Spreadsheet Engineering

Jácome Cunha

jacome@di.uminho.pt

www.di.uminho.pt/~jacome

Researcher @

HASLab / INESC TEC & Universidade do Minho, Portugal

Oregon State University (office 2077)

OSU - EECS Colloquium - 02/24/14

Agenda

I. Motivation

II. Spreadsheets Meet Models

III. Models for Spreadsheets – ClassSheets

IV. Inferring ClassSheets

V. Embedding ClassSheets

VI. Evolution!

VII. Model-Driven Spreadsheets

VIII. Summary

I. Motivation

Why do Spreadsheets matter?

Financial intelligence firm CODA reports that 95% of all U.S. firms use spreadsheets for financial reporting.

Sarbanes-Oxley: What About all the Spreadsheets?, Raymond R. Panko and Nicholas Ordway, 2008

Why do Spreadsheets matter?

They are the programming language of choice by non-professional programmers, a.k.a. *end users.*

In the U.S. alone, the number of end-user programmers is conservatively estimated at 11 million, compared to only 2.75 million other, professional programmers.

Estimating the numbers of end users and end-user programmers, Christopher Scaffidi, Mary Shaw, and Brad Myers, VL/HCC 2005

Omnipresent

Easy-to-use

Multi-purpose

Flexible

Why do Spreadsheets matter?

In 2004, RevenueRecognition.com (now Softtrax) had the International Data Corporation interview 118 business leaders.

IDC found that 85% were using spreadsheets in financial reporting and forecasting.

Sarbanes-Oxley: What About all the Spreadsheets?, Raymond R. Panko and Nicholas Ordway, 2008

In fact, spreadsheets lack:

- Abstraction
- Encapsulation
- Type system
- Testing
- IDE
- ...

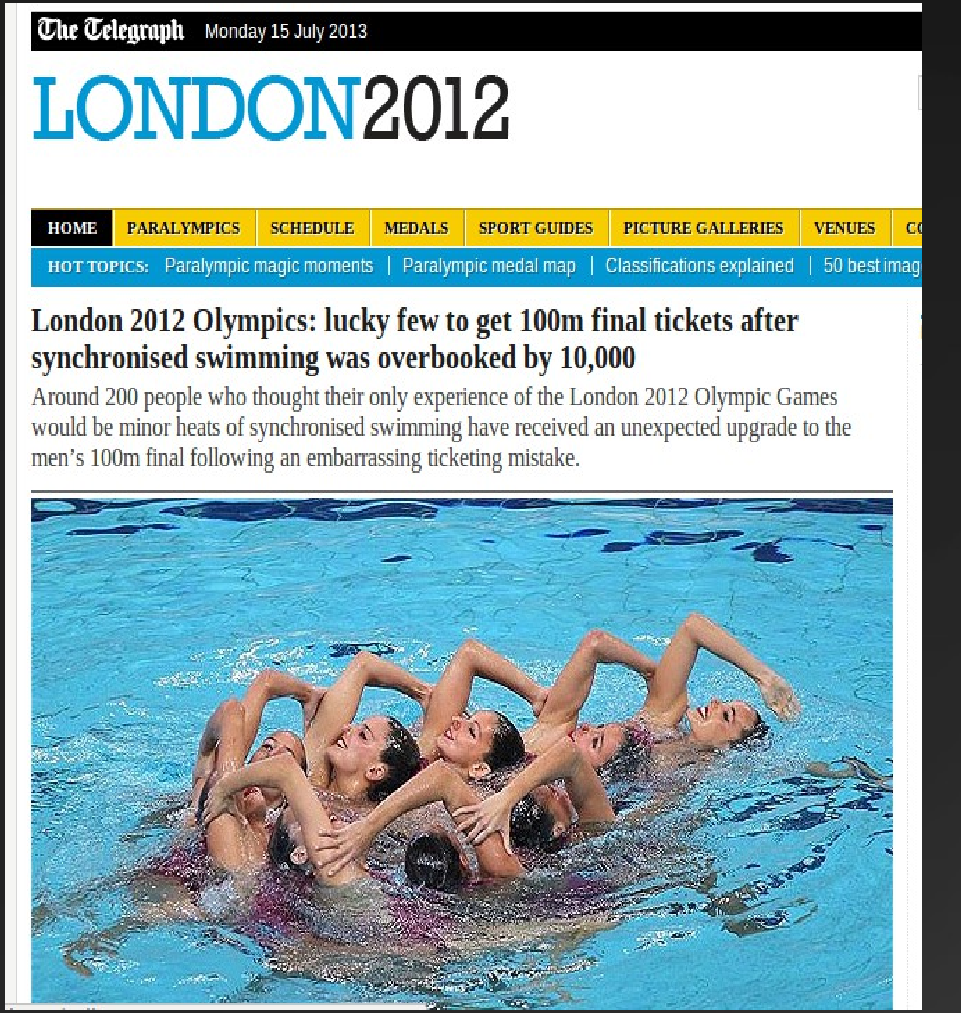
And the consequences may be...

Around 200 people who thought their only experience of the London 2012 Olympic Games would be minor heats of synchronised swimming have received an unexpected upgrade to the men’s 100m final following an embarrassing ticketing mistake.

...

*Locog said the error occurred in the summer, between the first and second round of ticket sales, when a member of staff made a single keystroke mistake and entered **‘20,000’ into a spreadsheet rather than the correct figure of 10,000** remaining tickets.*

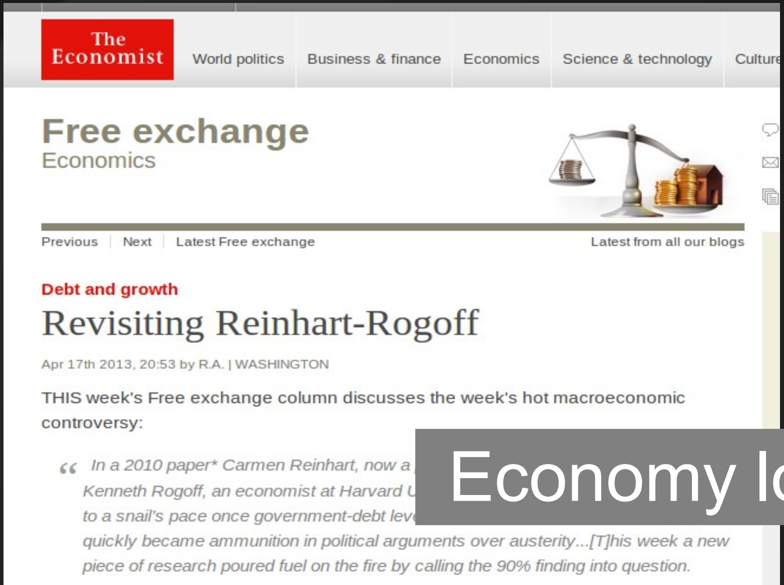
The Telegraph, 04 January 2012



London 2012 Olympics: lucky few to get 100m final tickets after synchronised swimming was overbooked by 10,000

Around 200 people who thought their only experience of the London 2012 Olympic Games would be minor heats of synchronised swimming have received an unexpected upgrade to the men’s 100m final following an embarrassing ticketing mistake.

And the consequences may be...



Free exchange
Economics

Debt and growth

Revisiting Reinhart-Rogoff

Apr 17th 2013, 20:53 by R.A. | WASHINGTON

THIS week's Free exchange column discusses the week's hot macroeconomic controversy:

> “ *In a 2010 paper* Carmen Reinhart, now a Kenneth Rogoff, an economist at Harvard U to a snail's pace once government-debt lev quickly became ammunition in political arguments over austerity...[T]his week a new piece of research poured fuel on the fire by calling the 90% finding into question.*

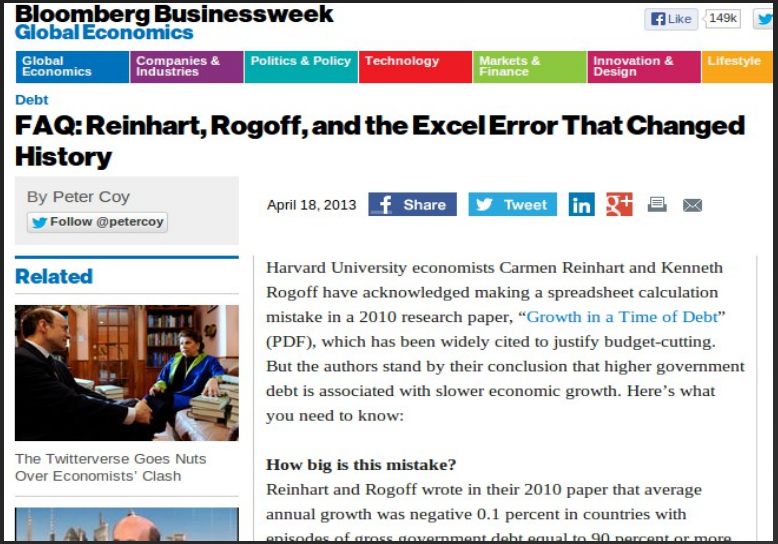
In a 2010 paper Carmen Reinhart, now a professor at Harvard Kennedy School, and Kenneth Rogoff, an economist at Harvard University...argued that GDP growth slows to a snail's pace once government-debt levels exceed 90% of GDP. The 90% figure quickly became ammunition in political arguments over austerity...This week a new piece of research poured fuel on the fire by calling the 90% finding into question..*

Economy losses of $10 billion/year!

The Economist, 17 April 2013

Harvard University economists Carmen Reinhart and Kenneth Rogoff have acknowledged making a spreadsheet calculation mistake in a 2010 research paper, “Growth in a Time of Debt”, which has been widely cited to justify budget-cutting.

Business Week, 18 April 2013



Debt

FAQ: Reinhart, Rogoff, and the Excel Error That Changed History

By Peter Coy

April 18, 2013

Related

The Twitterverse Goes Nuts Over Economists' Clash

Harvard University economists Carmen Reinhart and Kenneth Rogoff have acknowledged making a spreadsheet calculation mistake in a 2010 research paper, “Growth in a Time of Debt” (PDF), which has been widely cited to justify budget-cutting. But the authors stand by their conclusion that higher government debt is associated with slower economic growth. Here's what you need to know:

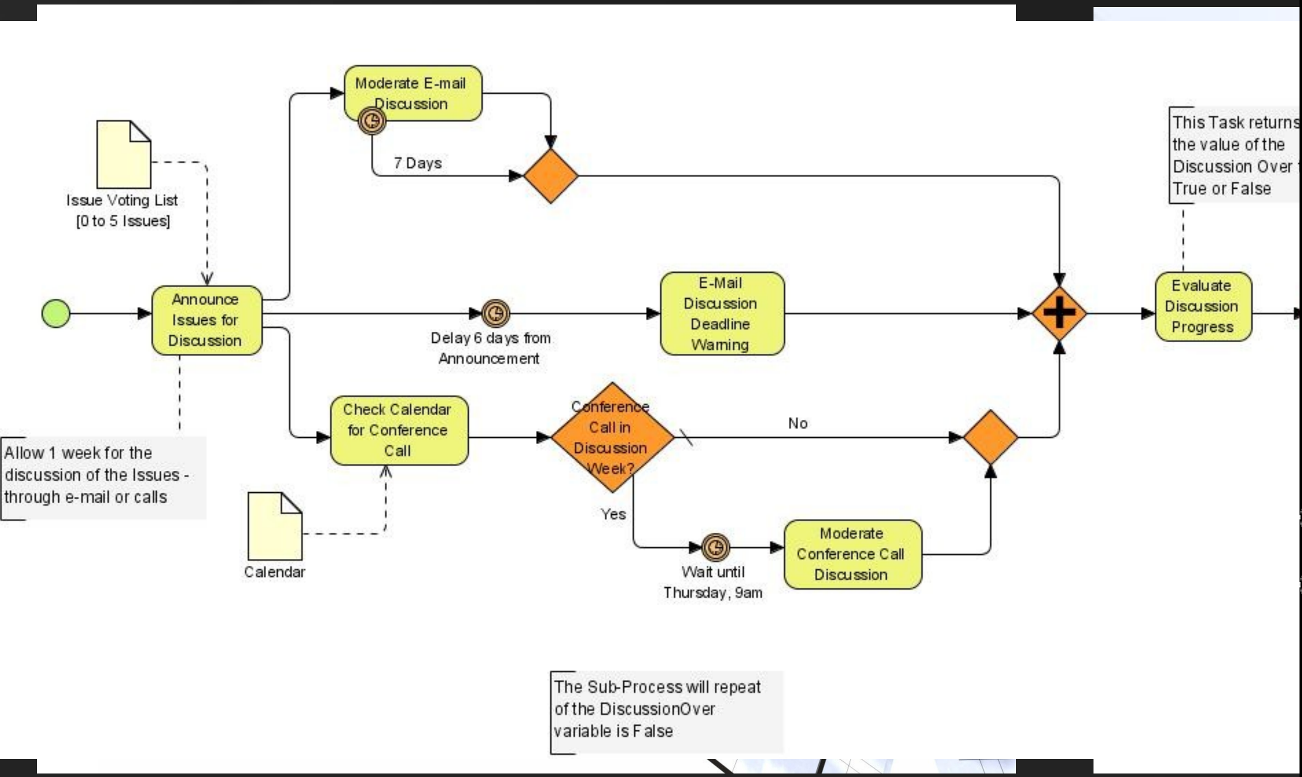
How big is this mistake?

