Model-based Spreadsheet Engineering

Jácome Cunha
and the other team members of the SSaaPP and FATBIT research projects

HASLab / INESC TEC & Universidade do Minho

FATBIT/SSaaPP Workshop 2012
17-18 September
Why do Spreadsheets Matter?

Omnipresent
Easy-to-use
Multi-purpose
Flexible

95% of all U.S. firms
90% of all analysts in industry
50% of all spreadsheets are the basis for decisions
Still...

http://www.eusprig.org/stories.htm

W. Baraboo to pay more for borrowed money than believed

December 09, 2011 8:17 pm • By Brian D. Bridgeford, News Republic

Due to a calculating error by their financial advisors, West Baraboo officials learned Thursday they will be paying about $400,000 more over the lifetime of their most recent 10-year borrowing plan than originally projected.

During its regular December meeting, the West Baraboo Village Board looked back over last month’s decision to sell $1.1 million in general obligation bonds to cover a variety of village projects, said Village Clerk Mary Klingenhoffer. The review was required after the board received a letter from its financial advisory firm, Echelers of Brookfield.

Echelers advisor James Mann said “operator error” resulted in a spreadsheet understimating the total cost of the 10-year bond.

Utah education officials make $25M school funding mistake

Education > State leaders say schools won’t likely be hurt by the miscalculation, which leads to the resignations of two top finance officials.

By Lisa Schnecker | The Salt Lake Tribune
First Published Apr 11 2012 11:54 am • Last Updated Apr 11 2012 11:22 pm

A miscalculation at the State Office of Education has led to a $25 million mistake in Utah’s education budget for next school year — and the resignation of two top finance officials.

Education leaders, however, say they don’t expect the potential shortfall to hurt schools or districts. State leaders are considering solutions ranging from using education money expected to be left over at the end of this school year to calling a special legislative session.

“We committed to fund [enrollment] growth and this is an important part of growth,” said Senate Budget Chairman Lyle Hillyard on Wednesday. “We would hope to get it fixed, and I think that’s going to be our first priority.”

The $25 million represents less than 1 percent of the state’s overall $3 billion-plus education budget.

The problem was that the state office essentially underestimated the number of students expected in schools next school year. The correct number will cost the state $25 million more than anticipated.

State Superintendent Larry Shumway attributed the mistake to “a faulty reference” in a spreadsheet. He emphasized that no money was misappropriated. He called the mistake “significant” but “manageable.”
The Contribution of Models
ClassSheets: automatic generation of spreadsheet applications from object-oriented specifications, Gregor Engels, Martin Erwig, ASE'05
Model-based Spreadsheet Engineering

I. ClassSheet Model Inference
[PEPM '09, VL/HCC '10]

II. Embedding ClassSheets in Spreadsheets
[VL/HCC '11, SAC '12]

III. Co-evolution of Models and Instances
[FASE '11, ICMT '12, ICSE '12]
I. ClassSheet Model Inference
I. ClassSheet Model Inference

Data mining techniques
Database normalization theory

Automatically Inferring ClassSheet Models from Spreadsheets, Jácome Cunha, Martin Erwig, João Saraiva, VL/HCC'10
Pruning Functional Dependencies

- **Label semantics**: usually keys are labeled “code” or “id”
- **Label arrangement**: we prefer FDs respecting the order of columns
- **Antecedent size**: small keys are preferable
- **Ratio**: small antecedents and big consequents
- **Single value columns**: columns always with the same value are too intrusive
II. Embedding ClassSheets in Spreadsheets
II. Embedding ClassSheets in Spreadsheets

Powerful interactive interface
Single environment for spreadsheet evolution
Model-instance synchronization

Syntactic restrictions

Embedding and Evolution of Spreadsheet Models in Spreadsheet Systems, Jácome Cunha, Jorge Mendes, João P. Fernandes, João Saraiva, VL/HCC '11
Vertically Expandable Tables

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 PILOTS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 ID</td>
<td>Name</td>
<td>Flight hours</td>
</tr>
<tr>
<td>3 pl1</td>
<td>John</td>
<td>3400</td>
</tr>
<tr>
<td>4 pl2</td>
<td>Mike</td>
<td>330</td>
</tr>
<tr>
<td>5 pl3</td>
<td>Anne</td>
<td>433</td>
</tr>
</tbody>
</table>

```python
id="" name="" flight_hours=0
```

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 PILOTS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 ID</td>
<td>Name</td>
<td>Flight hours</td>
</tr>
<tr>
<td>3 id=&quot;&quot; name=&quot;&quot; flight_hours=0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Horizontally Expandable Tables

