An Empirical Study on End-users Productivity Using Model-based Spreadsheets

Laura Beckwith  Jácome Cunha  João P. Fernandes  João Saraiva

HCIResearcher, Denmark, beckwith@hciResearcher.com
Universidade do Minho, Portugal, \{jacome,jpaulo,jas\}@di.uminho.pt
Universidade do Porto, Portugal, jpaulo@fe.up.pt

EuSpRIG 2011

July 14 – 15
1. Introduction

2. Model-based Spreadsheets
   - Inferring Models
   - Refactored Spreadsheet
   - Visual Spreadsheet

3. Study Design

4. Results
   - Effectiveness
   - Efficiency

5. Conclusions
Much research has been done to prevent spreadsheet end-user errors
Model-Driven Engineering (MDE) is a promising approach
We have created a set of different model-based spreadsheets
Much research has been done to prevent spreadsheet end-user errors
Model-Driven Engineering (MDE) is a promising approach
We have created a set of different model-based spreadsheets

Need to evaluate their impact
Original Spreadsheet - *original*

A project management system

Tabular spreadsheet (database like spreadsheet)
Functional Dependencies

- \( A \rightarrow B \) means that the values in column \( A \) uniquely determine values in column \( B \).
- If two values two rows in column \( A \) are equal, then the corresponding values in \( B \) are also equal.

For example, suppose \( A = \{A_1, A_2\} \) and \( B = \{B_1, B_2\} \):

(a) Table inducing \( A \rightarrow B \):

<table>
<thead>
<tr>
<th>A_1</th>
<th>A_2</th>
<th>B_1</th>
<th>B_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>a_1</td>
<td>a_2</td>
<td>b_1</td>
<td>b_2</td>
</tr>
<tr>
<td>a_1</td>
<td>a_3</td>
<td>b_1</td>
<td>b_1</td>
</tr>
<tr>
<td>a_1</td>
<td>a_2</td>
<td>b_1</td>
<td>b_2</td>
</tr>
</tbody>
</table>

(b) Table not inducing \( A \rightarrow B \):

<table>
<thead>
<tr>
<th>A_1</th>
<th>A_2</th>
<th>B_1</th>
<th>B_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>a_1</td>
<td>a_2</td>
<td>b_1</td>
<td>b_2</td>
</tr>
<tr>
<td>a_1</td>
<td>a_3</td>
<td>b_1</td>
<td>b_1</td>
</tr>
<tr>
<td>a_1</td>
<td>a_2</td>
<td>b_2</td>
<td>b_2</td>
</tr>
</tbody>
</table>
From the spreadsheet data (also from formulas) we can infer functional dependencies.

From these dependencies, we can compute a relational database schema.
We use the relational model to generate a new refactored spreadsheet (respecting the relational model)

Each entity separated in a different table

*Combo boxes* used to list allowed values (from other tables)
Visual Spreadsheet - visual

- From functional dependencies we can generate edit assistance
- $A \rightarrow B$: A columns become green combo boxes and B red
- Similar to original
- Combo boxes help user introducing data
Goal: compare efficiency and effectiveness of original, refactored and visual
Research Questions

**RQ1** Do end users introduce fewer errors when they use one of the model-based spreadsheets versus the original unmodified one?

**RQ2** Are end users more efficient using the model-based spreadsheets?

**RQ3** Do particular models lead to fewer errors in particular tasks?
Study Design

- Background questionnaire, followed by a list of tasks and a post questionnaire
- **No initial tutorial was given to the participants**
- 38 participants: 25 females and 11 males, all students
- Tasks for inserting and editing data and computing statistics
- 3 different spreadsheet applications (*PROJECTS, DISHES and PROPERTIES*)
- 3 different models (*original, refactored, visual*)
- Each participant received 3 spreadsheets, each of a different application/model
<table>
<thead>
<tr>
<th></th>
<th>original</th>
<th>refactored</th>
<th>visual</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISHES</td>
<td>86%</td>
<td>76%</td>
<td>78%</td>
</tr>
<tr>
<td>PROJECTS</td>
<td>73%</td>
<td>68%</td>
<td>78%</td>
</tr>
<tr>
<td>PROPERTIES</td>
<td>75%</td>
<td>64%</td>
<td>62%</td>
</tr>
</tbody>
</table>

- Globally, models do not seem very competitive
- Further analyses are necessary
Effectiveness: Inserting Data

<table>
<thead>
<tr>
<th></th>
<th>original</th>
<th>refactored</th>
<th>visual</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISHES</td>
<td>91%</td>
<td>90%</td>
<td>81%</td>
</tr>
<tr>
<td>PROJECTS</td>
<td>76%</td>
<td>60%</td>
<td>75%</td>
</tr>
<tr>
<td>PROPERTIES</td>
<td>86%</td>
<td>67%</td>
<td>68%</td>
</tr>
</tbody>
</table>

- Models always lose
- But very similar to original
Effectiveness: Editing Data

<table>
<thead>
<tr>
<th></th>
<th>original</th>
<th>refactored</th>
<th>visual</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISHES</td>
<td>91%</td>
<td>82%</td>
<td>82%</td>
</tr>
<tr>
<td>PROJECTS</td>
<td>54%</td>
<td>62%</td>
<td>50%</td>
</tr>
<tr>
<td>PROPERTIES</td>
<td>65%</td>
<td>98%</td>
<td>48%</td>
</tr>
</tbody>
</table>

- refactored in general was better
- Because there is less data to edit
Effectiveness: Computing Statistics

<table>
<thead>
<tr>
<th>Original</th>
<th>Refactored</th>
<th>Visual</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISHES</td>
<td>52%</td>
<td>37%</td>
</tr>
<tr>
<td>PROJECTS</td>
<td>19%</td>
<td>76%</td>
</tr>
<tr>
<td>PROPERTIES</td>
<td>44%</td>
<td>57%</td>
</tr>
</tbody>
</table>

Again, *refactored* was better

And again, less data do analyze
## Efficiency

<table>
<thead>
<tr>
<th></th>
<th>Original</th>
<th>Refactored</th>
<th>Visual</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISHES</td>
<td>35′</td>
<td>32′</td>
<td>28′</td>
</tr>
<tr>
<td>PROJECTS</td>
<td>39′</td>
<td>40′</td>
<td>41′</td>
</tr>
<tr>
<td>PROPERTIES</td>
<td>37′</td>
<td>36′</td>
<td>40′</td>
</tr>
</tbody>
</table>

- Models are competitive
- In further analyzes we discovered that users took more time understanding model, specially *refactored*
Answers to Research Questions

**RQ1** Do end users introduce fewer errors when they use one of the model-based spreadsheets versus the original unmodified one?

- We believe there is potential for model-based spreadsheets
- But a deep knowledge seems to be necessary to be effective

**RQ2** Are end users more efficient using model-based spreadsheets?

- Model-based spreadsheet can help in some cases
- We believe better knowledge about models could improve this result

**RQ3** Do particular models lead to fewer errors in particular tasks?

- For inserting data, this was not the case
- The models helped editing data and in computing statistics
Conclusions

- We did not give a tutorial to participants
- We believe better results could be achieved with initial training
- Nevertheless, our model-based spreadsheet are competitive
- Better for editing and statistics
- We plan to run a new study similar to this, but with an initial tutorial
Thank you for your attention!