Model-based Spreadsheet Engineering

Jácome Cunha

Universidade do Minho

PhD Defense
March 22, 2011
Motivation

- Spreadsheets are widely used;
- Their freedom makes people quickly start work with them;
- This freedom is what makes them notoriously error-prone;
- We will present techniques to help spreadsheet end users;
We believe models can help spreadsheet end users;
Most spreadsheets do not have a model/specification;
It is difficult to an end user to create models;
Thus, we wish to automatically infer them from spreadsheet data;
Using these models we will make them more efficient and effective.
An Example

This spreadsheet represents a movie renting system:

- It stores information about movies, renters and leases.
Functional Dependencies

- We wish to automatically infer models from spreadsheet data;
- We discover relationships among spreadsheet data using *functional dependencies*;
- *Functional dependencies* express that some set of columns $A$ uniquely determines another set of columns $B$, $A \rightarrow B$;
- Using data mining algorithms and making use of spreadsheet idiosyncrasies, we can generate a set functional dependencies characterizing the spreadsheet data.
Relational Model

Using the functional dependencies inferred, we generate a relational model characterizing the spreadsheet:

- **Language** $(language)$
- **Payment** $(rentStart, rentFinish, rent, totalToPay)$
- **Renter** $(renterNr, renterNm, renterPhone)$
- **Movie** $(movieID, title, year, director, rent)$

$<Rent > (\#language, \#rentStart, \#rentFinish, \#renterNr, \#movieID)$
From the relational model, we can generate a *ClassSheet* diagram fully specifying the spreadsheet:
Given the similarities between ClassSheets and UML class diagrams, we generate the latter from the former:
Using the functional dependencies inferred before, we generate a spreadsheet with edit assistance:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>movieID</td>
<td>title</td>
<td>year</td>
<td>director</td>
<td>language</td>
<td>renterNr</td>
<td>renterNm</td>
<td>renterPhone</td>
<td>rentStart</td>
<td>rentFinished</td>
<td>rent</td>
<td>totalToPay</td>
</tr>
<tr>
<td>2</td>
<td>mv23</td>
<td>Little Man</td>
<td>2006</td>
<td>Keenen Wayans</td>
<td>English</td>
<td>c33</td>
<td>Paul</td>
<td>3334433</td>
<td>01-04-2010</td>
<td>26-04-2010</td>
<td>0,5</td>
<td>12,50</td>
</tr>
<tr>
<td>3</td>
<td>mv1</td>
<td>The OH in Ohio</td>
<td>2005</td>
<td>Billy Kent</td>
<td>English</td>
<td>c33</td>
<td>Paul</td>
<td>3334433</td>
<td>30-03-2010</td>
<td>23-04-2010</td>
<td>0,5</td>
<td>12,00</td>
</tr>
<tr>
<td>4</td>
<td>mv21</td>
<td>Edmond</td>
<td>2005</td>
<td>Stuart Gordon</td>
<td>English</td>
<td>c26</td>
<td>Smith</td>
<td>4445467</td>
<td>02-04-2010</td>
<td>04-04-2010</td>
<td>0,5</td>
<td>1,00</td>
</tr>
<tr>
<td>5</td>
<td>mv102</td>
<td>You, Me and D.</td>
<td>2001</td>
<td>Anthony Russo</td>
<td>English</td>
<td>c3</td>
<td>Michael</td>
<td>5551212</td>
<td>22-03-2010</td>
<td>03-04-2010</td>
<td>0,3</td>
<td>3,60</td>
</tr>
<tr>
<td>6</td>
<td>mv23</td>
<td>Little Man</td>
<td>2006</td>
<td>Keenen Wayans</td>
<td>English</td>
<td>c26</td>
<td>Smith</td>
<td>4445467</td>
<td>02-12-2009</td>
<td>04-04-2010</td>
<td>0,5</td>
<td>61,50</td>
</tr>
<tr>
<td>7</td>
<td>mv23</td>
<td>Little Man</td>
<td>2006</td>
<td>Keenen Wayans</td>
<td>English</td>
<td>c14</td>
<td>John</td>
<td>3332425</td>
<td>12-04-2010</td>
<td>16-04-2010</td>
<td>0,5</td>
<td>2,00</td>
</tr>
<tr>
<td>8</td>
<td>mv3</td>
<td>Alice</td>
<td>2009</td>
<td>Mark Jones</td>
<td>English</td>
<td>c33</td>
<td>Paul</td>
<td>3334433</td>
<td>12-04-2010</td>
<td>23-04-2010</td>
<td>0,5</td>
<td>5,50</td>
</tr>
<tr>
<td>9</td>
<td>mv5</td>
<td>I'm Legend</td>
<td>2005</td>
<td>Paul Billy</td>
<td>English</td>
<td>c33</td>
<td>Paul</td>
<td>3334433</td>
<td>05-04-2010</td>
<td>06-04-2010</td>
<td>0,4</td>
<td>0,40</td>
</tr>
<tr>
<td>10</td>
<td>mv102</td>
<td>You, Me and D.</td>
<td>2001</td>
<td>Anthony Russo</td>
<td>English</td>
<td>c26</td>
<td>Smith</td>
<td>4445467</td>
<td>22-03-2010</td>
<td>25-03-2010</td>
<td>0,3</td>
<td>0,90</td>
</tr>
</tbody>
</table>