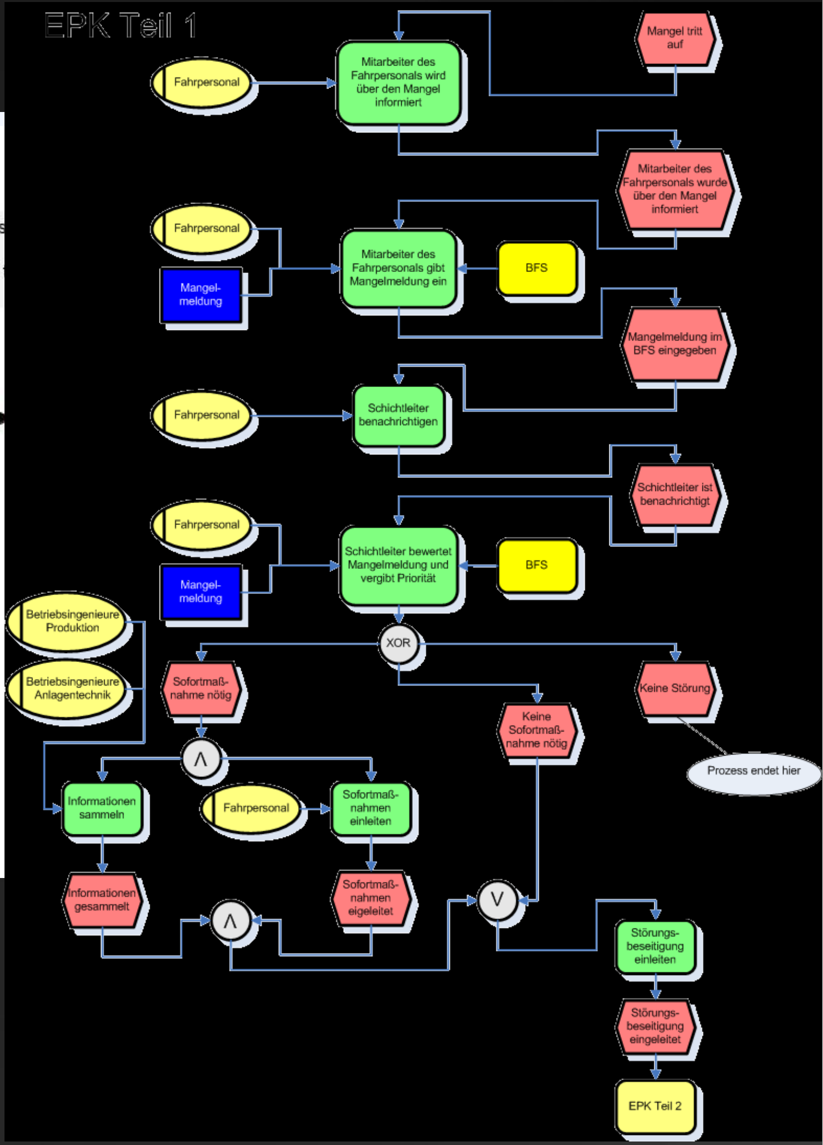
Reinhart and Rogoff wrote in their 2010 paper that average annual growth was negative 0.1 percent in countries with episodes of gross government debt equal to 90 percent or more

II. Spreadsheets Meet Models

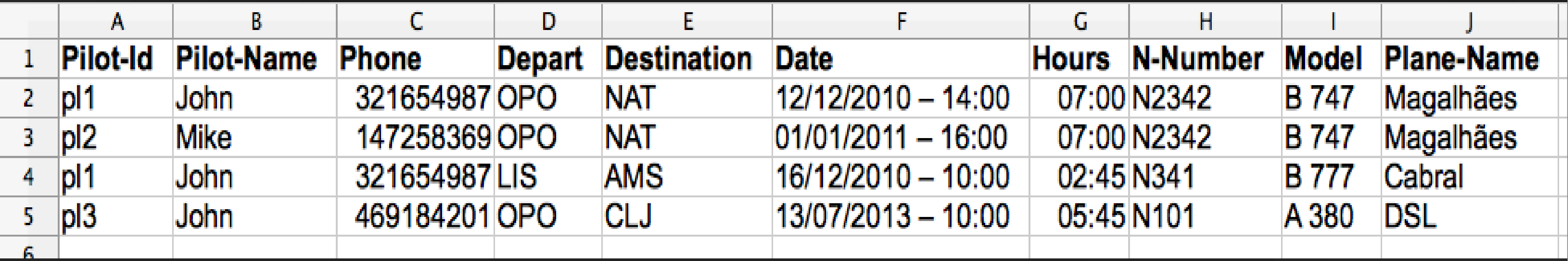
[PEPM'09, VL/HCC'10]

Why Models?


Moderate E-mail Discussion
7 Days
Issue Voting List
[0 to 5 Issues]
Announce Issues for Discussion
Delay 6 days from Announcement
E-Mail Discussion Deadline Warning
This Task returns the value of the Discussion Over True or False
Evaluate Discussion Progress
Check Calendar for Conference Call
Conference Call in Discussion Week?
No
Yes
Allow 1 week for the discussion of the Issues - through e-mail or calls
Calendar
Wait until Thursday, 9am
Moderate Conference Call Discussion
The Sub-Process will repeat of the DiscussionOver variable is False

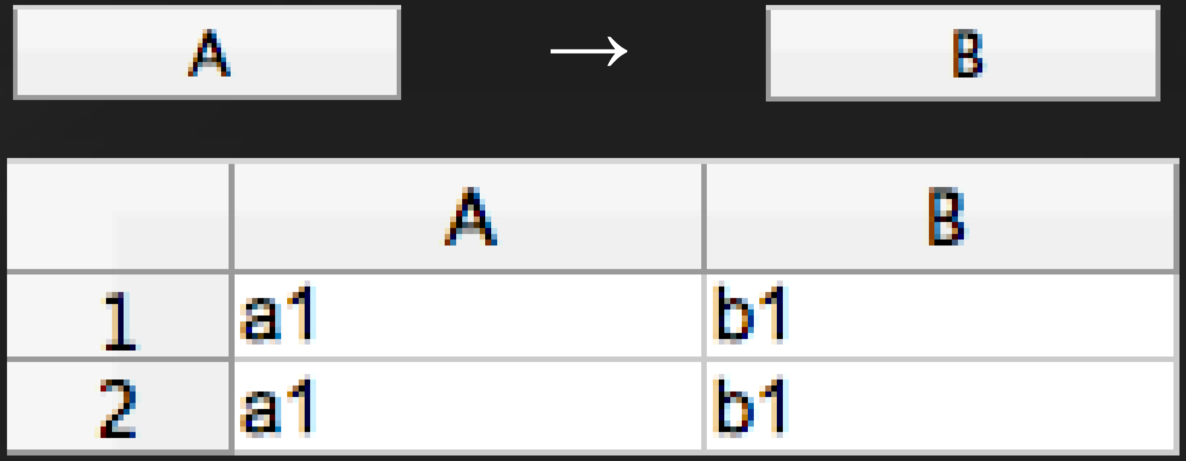

EPK Teil 1
Mangel tritt auf
Fahrpersonal
Mitarbeiter des Fahrpersonals wird über den Mangel informiert
Mitarbeiter des Fahrpersonals wurde über den Mangel informiert
Fahrpersonal
Mangel-meldung
Mitarbeiter des Fahrpersonals gibt Mangelmeldung ein
BFS
Mangelmeldung im BFS eingegeben
Fahrpersonal
Schichtleiter benachrichtigen
Schichtleiter ist benachrichtigt
Fahrpersonal
Mangel-meldung
Schichtleiter bewertet Mangelmeldung und vergibt Priorität
BFS
Betriebsingenieure Produktion
Betriebsingenieure Anlagentechnik
XOR
Sofortmaß-nahme nötig
Keine Störung
Keine Sofortmaß-nahme nötig
Prozess endet hier
Λ
Informationen sammeln
Fahrpersonal
Sofortmaß-nahmen einleiten
Informationen gesammelt
Λ
Sofortmaß-nahmen eigeleitet
V
Störungs-beseitigung einleiten
Störungs-beseitigung eingeleitet
EPK Teil 2

Spreadsheet Example



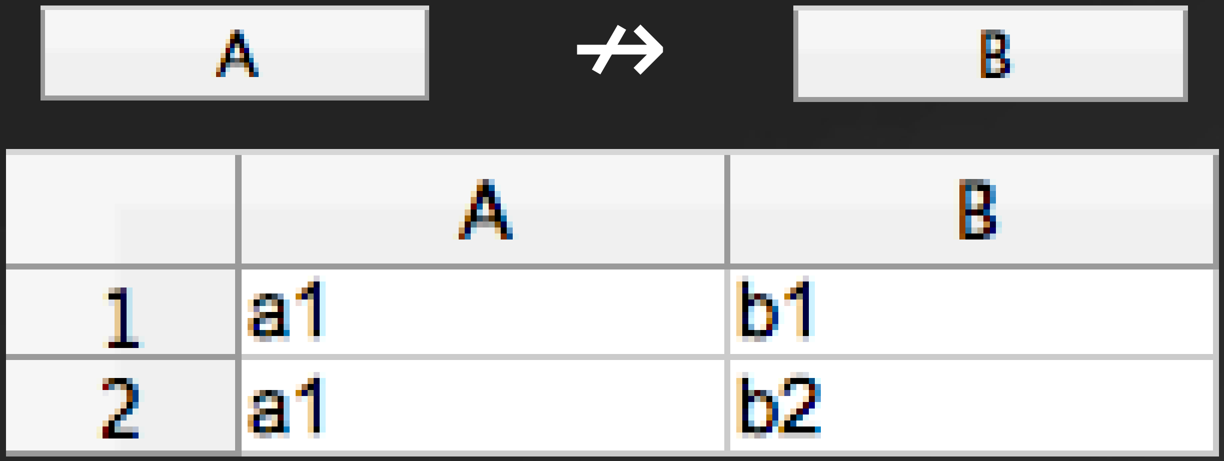
	A	B	C	D	E	F	G	H	I	J
1	**Pilot-Id**	**Pilot-Name**	**Phone**	**Depart**	**Destination**	**Date**	**Hours**	**N-Number**	**Model**	**Plane-Name**
2	pl1	John	321654987	OPO	NAT	12/12/2010 – 14:00	07:00	N2342	B 747	Magalhães
3	pl2	Mike	147258369	OPO	NAT	01/01/2011 – 16:00	07:00	N2342	B 747	Magalhães
4	pl1	John	321654987	LIS	AMS	16/12/2010 – 10:00	02:45	N341	B 777	Cabral
5	pl3	John	469184201	OPO	CLJ	13/07/2013 – 10:00	05:45	N101	A 380	DSL
6										

Functional Dependency?



A → B

	A	B
1	a1	b1
2	a1	b1



A ↛ B

	A	B
1	a1	b1
2	a1	b2

Functional Dependencies

- We compute the business logic from the data, by inferring FDs
- They are the building blocks inferring models for (legacy) spreadsheets
- The better the FDs we infer, the better the model we compute!

Too Many??

```
["A"] -> ["B","C","D","E","F"]

["C"] -> ["A","B","D","E","F"]

["D"] -> ["A","B","C","E","F"]

["E"] -> ["A","B","C","D","F"]

["F"] -> ["A","B","C","D","E"]

["G"] -> ["H","I","J"]

["H"] -> ["G","I","J"]

["I"] -> ["G","H","J"]

["J"] -> ["G","H","I"]

["K"] -> ["L","M"]

["L"] -> ["K","M"]

["M"] -> ["K","L"]

["B","K"] -> ["A","C","D","E","F"]

["B","L"] -> ["A","C","D","E","F"]

["B","M"] -> ["A","C","D","E","F"]
```

Accidents happen

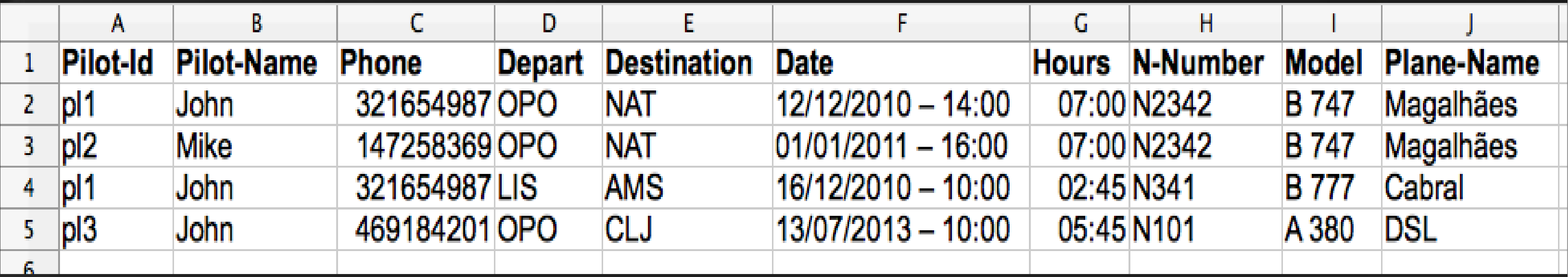
- We use a data mining algorithm which produces to many accidental FDs!

