5</td>
</tr>
<tr>
<td>D.1 Agile-Entwicklung-Umsetzung</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>D.2 Scrum-Umsetzung</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>D.3 verteilte Architektur</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>D.4 Optimistisches Locking</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>D.5 Changeset</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>E Sonstige MES-GUI-relevante Merkmale</td>
<td>9</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>E.1 Synergische Schnittstelle</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>E.2 Lizenz-Modell</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>E.3 Firmeninternes Know-How</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>

**Figure:** Example of decision analysis spreadsheet (MES GUI case study)
Agenda

1. Introduction

2. Related Work or Why We Reinvent the Wheel

3. Framework

4. Case Studies

5. Conclusions
Case Studies

- Industry Project – Scrum
Case Studies

- Industry Project – Scrum
- Ubiquitous Platform – VENUS
Case Studies

- Industry Project – Scrum
- Ubiquitous Platform – VENUS
- Research Project – Waterfall Model
Industry Project

Scrum

Project data

- GUI for the big Manufacturing Execution System (MES) (car industry)
- Scrum used
- Specific delivery of the product: source code to customer’s repository
- Version Control System is critical, SVN was not sufficient
Results

- 35 Systems evaluated
- 25 requirement collected
- 10 Systems evaluated selected for precise evaluation
- 5 Scenarios
- Interesting observations → optimization
Industry Project
Scrum

Results

- 35 Systems evaluated
- 25 requirement collected
- 10 Systems evaluated selected for precise evaluation
- 5 Scenarios
- Interesting observations \(\rightarrow\) optimization

<table>
<thead>
<tr>
<th>Criterion</th>
<th>relative Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Qualities</td>
<td>18%</td>
</tr>
<tr>
<td>Basic Features</td>
<td>17%</td>
</tr>
<tr>
<td>Integrability to the project Environment</td>
<td>22%</td>
</tr>
<tr>
<td>Integrability to the development process</td>
<td>34%</td>
</tr>
<tr>
<td>Other features relevant for the team</td>
<td>9%</td>
</tr>
</tbody>
</table>
Industry Project

Results

- 35 Systems evaluated
- 25 requirement collected
- 10 Systems evaluated selected for precise evaluation
- 5 Scenarios
- Interesting observations → optimization

<table>
<thead>
<tr>
<th>Criterion</th>
<th>relative Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Qualities</td>
<td>18%</td>
</tr>
<tr>
<td>Basic Features</td>
<td>17%</td>
</tr>
<tr>
<td>Integrability to the Project Environment</td>
<td></td>
</tr>
<tr>
<td>Integrability to the Development Process</td>
<td>65%</td>
</tr>
<tr>
<td>Other Features Relevant for the Team</td>
<td></td>
</tr>
</tbody>
</table>
### VCS

<table>
<thead>
<tr>
<th>VCS</th>
<th>DAS-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mercurial</td>
<td>77%</td>
</tr>
<tr>
<td>Bazaar</td>
<td>74%</td>
</tr>
<tr>
<td>Plastic SCM</td>
<td>69%</td>
</tr>
<tr>
<td>Git</td>
<td>67%</td>
</tr>
<tr>
<td>Synergy</td>
<td>63.5%</td>
</tr>
<tr>
<td>Perforce</td>
<td>50%</td>
</tr>
<tr>
<td>PureCM</td>
<td>43%</td>
</tr>
<tr>
<td>Integrity</td>
<td>40%</td>
</tr>
<tr>
<td>AccuRev SCM</td>
<td>39.5%</td>
</tr>
<tr>
<td>SVN</td>
<td>38%</td>
</tr>
</tbody>
</table>
### VCS and DAS-Score

<table>
<thead>
<tr>
<th>VCS</th>
<th>DAS-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mercurial</td>
<td>77%</td>
</tr>
<tr>
<td>Bazaar</td>
<td>74%</td>
</tr>
<tr>
<td>Plastic SCM</td>
<td>69%</td>
</tr>
<tr>
<td>Git</td>
<td>67%</td>
</tr>
<tr>
<td>Synergy</td>
<td>63.5%</td>
</tr>
<tr>
<td>Perforce</td>
<td>50%</td>
</tr>
<tr>
<td>PureCM</td>
<td>43%</td>
</tr>
<tr>
<td>Integrity</td>
<td>40%</td>
</tr>
<tr>
<td>AccuRev SCM</td>
<td>39.5%</td>
</tr>
<tr>
<td>SVN</td>
<td>38%</td>
</tr>
</tbody>
</table>
Ubiquitous Platform Ubicon

VENUS

Project data

- Platform for ubiquitous systems & applications
- Everyaware applications (http://cs.everyaware.eu/), Conferator (http://www.conferator.org/), MyGroup are powered by Ubicon (http://ubicon.eu/)
- VENUS (aka “Kasseler Methodik”) partially used for development
- Distributed team
- Current system - FusionForge: security and usability issues
BibSonomy: The Blue Social Bookmark And Publication Sharing System (http://www.bibsonomy.org)

FusionForge: Similar problems as by Ubicon

The main question: should there be the one project management software?
### Table: Weight of different requirement types in the Ubicon and BibSonomy cases.