Delete
Refactoring Spreadsheets

From the relational model, we generate a spreadsheet that simulates it, but in a spreadsheet environment:

(a) First part of the refactored properties spreadsheet.

(b) Second part of the refactored properties spreadsheet.
We have calculated the formal relationship between spreadsheets and relational databases;

This relationship was expressed using data refinement theory;

Using the 2LT framework (an implementation of data refinement theory), we can transform spreadsheets into databases and vice versa;

This theory provides functions to safely migrate the data back and forth.
Co-evolution of Spreadsheet Models and Instances

- We have encoded *ClassSheets* in the 2LT framework;
- In particular, we have encoded reference as type-safe projection functions;
- Thus, rules to spreadsheet evolution can be defined;
- We defined a set of common spreadsheet evolution steps:
  - Add/remove column;
  - Make a block expandable;
  - Split;
- This steps can be safely applied to spreadsheet models and data will be migrated automatically.
We believe that model can help end user be more efficient and effective;

We organized and run an empirical study with 38 participants;

These participants are students from the university, but non were studying engineering or computer science;

They were asked to do several tasks in different model-based spreadsheets;

We concluded that, in some case, our spreadsheets helped them being more effective and efficient.
We have developed a framework to integrate our techniques in a single platform;

It is composed by Haskell libraries, batch and online tools and OpenOffice.org extensions;

It can import and export spreadsheets in different formats;

It can be used to infer functional dependencies and the different models we presented before;

The model-based spreadsheets can also be generated with HaExcel.
Contributions

- We presented techniques to infer and reason about functional dependencies in the context of spreadsheets;
- Using the idiosyncrasies of spreadsheets, we presented techniques to automatic inference of relational schemas, *ClassSheets* and UML class diagrams;
- Using functional dependencies we can infer edit assistance for spreadsheets including, for example, the auto-completion of some columns;
- We calculated the formal relationship between spreadsheet models and relational schemas. Rules for the migration between these two fields were designed;
Contributions - cont.

- Based on a relational schema we are able to produce a new spreadsheet that is more organized than the original one and thus better for handling data;
- We improved 2LT to support spreadsheet models. We also develop a series of common evolution steps including, for example, insertion of a column in each instance of a model;
- A study with end users validating the results of our work is presented;
- All the techniques here presented are available under an open source framework, HaExcel, that can be reused in other projects.
Jácome Cunha, João Saraiva, Joost Visser. From spreadsheets to relational databases and back. PEPM '2009. 179–188.