- We introduce some spreadsheet specific heuristics to filter out “accidental” FDs

Organize them

- **Label semantics**: often keys are labeled “code” or “id”
- **Label arrangement**: we prefer FDs respecting the order of columns
- **Antecedent size**: small keys are preferable
- **Ratio**: small ratio between keys and non-keys
- **Single value columns**: columns always with the same value appear in too many FDs

Final set



	A	B	C	D	E	F	G	H	I	J
1	**Pilot-Id**	**Pilot-Name**	**Phone**	**Depart**	**Destination**	**Date**	**Hours**	**N-Number**	**Model**	**Plane-Name**
2	pl1	John	321654987	OPO	NAT	12/12/2010 – 14:00	07:00	N2342	B 747	Magalhães
3	pl2	Mike	147258369	OPO	NAT	01/01/2011 – 16:00	07:00	N2342	B 747	Magalhães
4	pl1	John	321654987	LIS	AMS	16/12/2010 – 10:00	02:45	N341	B 777	Cabral
5	pl3	John	469184201	OPO	CLJ	13/07/2013 – 10:00	05:45	N101	A 380	DSL
6										

Pilot-Id → Pilot-Name, Phone

N-Number → Model, Plane-Name

Pilot-Id, N-Number, Depart, Destination, Date, Hours → {}

The first model: a relational model

Having computed the FDs, we can now use the FUN algorithm to produce a *relational model* for the spreadsheet:

***Pilots* (Pilot-Id, Pilot-Name, Phone)**

***Planes* (N-Number, Model, Plane-Name)**

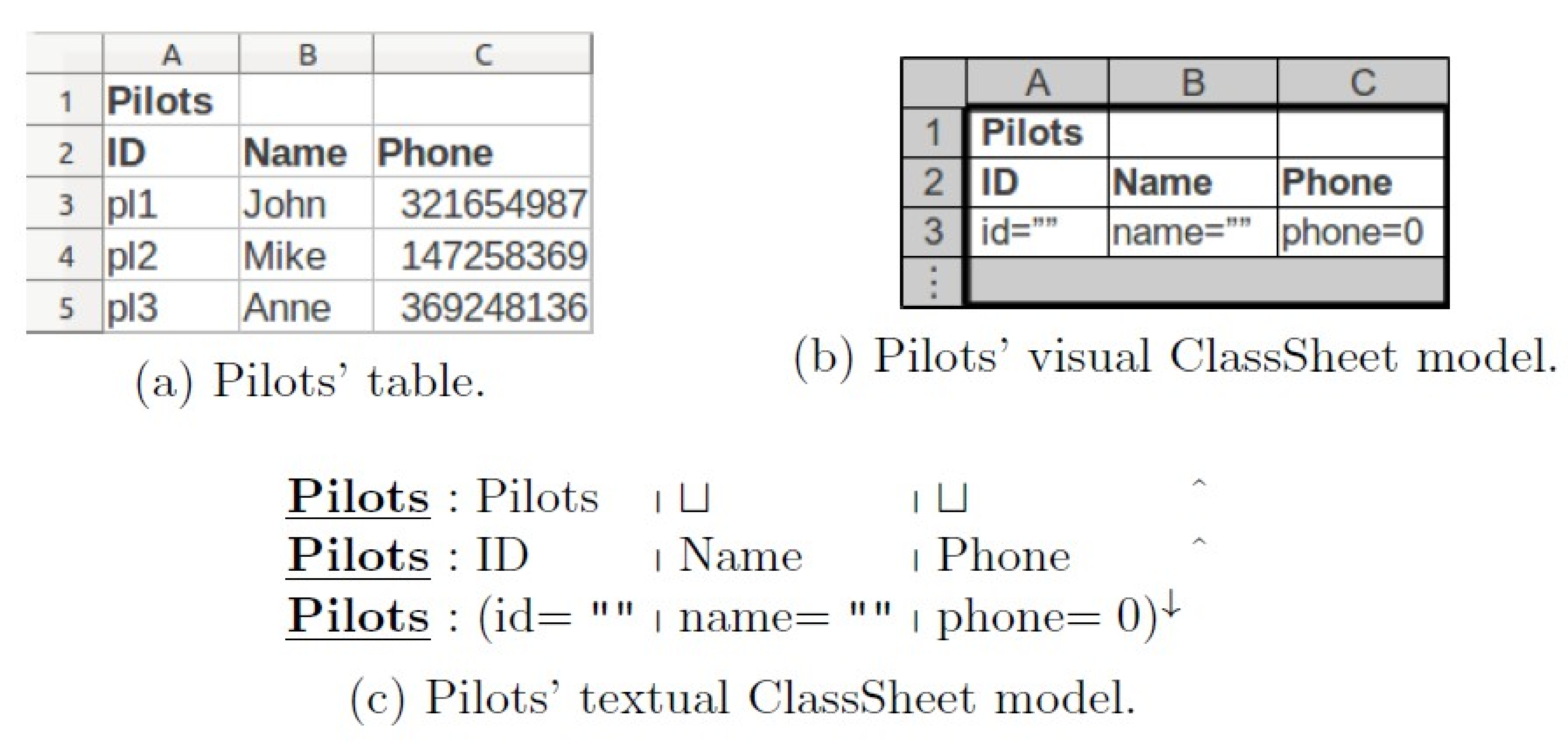
<*Flights*> (#Pilot-Id,# N-Number, Depart, Destination, Date Hours)

III. Models for Spreadsheet – ClassSheets

Engels and Erwig ASE'05

ClassSheets - Models for Spreadsheets

ClassSheets are a high-level, object-oriented formalism to specify spreadsheets



	A	B	C
1	**Pilots**		
2	**ID**	**Name**	**Phone**
3	pl1	John	321654987
4	pl2	Mike	147258369
5	pl3	Anne	369248136

(a) Pilots' table.

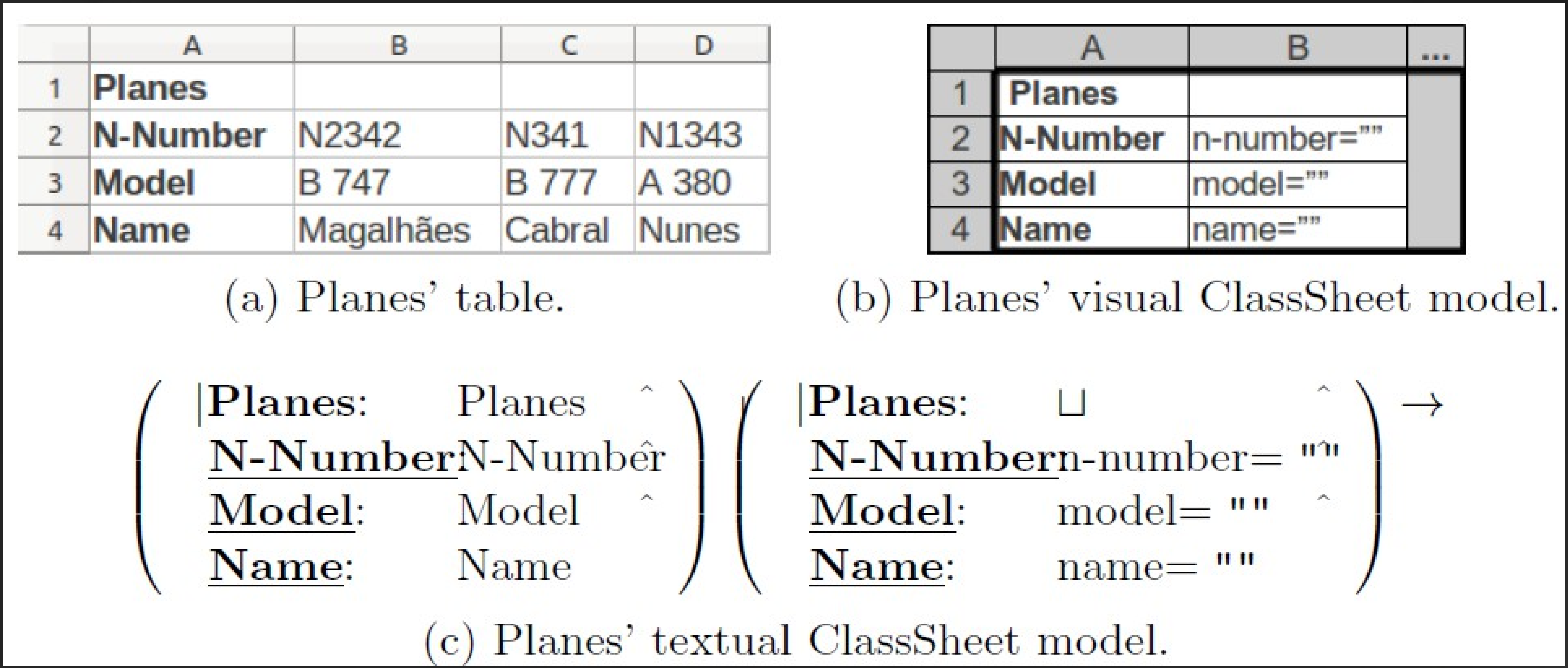
	A	B	C
1	**Pilots**		
2	**ID**	**Name**	**Phone**
3	id=""	name=""	phone=0
⋮			

(b) Pilots' visual ClassSheet model.

$$\begin{array}{llll}
\underline{\textbf{Pilots}} : \text{Pilots} & \mid \sqcup & \mid \sqcup & \hat{} \\
\underline{\textbf{Pilots}} : \text{ID} & \mid \text{Name} & \mid \text{Phone} & \hat{} \\
\underline{\textbf{Pilots}} : (\text{id}= \texttt{""} & \mid \text{name}= \texttt{""} & \mid \text{phone}= 0)^{\downarrow} &
\end{array}$$

(c) Pilots' textual ClassSheet model.

ClassSheets - Models for Spreadsheets



	A	B	C	D
1	**Planes**			
2	**N-Number**	N2342	N341	N1343
3	**Model**	B 747	B 777	A 380
4	**Name**	Magalhães	Cabral	Nunes

(a) Planes' table.

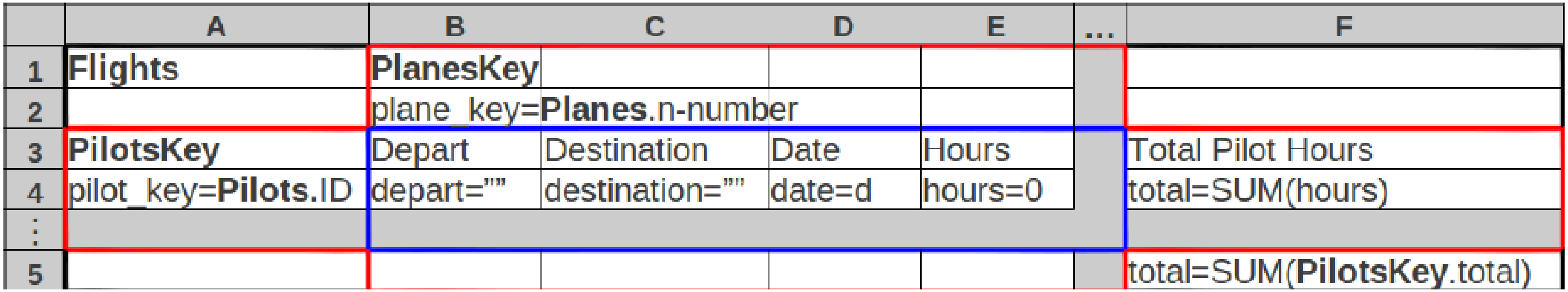
	A	B	...
1	**Planes**		
2	**N-Number**	n-number=""	
3	**Model**	model=""	
4	**Name**	name=""	

(b) Planes' visual ClassSheet model.

$$\left(\begin{array}{lll} |\textbf{Planes}: & \text{Planes} & \hat{} \\ \underline{\textbf{N-Number}} & \text{N-Number} & \hat{} \\ \underline{\textbf{Model}}: & \text{Model} & \hat{} \\ \underline{\textbf{Name}}: & \text{Name} & \end{array}\right) \left(\begin{array}{lll} |\textbf{Planes}: & \sqcup & \hat{} \\ \underline{\textbf{N-Number}} & \text{n-number= ""} & \hat{} \\ \underline{\textbf{Model}}: & \text{model= ""} & \hat{} \\ \underline{\textbf{Name}}: & \text{name= ""} & \end{array}\right) \rightarrow$$

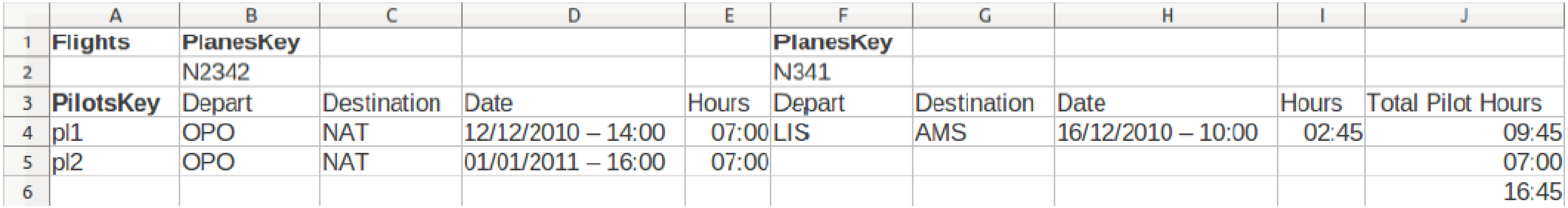
(c) Planes' textual ClassSheet model.

ClassSheets - Models for Spreadsheets



	A	B	C	D	E	...	F
1	**Flights**	**PlanesKey**					
2		plane_key=**Planes**.n-number					
3	**PilotsKey**	Depart	Destination	Date	Hours		Total Pilot Hours
4	pilot_key=**Pilots**.ID	depart=""	destination=""	date=d	hours=0		total=SUM(hours)
⋮							
5							total=SUM(**PilotsKey**.total)

(a) Flights' visual ClassSheet model.