<table>
<thead>
<tr>
<th>Criterion</th>
<th>relative Weight Ubicon</th>
<th>relative Weight BibSonomy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issue Tracker</td>
<td>16.2%</td>
<td>15.8%</td>
</tr>
<tr>
<td>Continuous Integration Interface</td>
<td>8.1%</td>
<td>7.3%</td>
</tr>
<tr>
<td>User Administration</td>
<td>5.9%</td>
<td>5.6%</td>
</tr>
<tr>
<td>Software Reliability</td>
<td>12.5%</td>
<td>17.8%</td>
</tr>
<tr>
<td>Version Control System</td>
<td>27.8%</td>
<td>31.7%</td>
</tr>
<tr>
<td>Project Management</td>
<td>6%</td>
<td>3.2%</td>
</tr>
<tr>
<td>Developer Support</td>
<td>23.5%</td>
<td>18.6%</td>
</tr>
</tbody>
</table>

The requirements are very similar (was not expected).
### Table: Decision analysis spreadsheet scores for the top 4 project management systems selected for the Ubicon and BibSonomy projects.

<table>
<thead>
<tr>
<th>System</th>
<th>Ubicon DAS-Score</th>
<th>BibSonomy DAS-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jira</td>
<td>85%</td>
<td>90%</td>
</tr>
<tr>
<td>Redmine</td>
<td>80%</td>
<td>88%</td>
</tr>
<tr>
<td>Trac</td>
<td>57%</td>
<td>66%</td>
</tr>
<tr>
<td>FusionForge</td>
<td>45%</td>
<td>53%</td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th>System</th>
<th>Ubicon DAS-Score</th>
<th>BibSonomy DAS-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jira</td>
<td>85%</td>
<td>90%</td>
</tr>
<tr>
<td>Redmine</td>
<td>80%</td>
<td>88%</td>
</tr>
<tr>
<td>Trac</td>
<td>57%</td>
<td>66%</td>
</tr>
<tr>
<td>FusionForge</td>
<td>45%</td>
<td>53%</td>
</tr>
</tbody>
</table>

Table: Decision analysis spreadsheet scores for the top 4 project management systems selected for the Ubicon and BibSonomy projects.
Agenda

1 Introduction

2 Related Work or Why We Reinvent the Wheel

3 Framework

4 Case Studies

5 Conclusions
Conclusions

Lessons Learned

Framework Overall

- The framework can be used in different environments

Case Studies

MES GUI: the new VCS-system integrated, workflows updated (optimized)

BibSonomy and Ubicon: transition to Redmine in progress

Overall better understanding of development processes
Conclusions

Lessons Learned

Framework Overall

- The framework can be used in different environments
- The framework can not make decisions: the best score does not necessary “win”

Case Studies

MES GUI: the new VCS-system integrated, workflows updated (optimized)
BibSonomy and Ubicon: transition to Redmine in progress
Overall better understanding of development processes
Conclusions

Lessons Learned

Framework Overall
- The framework can be used in different environments
- The framework can not make decisions: the best score does not necessary “win”
- The framework provides some additional useful information about current workflows

Case Studies

MES GUI: the new VCS-system integrated, workflows updated (optimized)
BibSonomy and Ubicon: transition to Redmine in progress
Overall better understanding of development processes
Conclusions
Lessons Learned

Framework Overall
- The framework can be used in different environments
- The framework cannot make decisions: the best score does not necessarily "win"
- The framework provides some additional useful information about current workflows

Case Studies
- MES GUI: the new VCS-system integrated, workflows updated (optimized)
Conclusions

Lessons Learned

**Framework Overall**

- The framework can be used in different environments
- The framework can not make decisions: the best score does not necessary “win”
- The framework provides some additional useful information about current workflows

**Case Studies**

- MES GUI: the new VCS-system integrated, workflows updated (optimized)
- BibSonomy and Ubicon: transition to Redmine in progress
Conclusions

Lessons Learned

Framework Overall

- The framework can be used in different environments
- The framework cannot make decisions: the best score does not necessarily “win”
- The framework provides some additional useful information about current workflows

Case Studies

- MES GUI: the new VCS-system integrated, workflows updated (optimized)
- BibSonomy and Ubicon: transition to Redmine in progress
- Overall better understanding of development processes
Thank you for your attention!