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Planes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>N-Number</td>
<td>N2342</td>
<td>N341</td>
<td>N1343</td>
</tr>
<tr>
<td>3</td>
<td>Model</td>
<td>B 747</td>
<td>B 777</td>
<td>A 380</td>
</tr>
<tr>
<td>4</td>
<td>Name</td>
<td>Magalhães</td>
<td>Cabral</td>
<td>Nunes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Planes</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>N-Number</td>
<td>n-number=</td>
</tr>
<tr>
<td>3</td>
<td>Model</td>
<td>model=</td>
</tr>
<tr>
<td>4</td>
<td>Name</td>
<td>name=</td>
</tr>
</tbody>
</table>
### Relationship Tables

<table>
<thead>
<tr>
<th>Flights</th>
<th>PlanesKey</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flights</td>
<td>PlanesKey</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PlanesKey</th>
<th>N341</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>PilotsKey</th>
<th>Depart</th>
<th>Destination</th>
<th>Date</th>
<th>Hours</th>
<th>Depart</th>
<th>Destination</th>
<th>Date</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>pl1</td>
<td>OPO</td>
<td>NAT</td>
<td>12/12/2010</td>
<td>07:00</td>
<td>LIS</td>
<td>AMS</td>
<td>16/12/2010</td>
<td>02:45</td>
</tr>
<tr>
<td>pl1</td>
<td>OPO</td>
<td>NAT</td>
<td>01/01/2011</td>
<td>07:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Total Pilot Hours**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>09:45</td>
</tr>
<tr>
<td></td>
<td>07:00</td>
</tr>
</tbody>
</table>

**Flights**

<table>
<thead>
<tr>
<th>Flights</th>
<th>PlanesKey</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flights</td>
<td>PlanesKey</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PlanesKey</th>
<th>N341</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>PilotsKey</th>
<th>Depart</th>
<th>Destination</th>
<th>Date</th>
<th>Hours</th>
<th>Depart</th>
<th>Destination</th>
<th>Date</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>pilot_key=Plots.ID</td>
<td>depart=&quot;&quot;</td>
<td>destination=&quot;&quot;</td>
<td>date=d hours=0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Total Pilot Hours**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>09:45</td>
</tr>
<tr>
<td></td>
<td>07:00</td>
</tr>
</tbody>
</table>

**Total Pilot Hours**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16:45</td>
</tr>
</tbody>
</table>

**Total Pilot Hours**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16:45</td>
</tr>
</tbody>
</table>

**Total Pilot Hours**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16:45</td>
</tr>
</tbody>
</table>

**Total Pilot Hours**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16:45</td>
</tr>
</tbody>
</table>
III. Co-evolution of Models and Instances
Model-Driven Spreadsheet Environment (Model/ClassSheet Worksheet)
Model-Driven Spreadsheet Environment (Instance/Data Worksheet)
Type-safe Evolution of Spreadsheets, Jácome Cunha, Joost Visser, Tiago Alves, João Saraiva, FASE'11
Bidirectional Transformation of Model-Driven Spreadsheets, Jácome Cunha, João P. Fernandes, Jorge Mendes, Hugo Pacheco, and João Saraiva, ICMT '11
Bidirectional Transformation System

ClassSheet \xrightarrow{Op_M} ClassSheet'

conforms to

Spreadsheet \xleftarrow{Op_D} Spreadsheet'

to

from

conforms to
Example of a Transformation we Want: 
*Add a New Column*
(Model) Operations on ClassSheets

\[
\text{data } Op_M : \text{Model} \rightarrow \text{Model} =
\]

- `addColumn_M` \text{Where Index} \quad -- add a new column
- `delColumn_M` \text{Index} \quad -- delete a column
- `addRow_M` \text{Where Index} \quad -- add a new row
- `delRow_M` \text{Index} \quad -- delete a row
- `setLabel_M` \text{(Index, Index) Label} \quad -- set a label
- `setFormula_M` \text{(Index, Index) Formula} \quad -- set a formula
- `replicate_M` \text{ClassName Direction Int Int} \quad -- replicate a class
- `addClass_M` \text{ClassName (Index, Index) (Index, Index)} \quad -- add a static class
- `addClassExp_M` \text{ClassName Direction (Index, Index) (Index, Index)} \quad -- add an expandable class
(Data) Operations on Instances

\[
data \, Op_D : Data \rightarrow Data = \\
\begin{array}{ll}
addColumn_D & \text{Where Index} \quad \text{-- add a column} \\
delColumn_D & \text{Index} \quad \text{-- delete a column} \\
addRow_D & \text{Where Index} \quad \text{-- add a row} \\
delRow_D & \text{Index} \quad \text{-- delete a row} \\
AddColumn_D & \text{Where Index} \quad \text{-- add a column to all instances} \\
DelColumn_D & \text{Index} \quad \text{-- delete a column from all instances} \\
AddRow_D & \text{Where Index} \quad \text{-- add a row to all instances} \\
DelRow_D & \text{Index} \quad \text{-- delete a row from all instances} \\
replicate_D & \text{ClassName Direction Int Int} \quad \text{-- replicate a class} \\
addInstance_D & \text{ClassName Direction Model} \quad \text{-- add a class instance} \\
setLabel_D & (Index, Index) Label \quad \text{-- set a label} \\
setValue_D & (Index, Index) Value \quad \text{-- set a cell value} \\
SetLabel_D & (Index, Index) Label \quad \text{-- set a label in all instances} \\
SetValue_D & (Index, Index) Value \quad \text{-- set a cell value in all instances}
\end{array}
\]
Bidirectional Transformation Functions