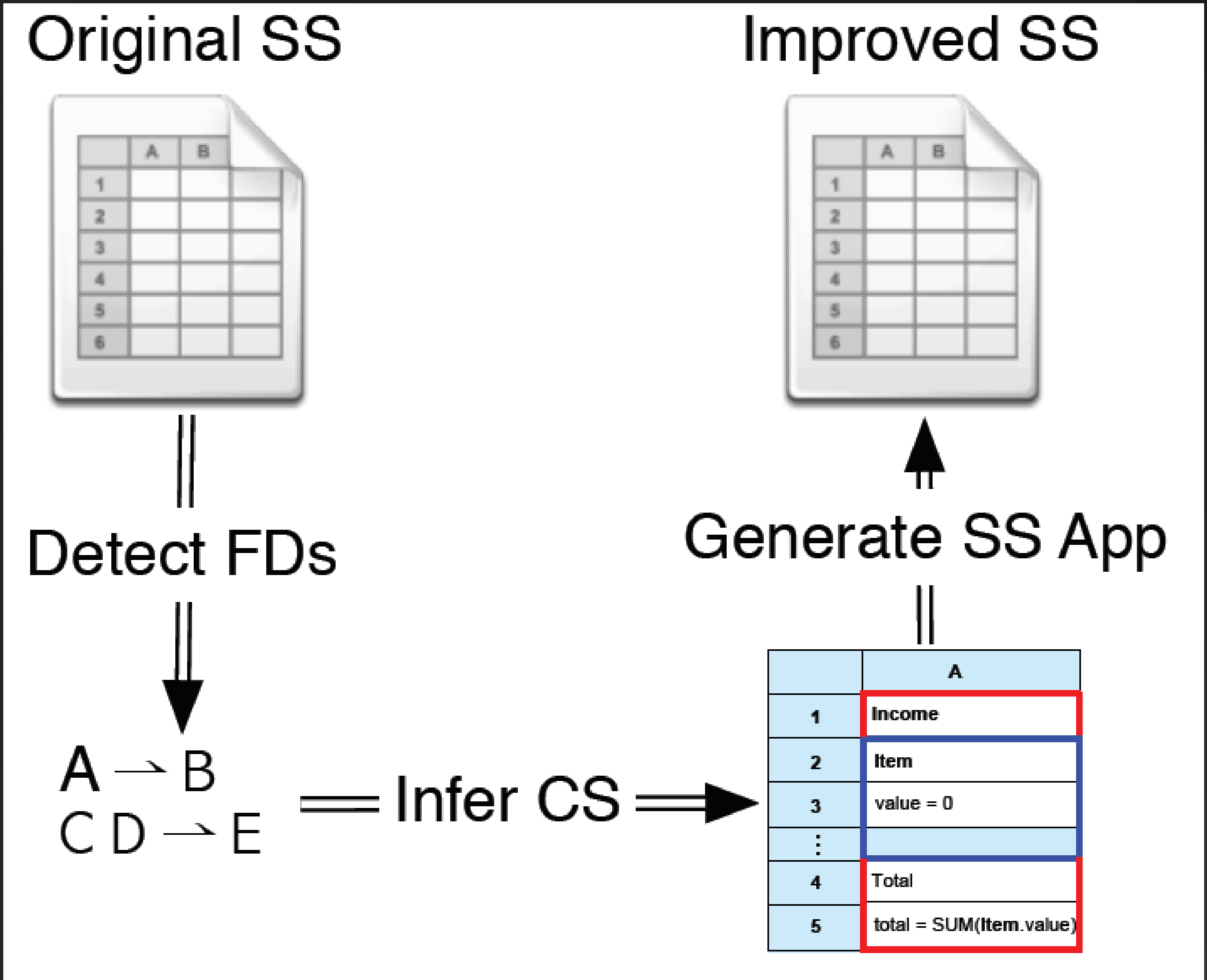
	A	B	C	D	E	F	G	H	I	J
1	**Flights**	**PlanesKey**				**PlanesKey**				
2		N2342				N341				
3	**PilotsKey**	Depart	Destination	Date	Hours	Depart	Destination	Date	Hours	Total Pilot Hours
4	pl1	OPO	NAT	12/12/2010 – 14:00	07:00	LIS	AMS	16/12/2010 – 10:00	02:45	09:45
5	pl2	OPO	NAT	01/01/2011 – 16:00	07:00					07:00
6										16:45

(b) Flights' table.

IV. Inferring ClassSheets

[VL/HCC'10]

ClassSheet Inference



Original SS

Improved SS

Detect FDs

A → B

C D → E

Infer CS

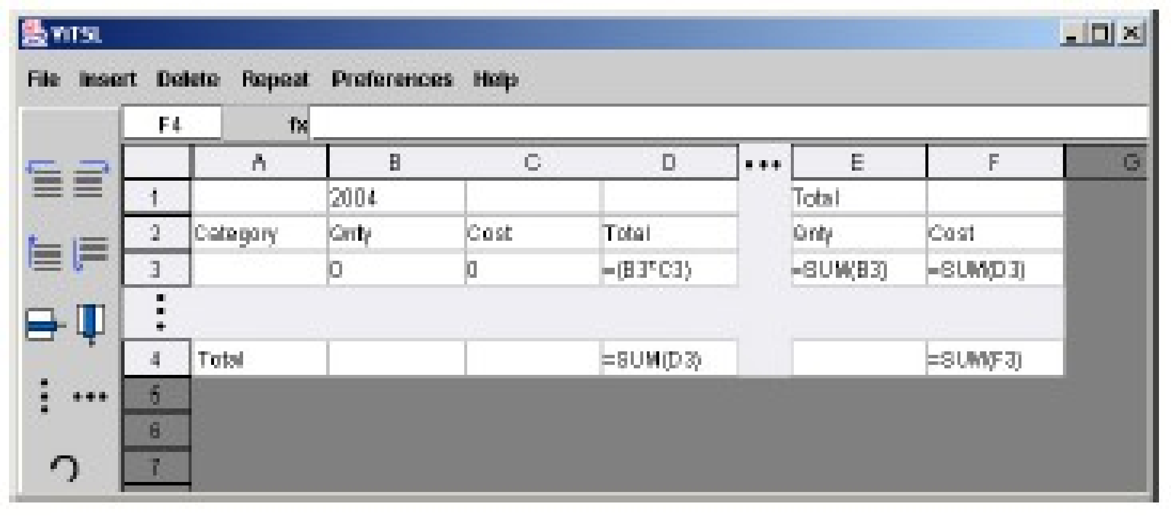
Generate SS App

	A
1	Income
2	Item
3	value = 0
⋮	
4	Total
5	total = SUM(Item.value)

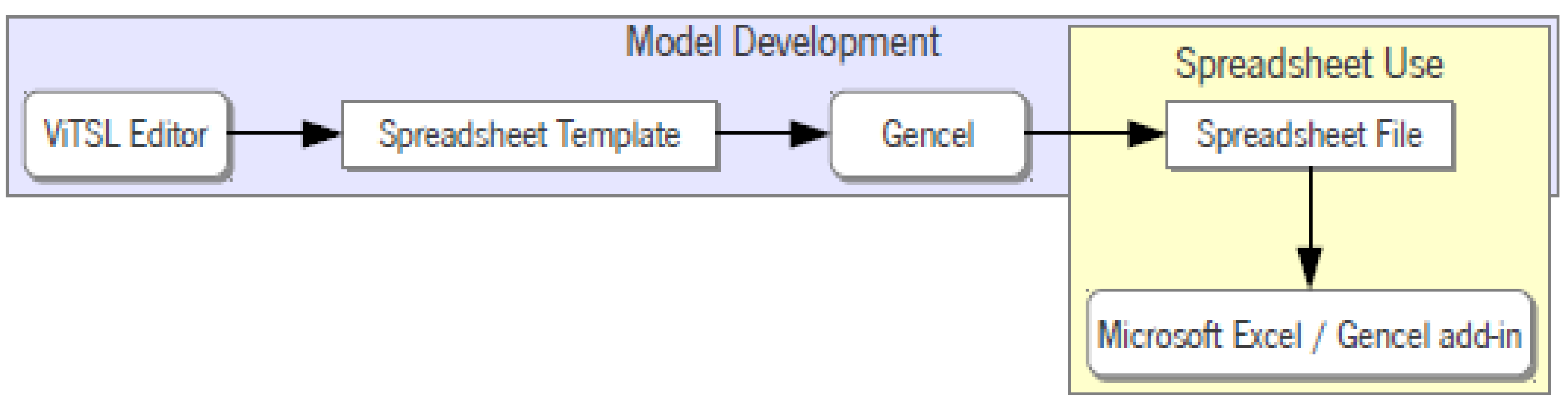
V. Embedding ClassSheets

[VL/HCC'11]

Why the Embedding?



	A	B	C	D	•••	E	F	G
1		2004				Total		
2	Category	Qnty	Cost	Total		Qnty	Cost	
3		0	0	=(B3*C3)		=SUM(B3)	=SUM(D3)	
⋮								
4	Total			=SUM(D3)			=SUM(F3)	
5								
6								
7								

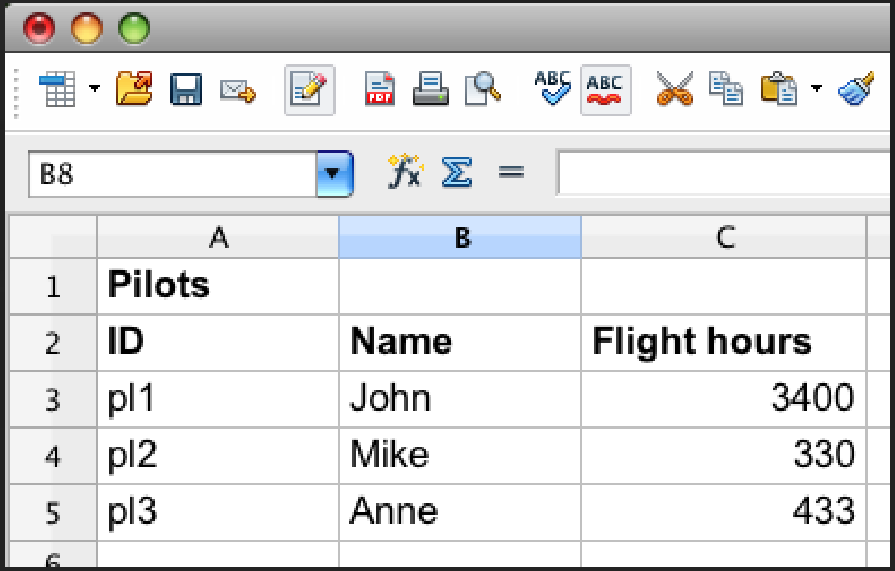


Embedding...

- Embedding a language into another language is a recurring strategy (e.g. for DSLs)
  - Embedded language inherit all the power of the host language :-)
  - Users are used to the host language and do not need to learn a (complete) new language :-)
  - Implementation effort is much reduced :-)
  - It may have some restrictions :-(

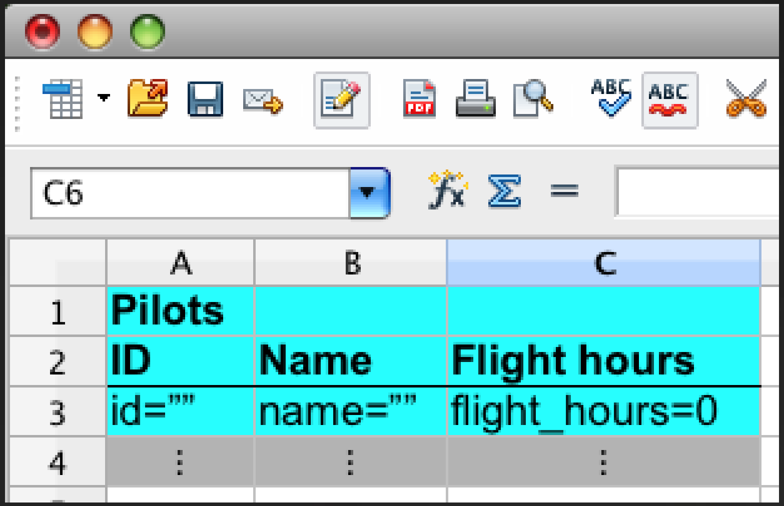
- We embedded *ClassSheets* in traditional *spreadsheet systems*

Vertically Expandable Tables



B8

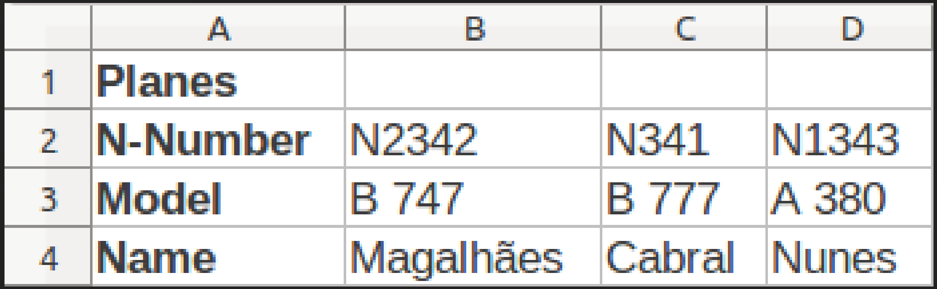
	A	B	C
1	**Pilots**		
2	**ID**	**Name**	**Flight hours**
3	pl1	John	3400
4	pl2	Mike	330
5	pl3	Anne	433



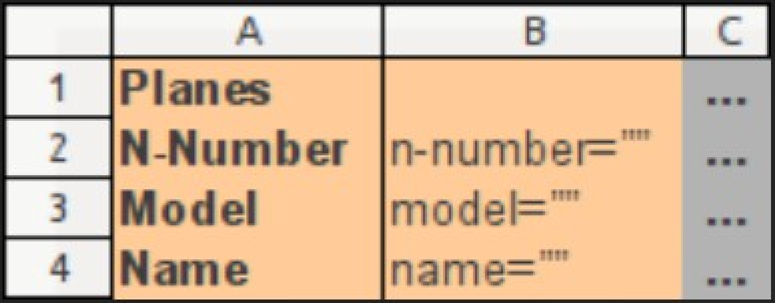
C6

	A	B	C
1	**Pilots**		
2	**ID**	**Name**	**Flight hours**
3	id=""	name=""	flight_hours=0
4	⋮	⋮	⋮

Horizontally Expandable Tables

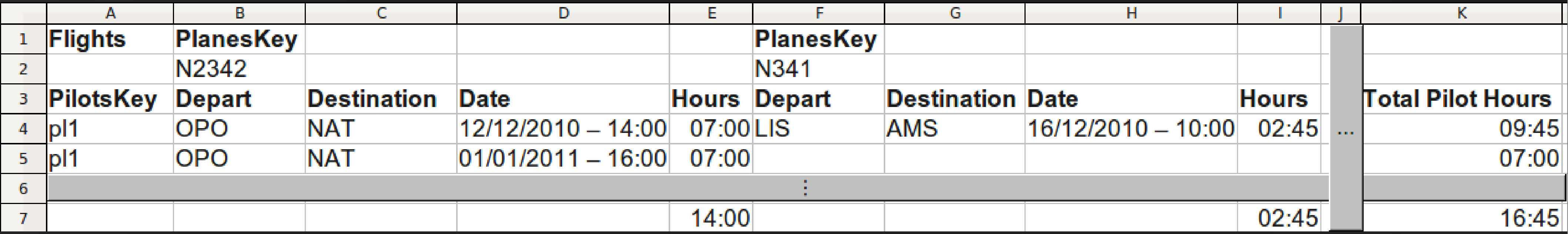


	A	B	C	D
1	**Planes**			
2	**N-Number**	N2342	N341	N1343
3	**Model**	B 747	B 777	A 380
4	**Name**	Magalhães	Cabral	Nunes

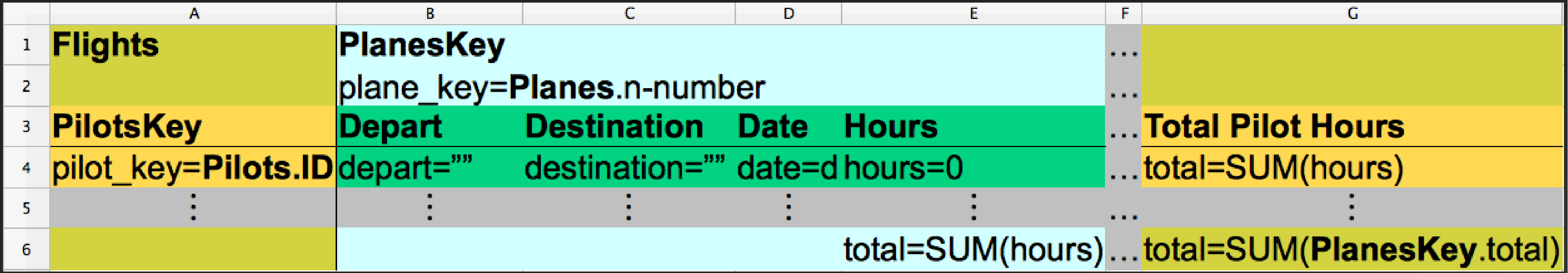


	A	B	C
1	**Planes**		...
2	**N-Number**	n-number=""	...
3	**Model**	model=""	...
4	**Name**	name=""	...

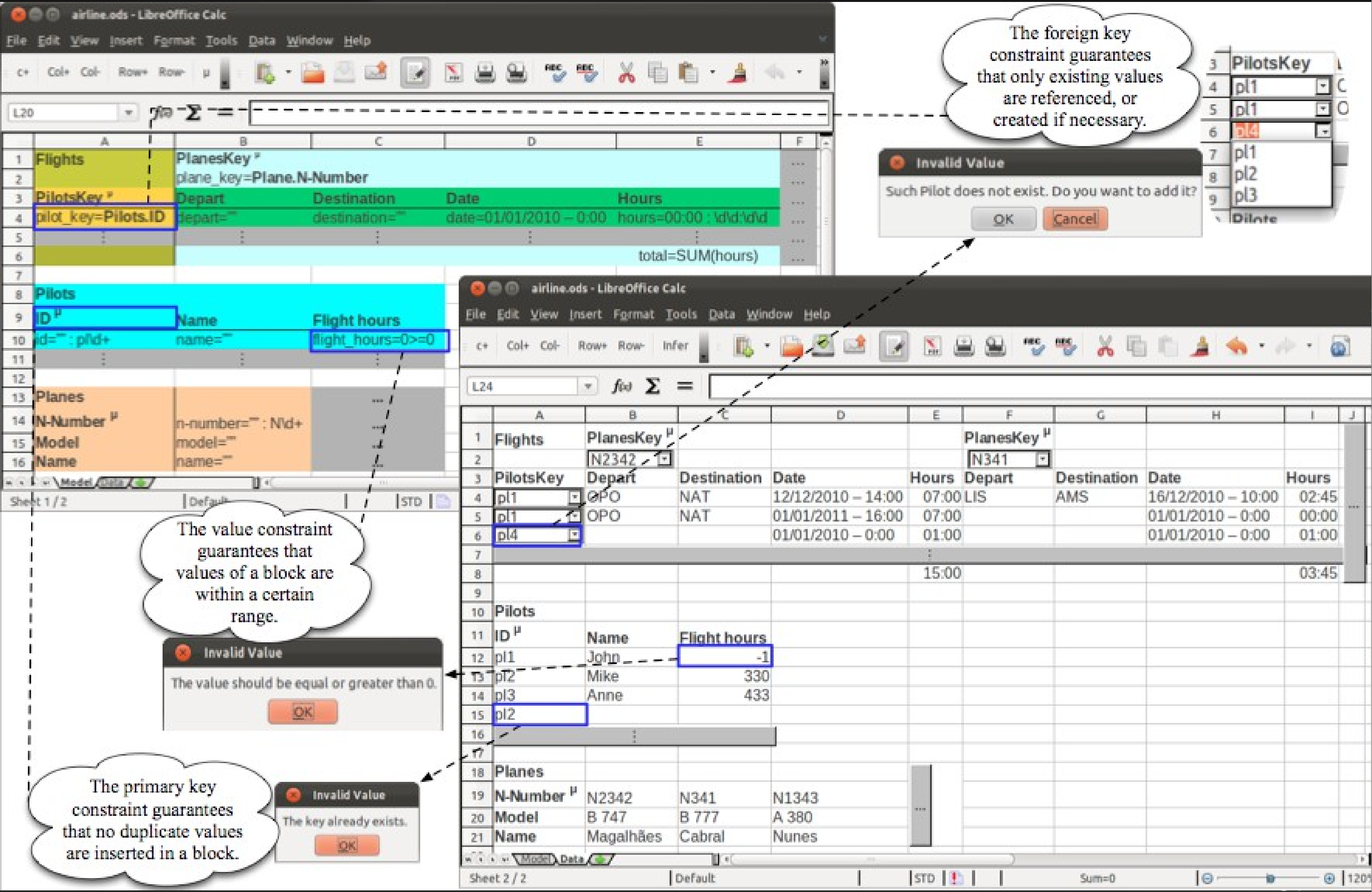
Relationship Tables



	A	B	C	D	E	F	G	H	I	J	K
1	**Flights**	**PlanesKey**				**PlanesKey**					
2		N2342				N341					
3	**PilotsKey**	**Depart**	**Destination**	**Date**	**Hours**	**Depart**	**Destination**	**Date**	**Hours**		**Total Pilot Hours**
4	pl1	OPO	NAT	12/12/2010 – 14:00	07:00	LIS	AMS	16/12/2010 – 10:00	02:45	...	09:45
5	pl1	OPO	NAT	01/01/2011 – 16:00	07:00						07:00
6						⋮					
7					14:00				02:45		16:45



	A	B	C	D	E	F	G
1	**Flights**	**PlanesKey**				…	
2		plane_key=**Planes**.n-number				…	
3	**PilotsKey**	**Depart**	**Destination**	**Date**	**Hours**	…	**Total Pilot Hours**
4	pilot_key=**Pilots.ID**	depart=""	destination=""	date=d	hours=0	…	total=SUM(hours)
5	⋮	⋮	⋮	⋮	⋮	…	⋮
6					total=SUM(hours)	…	total=SUM(**PlanesKey**.total)



airline.ods - LibreOffice Calc
File Edit View Insert Format Tools Data Window Help
L20

	A	B	C	D	E	F
1	Flights	PlanesKey ᵘ				...
2		plane_key=Plane.N-Number				...
3	PilotsKey ᵘ	Depart	Destination	Date	Hours	...
4	pilot_key=Pilots.ID	depart=""	destination=""	date=01/01/2010 – 0:00	hours=00:00 : \d\d:\d\d	...
5	⋮	⋮	⋮	⋮	⋮	...
6					total=SUM(hours)	...
7						
8	Pilots					
9	ID ᵘ	Name	Flight hours			
10	id="" : pl\d+	name=""	flight_hours=0>=0			
11	⋮	⋮	⋮			
12						
13	Planes		...			
14	N-Number ᵘ	n-number="" : N\d+	...			
15	Model	model=""	...			
16	Name	name=""	...			

Model
Data
Sheet 1 / 2
Default
STD

The foreign key constraint guarantees that only existing values are referenced, or created if necessary.

	PilotsKey
3	PilotsKey
4	pl1
5	pl1
6	pl4
7	pl1
8	pl2
9	pl3

Invalid Value

Such Pilot does not exist. Do you want to add it?

airline.ods - LibreOffice Calc
File Edit View Insert Format Tools Data Window Help
L24

	A	B	C	D	E	F	G	H	I
1	Flights	PlanesKey ᵘ				PlanesKey ᵘ			
2		N2342				N341			
3	PilotsKey	Depart	Destination	Date	Hours	Depart	Destination	Date	Hours
4	pl1	OPO	NAT	12/12/2010 – 14:00	07:00	LIS	AMS	16/12/2010 – 10:00	02:45
5	pl1	OPO	NAT	01/01/2011 – 16:00	07:00			01/01/2010 – 0:00	00:00
6	pl4			01/01/2010 – 0:00	01:00			01/01/2010 – 0:00	01:00
7					⋮				
8					15:00				03:45
9									
10	Pilots								
11	ID ᵘ	Name	Flight hours						
12	pl1	John	-1						
13	pl2	Mike	330						
14	pl3	Anne	433						
15	pl2								
16		⋮							
17									
18	Planes								
19	N-Number ᵘ	N2342	N341	N1343	...				
20	Model	B 747	B 777	A 380					
21	Name	Magalhães	Cabral	Nunes					

The value constraint guarantees that values of a block are within a certain range.

Invalid Value

The value should be equal or greater than 0.

The primary key constraint guarantees that no duplicate values are inserted in a block.

Invalid Value

The key already exists.

VI. Evolution!

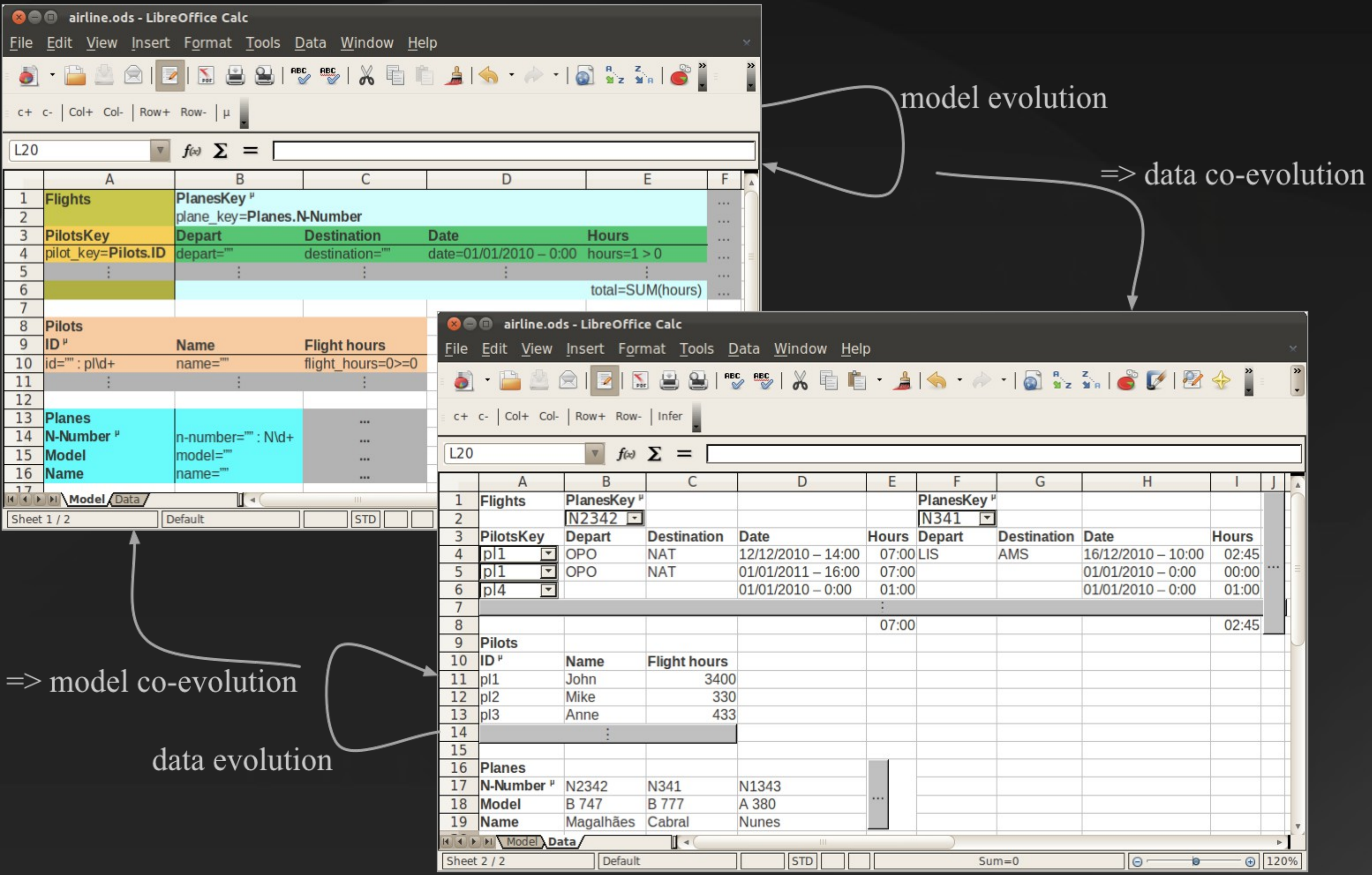
[FASE'11, ICMT'12]

Why do Spreadsheet Models Need Evolution?

- Suppose now you need to add new information to the spreadsheet
- For instance, the number of passengers of each flight
- It would require to do several error-prone tasks
- Add columns, labels, update formulas, etc.
- We can do it automatically!

Why do Spreadsheet Instances Need Evolution?