\[ \text{to} : \text{Model} \times \text{Op}_M \rightarrow \text{Op}_D^* \]

\[ \text{from} : \text{Data} \times \text{Op}_D \rightarrow \text{Op}_M^* \]

\[
\begin{align*}
\text{to} : \text{Op}_M & \rightarrow \text{Op}_D^* \\
\text{to} \ (\text{addColumn}_M \ w \ i) & = \text{AddColumn}_D \ w \ (\text{columnIndex}_D \ i) \\
\text{to} \ (\text{delColumn}_M \ w \ i) & = \text{DelColumn}_D \ (\text{columnIndex}_D \ i) \\
\text{to} \ (\text{addRow}_M \ w \ i) & = \text{AddRow}_D \ w \ (\text{rowIndex}_D \ i) \\
\text{to} \ (\text{delRow}_M \ w \ i) & = \text{DelRow}_D \ (\text{rowIndex}_D \ i) \\
\text{to} \ (\text{setLabel}_M \ (i, j) \ l) & = \text{setLabel}_D \ (\text{position}_D \ (i, j)) \ l \\
\text{to} \ (\text{setFormula}_M \ (i, j) \ f) & = \text{setValue}_D \ (\text{position}_D \ (i, j)) \ f
\end{align*}
\]

\[
\begin{align*}
\text{from} : \text{Op}_D & \rightarrow \text{Op}_M^* \\
\text{from} \ (\text{addColumn}_D \ w \ i) & = \\
& \quad \text{replicate}_M \ \text{className} \ \text{Horizontal} \ \text{classInstances} \ \text{instanceIndex}_M \\
& \quad ; \ \text{addColumn}_M \ w \ \text{columnOffsetIndex}_M \\
\text{from} \ (\text{delColumn}_D \ i) & = \\
& \quad \text{replicate}_M \ \text{className} \ \text{Horizontal} \ \text{classInstances} \ \text{instanceIndex}_M \\
& \quad ; \ \text{delColumn}_M \ \text{columnOffsetIndex}_M \\
\text{from} \ (\text{addRow}_D \ w \ i) & = \\
& \quad \text{replicate}_M \ \text{className} \ \text{Vertical} \ \text{classInstances} \ \text{rowIndex}_M \\
& \quad ; \ \text{addRow}_M \ \text{rowOffsetIndex}_M \\
\text{from} \ (\text{delRow}_D \ i) & = \\
& \quad \text{replicate}_M \ \text{className} \ \text{Vertical} \ \text{classInstances} \ \text{rowIndex}_M \\
& \quad ; \ \text{delRow}_M \ \text{rowOffsetIndex}_M \\
\text{from} \ (\text{setLabel}_D \ (i, j) \ l) & = \\
& \quad \text{replicate}_M \ \text{className} \ \text{Horizontal} \ \text{classInstances} \ \text{columnIndex}_M \\
& \quad ; \ \text{replicate}_M \ \text{className} \ \text{Vertical} \ \text{classInstances} \ \text{rowIndex}_M \\
& \quad ; \ \text{setLabel}_M \ \text{positionOffset}_M \ l \\
\text{from} \ (\text{setValue}_D \ (i, j) \ l) & = \emptyset \\
\text{from} \ (\text{addInstance}_D \ \text{cn} \ \text{dir} \ m) & = \emptyset
\end{align*}
\]
Example: *Add a Column and a Class*

```plaintext
addRowM Before 3 ; addClassExpM "BlueClass" Horizontal (2,1) (3,4)
```

![Diagram](image)
Available at http://ssaapp.di.uminho.pt

Built out of ~7200 LOC:
- ~3700 in Haskell, for the evolution and inference
- ~600 in Basic, for the embedding
- ~2880 in C++, for gluing all components

MDSheet: A Framework for Model-driven Spreadsheet Engineering, Jácome Cunha, João P. Fernandes, Jorge Mendes, João Saraiva, ICSE '12
Conclusions

- We created a technique to infer ClassSheet models from legacy spreadsheets
- We have embedded a visual DSL into a traditional spreadsheet system
- This allows user to create models and instances in the same environment
- We used a formal framework to design and implement co-evolution steps
- The model and its instances are always synchronized
Conclusions

- Framework for bidirectional transformations
  - When evolving the data a new model is inferred
  - When evolving the model the data changes
  - The data always conforms to the model

- Available as a OpenOffice/LibreOffice extension
Thank You!

SSaaPP:
http://ssaapp.di.uminho.pt

FATBIT:
http://fatbit.di.uminho.pt