- Some evolution steps are easier to perform on the instance
- For instance, to add a column to one of the repetition blocks
- People felt the need to evolve the data



airline.ods - LibreOffice Calc
File Edit View Insert Format Tools Data Window Help
c+ c- Col+ Col- Row+ Row- µ
L20
Model Data
Sheet 1 / 2
Default
STD
airline.ods - LibreOffice Calc
File Edit View Insert Format Tools Data Window Help
c+ c- Col+ Col- Row+ Row- Infer
L20
Model Data
Sheet 2 / 2
Default
STD
Sum=0
120%

model evolution

=> data co-evolution

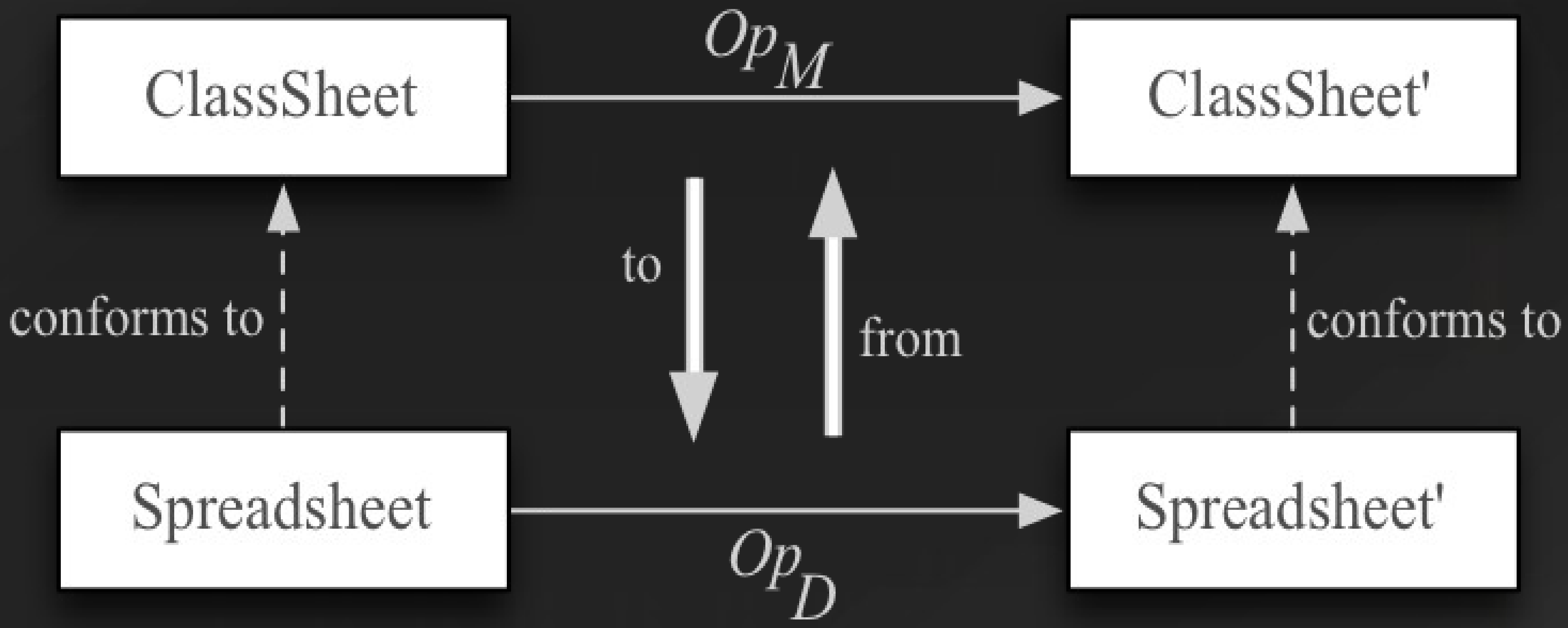
=> model co-evolution

data evolution

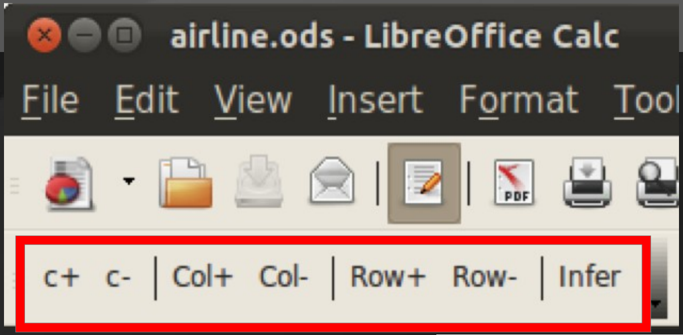
	A	B	C	D	E	F
1	Flights	PlanesKey ᵖ				...
2		plane_key=Planes.N-Number				...
3	PilotsKey	Depart	Destination	Date	Hours	...
4	pilot_key=Pilots.ID	depart=""	destination=""	date=01/01/2010 – 0:00	hours=1 > 0	...
5	⋮	⋮	⋮	⋮	⋮	...
6					total=SUM(hours)	...
7						
8	Pilots					
9	ID ᵖ	Name	Flight hours			
10	id="" : pl\d+	name=""	flight_hours=0>=0			
11	⋮	⋮	⋮			
12						
13	Planes		...			
14	N-Number ᵖ	n-number="" : N\d+	...			
15	Model	model=""	...			
16	Name	name=""	...			

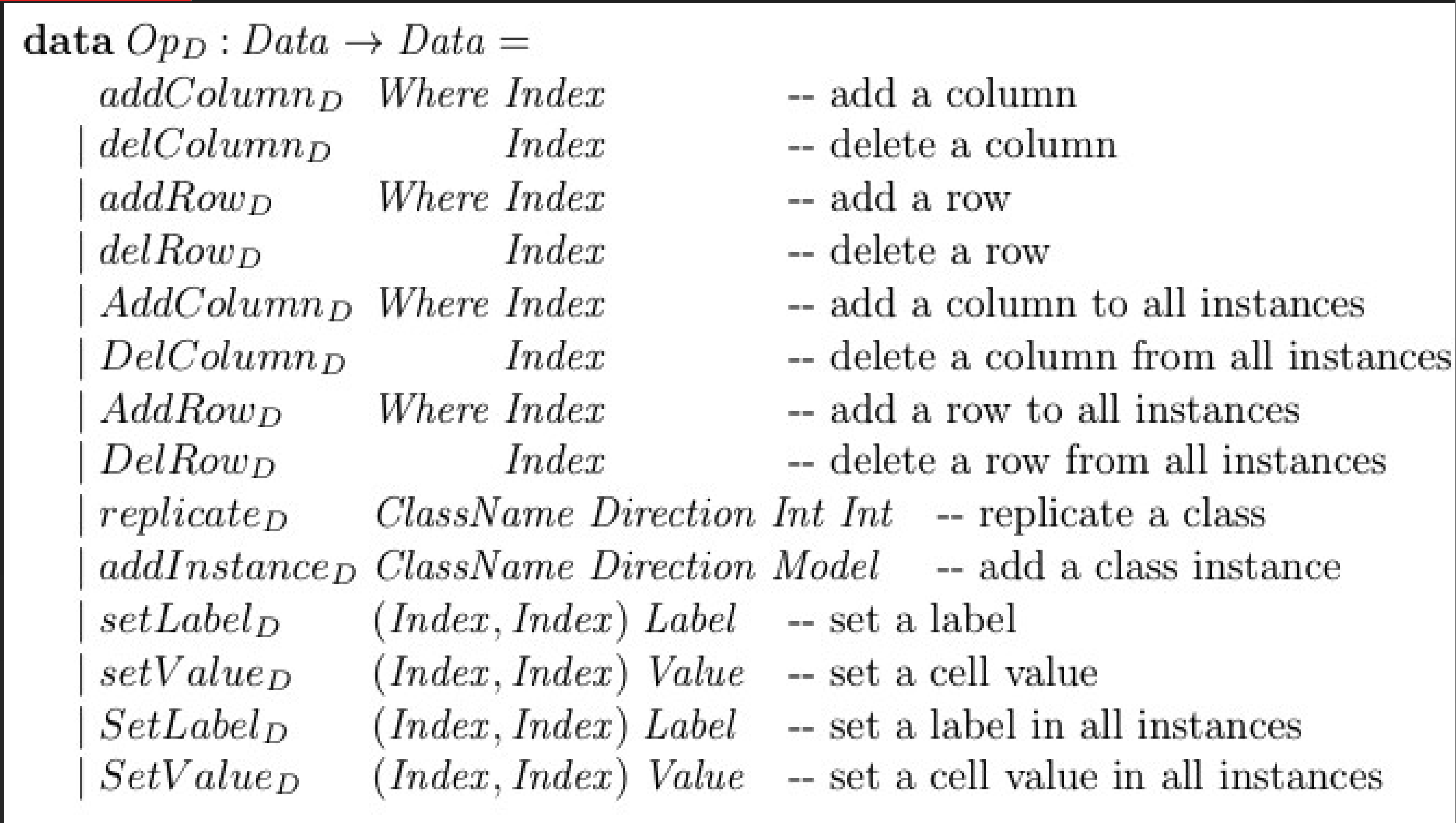
	A	B	C	D	E	F	G	H	I	J
1	Flights	PlanesKey ᵖ				PlanesKey ᵖ				
2		N2342 ▾				N341 ▾				
3	PilotsKey	Depart	Destination	Date	Hours	Depart	Destination	Date	Hours	
4	pl1 ▾	OPO	NAT	12/12/2010 – 14:00	07:00	LIS	AMS	16/12/2010 – 10:00	02:45	
5	pl1 ▾	OPO	NAT	01/01/2011 – 16:00	07:00			01/01/2010 – 0:00	00:00	...
6	pl4 ▾			01/01/2010 – 0:00	01:00			01/01/2010 – 0:00	01:00	
7					⋮					
8					07:00				02:45	
9	Pilots									
10	ID ᵖ	Name	Flight hours							
11	pl1	John	3400							
12	pl2	Mike	330							
13	pl3	Anne	433							
14		⋮								
15										
16	Planes									
17	N-Number ᵖ	N2342	N341	N1343	...					
18	Model	B 747	B 777	A 380						
19	Name	Magalhães	Cabral	Nunes						

Bidirectional Transformation System



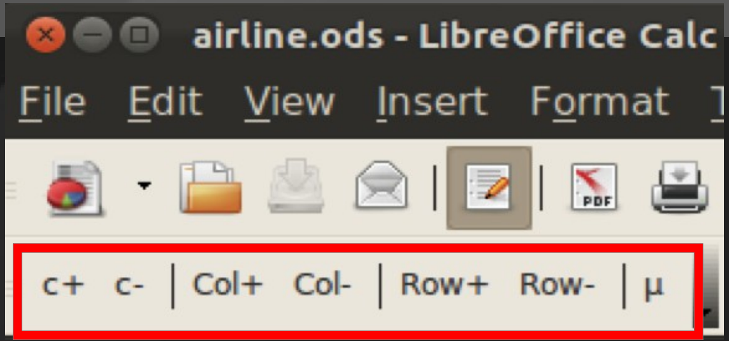
(Data) Operations on Instances

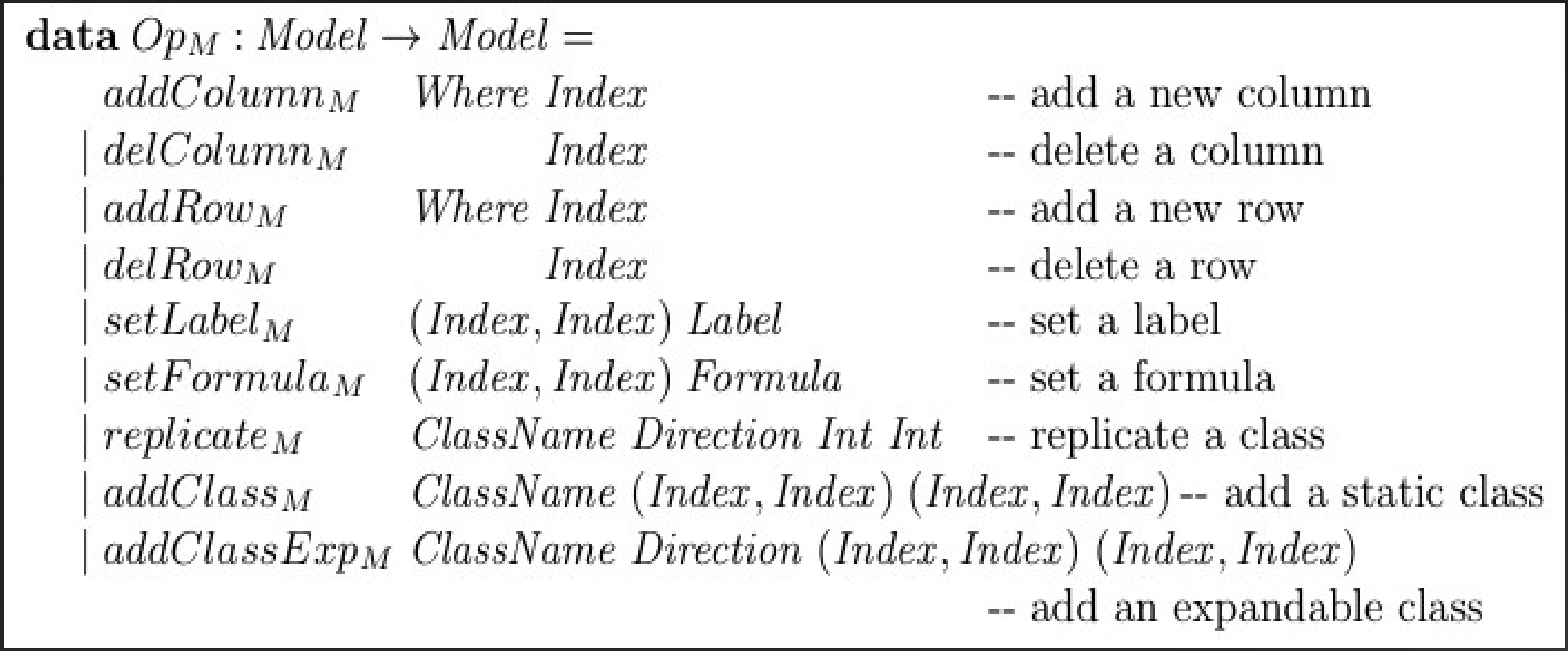




$$
\begin{array}{llll}
\mathbf{data}\ Op_D : Data \to Data = & & & \\
 & addColumn_D & Where\ Index & \text{-- add a column} \\
| & delColumn_D & Index & \text{-- delete a column} \\
| & addRow_D & Where\ Index & \text{-- add a row} \\
| & delRow_D & Index & \text{-- delete a row} \\
| & AddColumn_D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

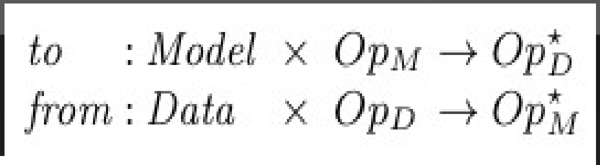
(Model) Operations on ClassSheets



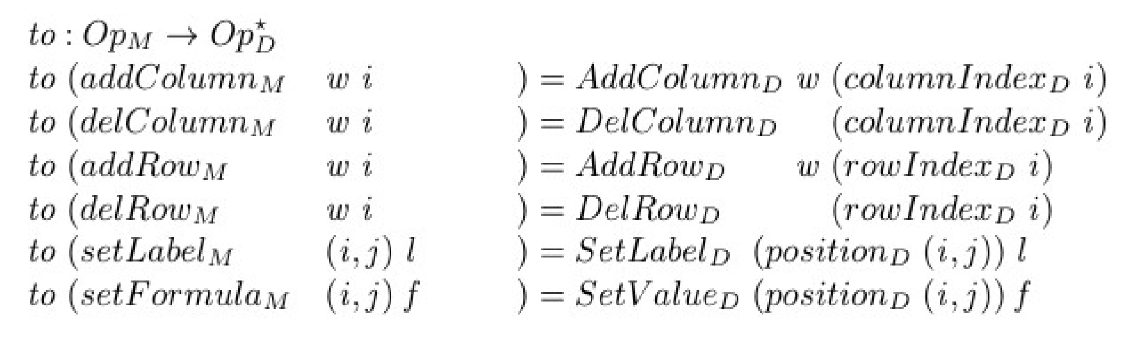


```
data Op_M : Model → Model =
      addColumn_M    Where Index                          -- add a new column
    | delColumn_M          Index                          -- delete a column
    | addRow_M       Where Index                          -- add a new row
    | delRow_M             Index                          -- delete a row
    | setLabel_M     (Index, Index) Label                 -- set a label
    | setFormula_M   (Index, Index) Formula               -- set a formula
    | replicate_M    ClassName Direction Int Int          -- replicate a class
    | addClass_M     ClassName (Index, Index) (Index, Index) -- add a static class
    | addClassExp_M  ClassName Direction (Index, Index) (Index, Index)
                                                          -- add an expandable class
```

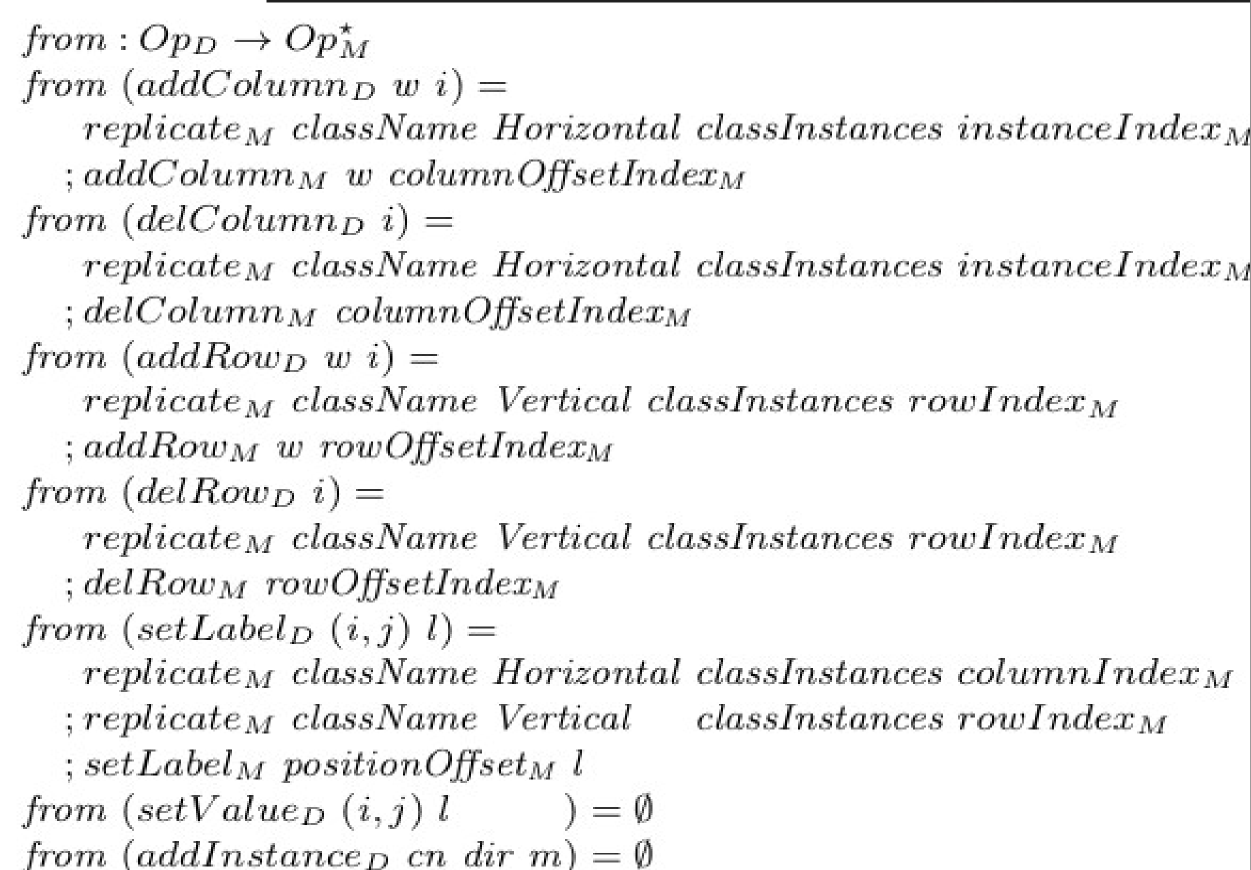
Bidirectional Transformation Functions



$$\begin{array}{llll} \mathit{to} & : \mathit{Model} & \times\ \mathit{Op}_M & \to \mathit{Op}_D^\star \\ \mathit{from} & : \mathit{Data} & \times\ \mathit{Op}_D & \to \mathit{Op}_M^\star \end{array}$$



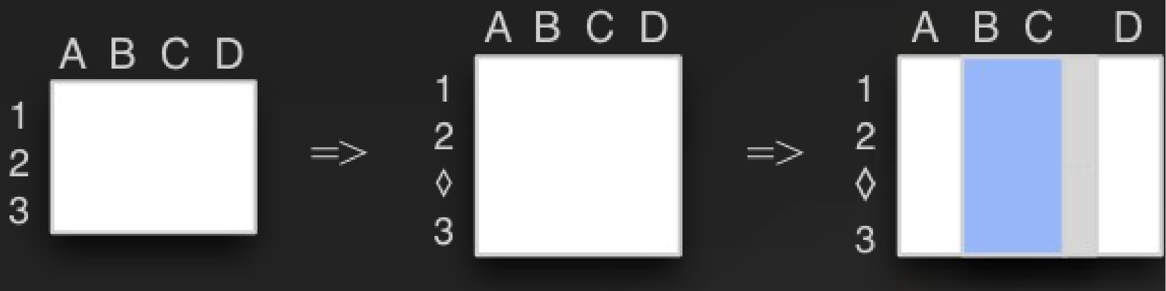
$$\begin{array}{llll} \multicolumn{4}{l}{\mathit{to} : \mathit{Op}_M \to \mathit{Op}_D^\star} \\ \mathit{to}\ (\mathit{addColumn}_M & w\ i &) = \mathit{AddColumn}_D & w\ (\mathit{columnIndex}_D\ i) \\ \mathit{to}\ (\mathit{delColumn}_M & w\ i &) = \mathit{DelColumn}_D & (\mathit{columnIndex}_D\ i) \\ \mathit{to}\ (\mathit{addRow}_M & w\ i &) = \mathit{AddRow}_D & w\ (\mathit{rowIndex}_D\ i) \\ \mathit{to}\ (\mathit{delRow}_M & w\ i &) = \mathit{DelRow}_D & (\mathit{rowIndex}_D\ i) \\ \mathit{to}\ (\mathit{setLabel}_M & (i,j)\ l &) = \mathit{SetLabel}_D & (\mathit{position}_D\ (i,j))\ l \\ \mathit{to}\ (\mathit{setFormula}_M & (i,j)\ f &) = \mathit{SetValue}_D & (\mathit{position}_D\ (i,j))\ f \end{array}$$



$$\begin{array}{l} \mathit{from} : \mathit{Op}_D \to \mathit{Op}_M^\star \\ \mathit{from}\ (\mathit{addColumn}_D\ w\ i) = \\ \quad \mathit{replicate}_M\ \mathit{className}\ \mathit{Horizontal}\ \mathit{classInstances}\ \mathit{instanceIndex}_M \\ \quad ; \mathit{addColumn}_M\ w\ \mathit{columnOffsetIndex}_M \\ \mathit{from}\ (\mathit{delColumn}_D\ i) = \\ \quad \mathit{replicate}_M\ \mathit{className}\ \mathit{Horizontal}\ \mathit{classInstances}\ \mathit{instanceIndex}_M \\ \quad ; \mathit{delColumn}_M\ \mathit{columnOffsetIndex}_M \\ \mathit{from}\ (\mathit{addRow}_D\ w\ i) = \\ \quad \mathit{replicate}_M\ \mathit{className}\ \mathit{Vertical}\ \mathit{classInstances}\ \mathit{rowIndex}_M \\ \quad ; \mathit{addRow}_M\ w\ \mathit{rowOffsetIndex}_M \\ \mathit{from}\ (\mathit{delRow}_D\ i) = \\ \quad \mathit{replicate}_M\ \mathit{className}\ \mathit{Vertical}\ \mathit{classInstances}\ \mathit{rowIndex}_M \\ \quad ; \mathit{delRow}_M\ \mathit{rowOffsetIndex}_M \\ \mathit{from}\ (\mathit{setLabel}_D\ (i,j)\ l) = \\ \quad \mathit{replicate}_M\ \mathit{className}\ \mathit{Horizontal}\ \mathit{classInstances}\ \mathit{columnIndex}_M \\ \quad ; \mathit{replicate}_M\ \mathit{className}\ \mathit{Vertical}\ \mathit{classInstances}\ \mathit{rowIndex}_M \\ \quad ; \mathit{setLabel}_M\ \mathit{positionOffset}_M\ l \\ \mathit{from}\ (\mathit{setValue}_D\ (i,j)\ l\ ) = \emptyset \\ \mathit{from}\ (\mathit{addInstance}_D\ \mathit{cn}\ \mathit{dir}\ m) = \emptyset \end{array}$$

Compositional
Example: *Add a Column and a Class*

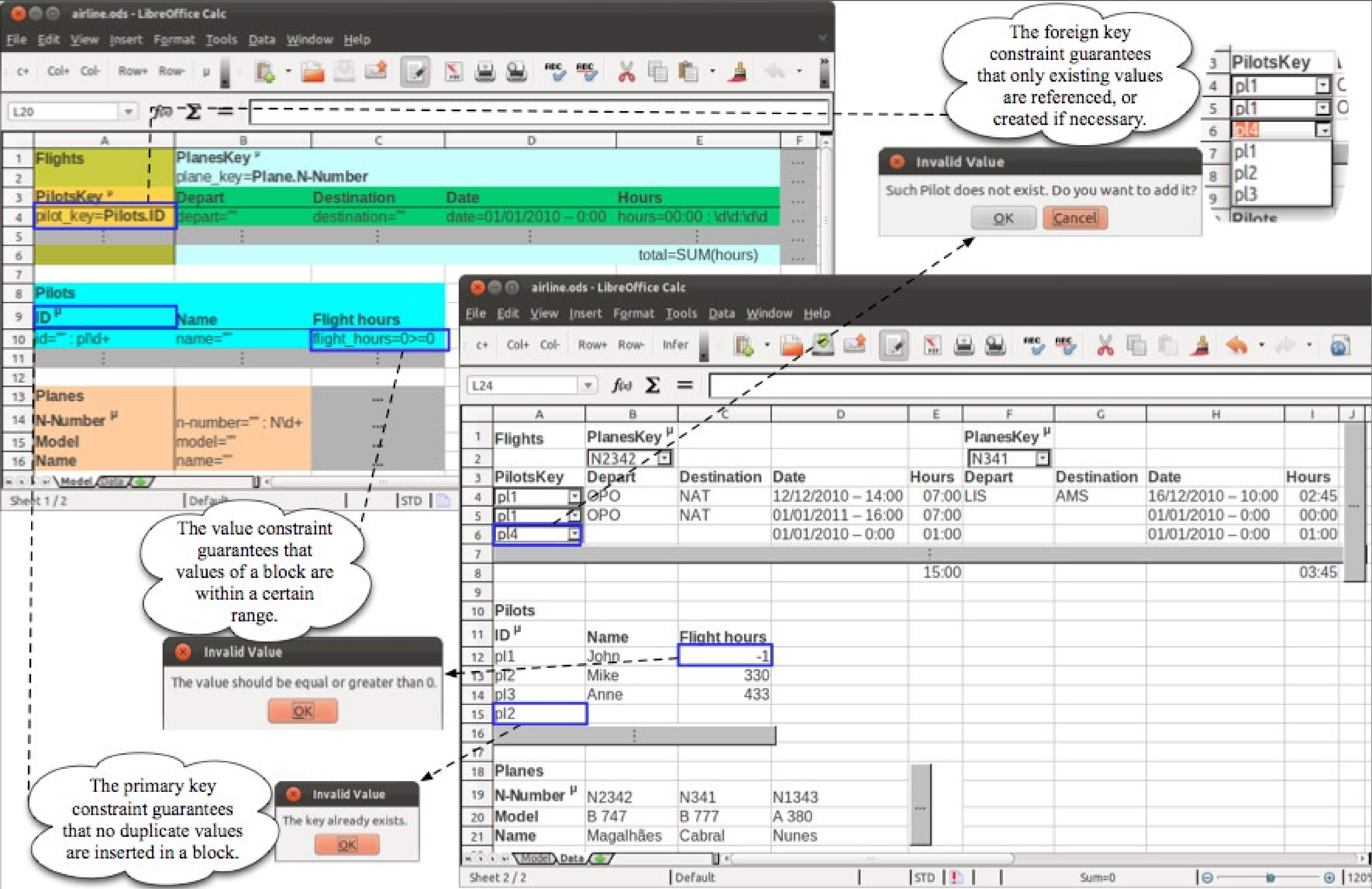
addRow$_M$ Before 3 ; addClassExp$_M$ "BlueClass" Horizontal (2,1) (3,4)



VII. MDSheet – Model-Driven Spreadsheets
[ICSE'12]

MDSheet Tool

http://youtu.be/6LNdTdCpV2U


airline.ods - LibreOffice Calc
File Edit View Insert Format Tools Data Window Help
C+ Col+ Col- Row+ Row- µ
L20
Sheet 1 / 2
Default
STD
Model Data
airline.ods - LibreOffice Calc
File Edit View Insert Format Tools Data Window Help
C+ Col+ Col- Row+ Row- Infer
L24
Sheet 2 / 2
Default
STD
Sum=0
120%

	A	B	C	D	E
1	Flights	PlanesKey µ			
2		plane_key=Plane.N-Number			
3	PilotsKey µ	Depart	Destination	Date	Hours
4	pilot_key=Pilots.ID	depart=""	destination=""	date=01/01/2010 – 0:00	hours=00:00 : \d\d:\d\d
5	⋮	⋮	⋮	⋮	⋮
6					total=SUM(hours)
7					
8	Pilots				
9	ID µ	Name	Flight hours		
10	id="" : pl\d+	name=""	flight_hours=0>=0		
11	⋮	⋮	⋮		
12					
13	Planes		...		
14	N-Number µ	n-number="" : N\d+	...		
15	Model	model=""	...		
16	Name	name=""	...		

	A	B	C	D	E	F	G	H	I
1	Flights	PlanesKey µ				PlanesKey µ			
2		N2342				N341			
3	PilotsKey	Depart	Destination	Date	Hours	Depart	Destination	Date	Hours
4	pl1	OPO	NAT	12/12/2010 – 14:00	07:00	LIS	AMS	16/12/2010 – 10:00	02:45
5	pl1	OPO	NAT	01/01/2011 – 16:00	07:00			01/01/2010 – 0:00	00:00
6	pl4			01/01/2010 – 0:00	01:00			01/01/2010 – 0:00	01:00
7					⋮				
8					15:00				03:45
9									
10	Pilots								
11	ID µ	Name	Flight hours						
12	pl1	John	-1						
13	pl2	Mike	330						
14	pl3	Anne	433						
15	pl2								
16		⋮							
17									
18	Planes								
19	N-Number µ	N2342	N341	N1343	...				
20	Model	B 747	B 777	A 380					
21	Name	Magalhães	Cabral	Nunes					

	PilotsKey
4	pl1
5	pl1
6	pl4
7	pl1
8	pl2
9	pl3

The foreign key constraint guarantees that only existing values are referenced, or created if necessary.

Invalid Value

Such Pilot does not exist. Do you want to add it?

OK Cancel

The value constraint guarantees that values of a block are within a certain range.

Invalid Value

The value should be equal or greater than 0.

OK

The primary key constraint guarantees that no duplicate values are inserted in a block.

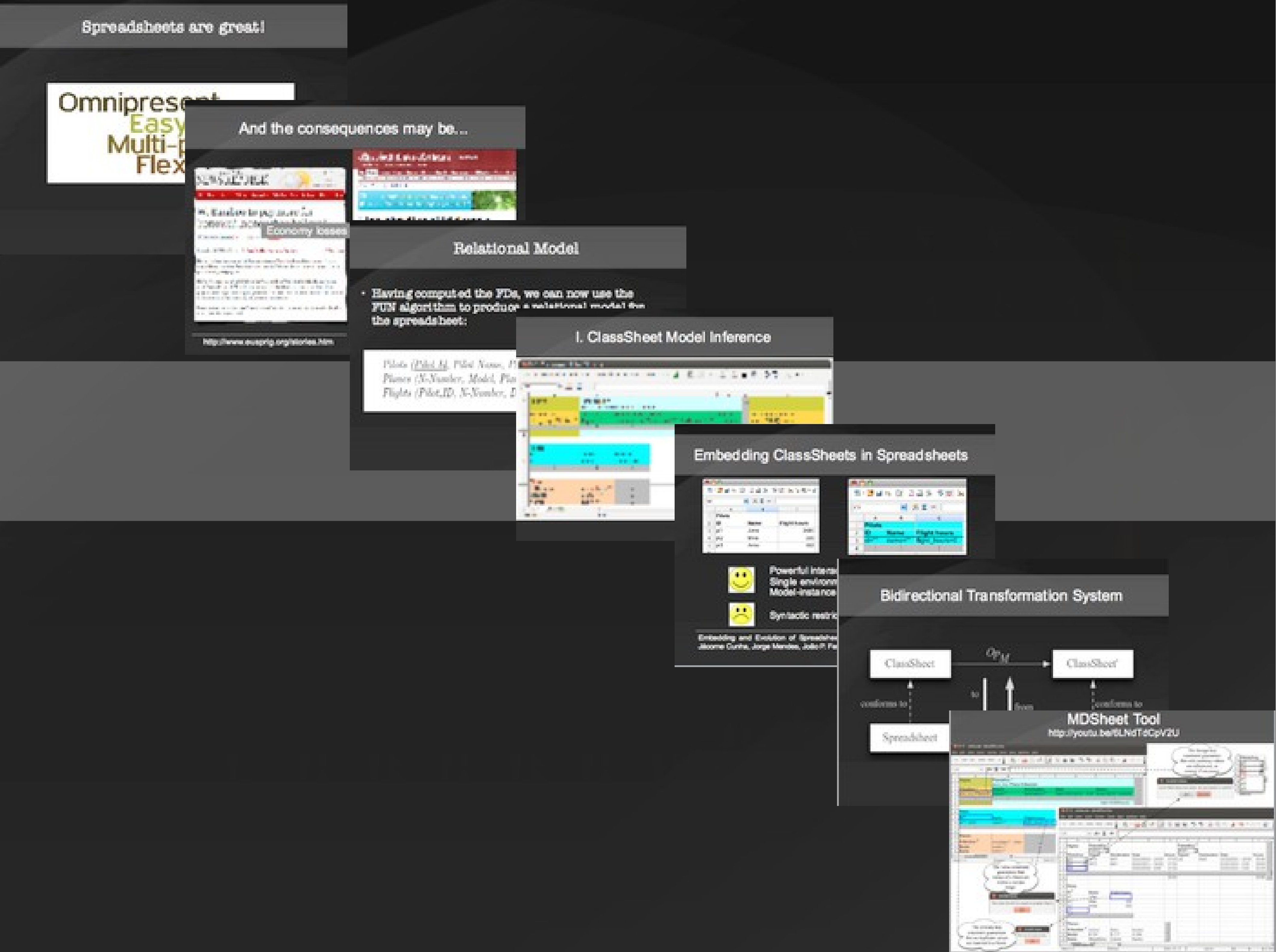
Invalid Value

The key already exists.

OK

- Available at http://ssaapp.di.uminho.pt

- Built out of 7886 LOC:
  - 3181 in Haskell, for the inference and evolution
  - 980 in Basic, for the embedding
  - 2884 in C++, for gluing all components
  - 340 in Perl, for compilation and setup
  - 722, for makefiles



Spreadsheets are great!

Omnipresent
Easy
Multi-p
Flex

And the consequences may be...

Economy losses

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

Relational Model

- Having computed the FDs, we can now use the FUN algorithm to produce a relational model for the spreadsheet:

Pilots (Pilot Id, Pilot Name, P
Planes (N-Number, Model, Pla
Flights (Pilot ID, N-Number, D

I. ClassSheet Model Inference

Embedding ClassSheets in Spreadsheets

Powerful intera
Single environm
Model-instance

Syntactic restric

Embedding and Evolution of Spreadshe
Jácome Cunha, Jorge Mendes, João P. Fe

Bidirectional Transformation System

MDSheet Tool

http://youtu.be/5LNdT-dCpV2U

Acknowledgments

This work has been done in collaboration with many people:

Martin Erwig, João Paulo Fernandes, Jorge Mendes, Hugo Pacheco, Rui Pereira, João Saraiva, Joost Visser

Thanks!

Questions?

More?

- More at http://ssaapp.di.uminho.pt
- Querying model-driven spreadsheet
- Visually querying model-driven spreadsheets
- Detections of bad smells
- Edit assistance
- Empirical validations
- Variational spreadsheets (@ OSU)
- ...

Does It Work?

Empirical Study Settings

- 17 student from a MSc course
- 2 different spreadsheets
  - Microsoft budget
  - Local company responsible for water supply of Braga, Portugal - agere

Study Setting

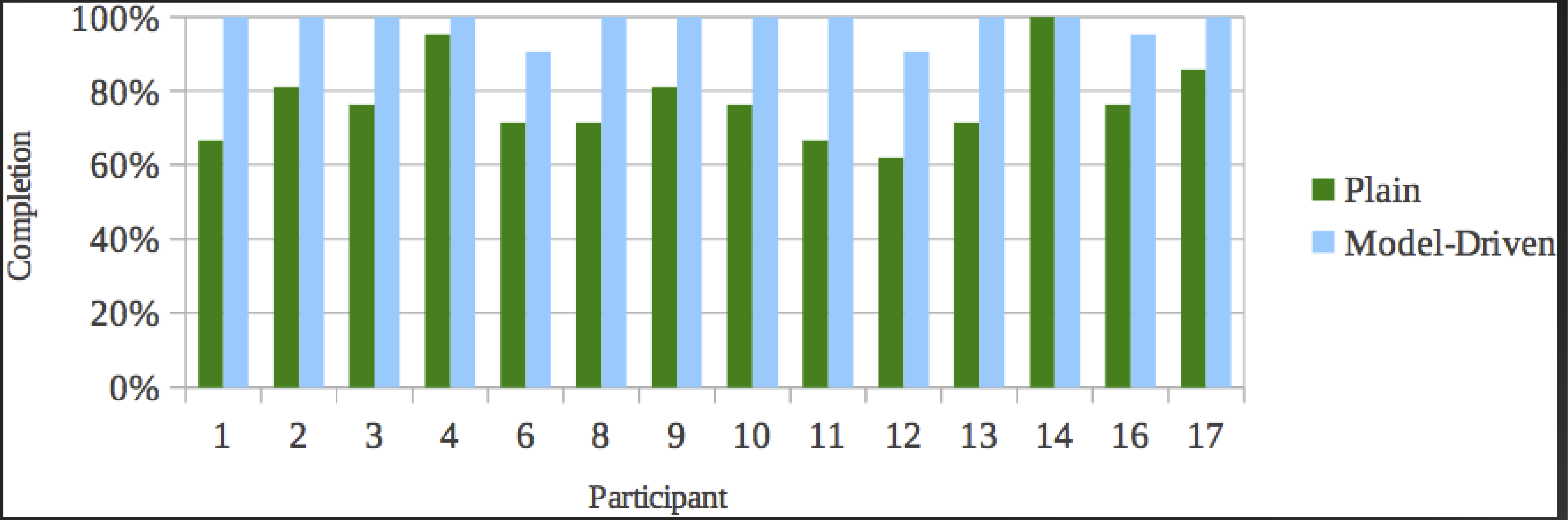
- Hypotheses:

(1) In order to perform a given set of tasks, users spend less time when using model-driven spreadsheets instead of plain ones.

(2) Spreadsheets developed in the model-driven environment hold less errors than plain ones.

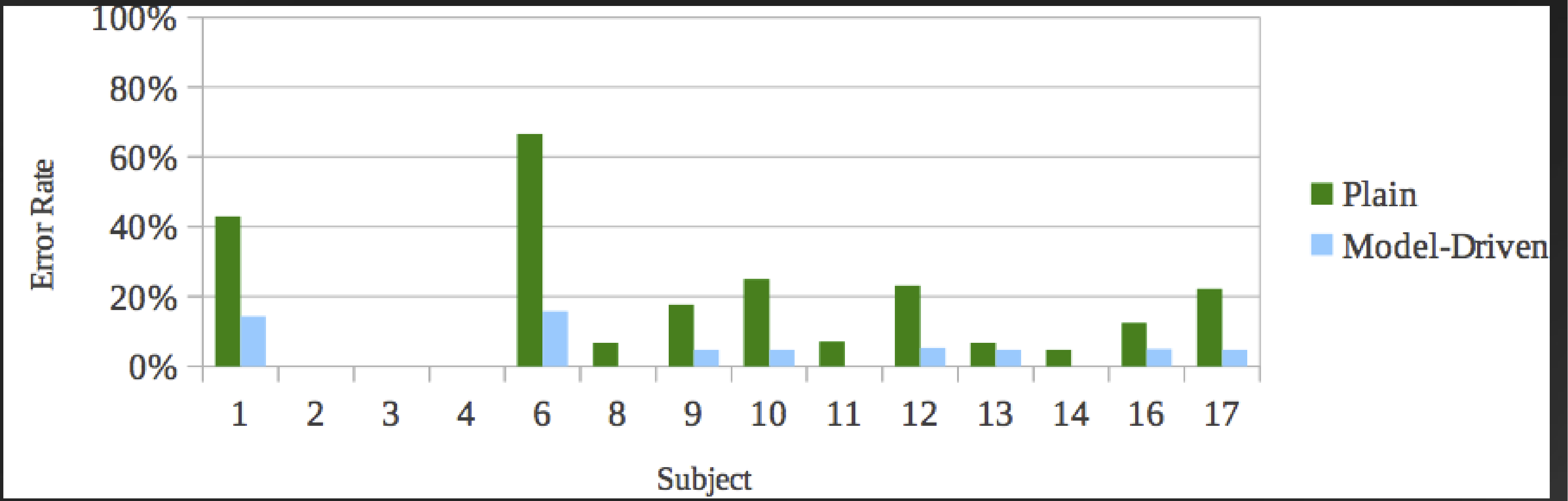
Main Results

Number of tasks performed on the MS spreadsheet



Main Results

Error rate in the budget spreadsheet


Error Rate
Subject
Plain
Model-Driven