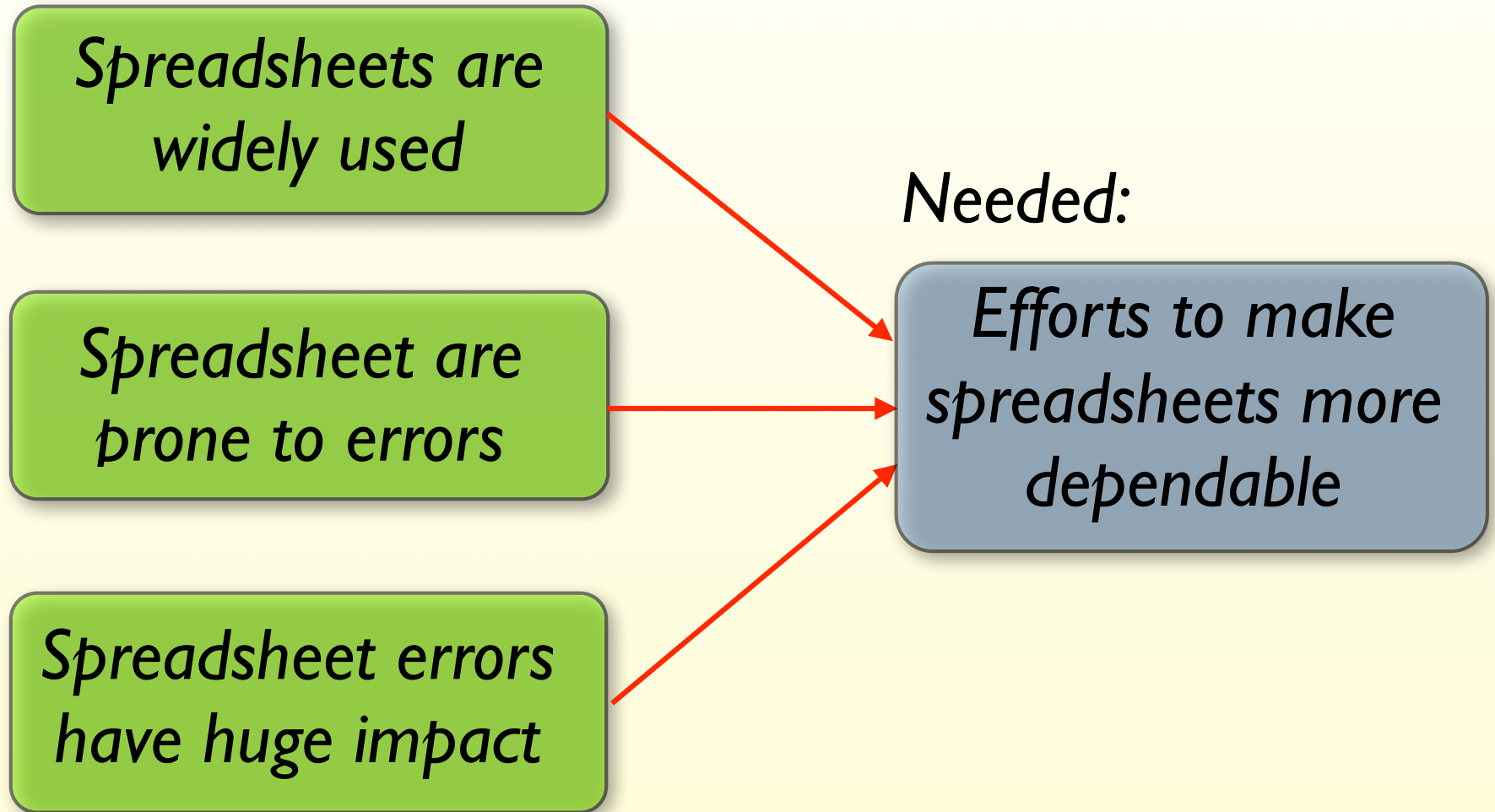


Software Engineering for Spreadsheets: Challenges and Opportunities

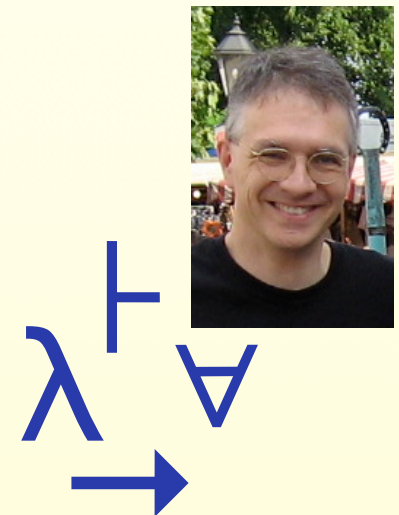
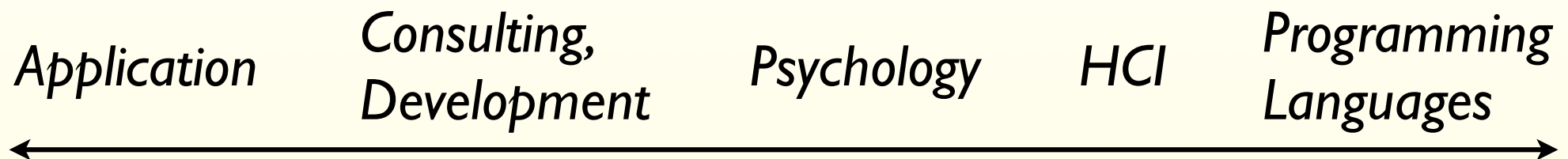
Martin Erwig
Oregon State University

Supported by NSF (ITR-0325273, CCF-0741584)

Motivation



Spectrum of Viewpoints



Let's Use Tools!

Processes is important ...

... but humans are fallible



Tools are indispensable in all areas of our daily lives



...

	A	B	C	D	E
1		Fruit			
2	Month	Apple	Orange	Total	
3	May		17	9	26
4	June		8	13	21
5	Total		25	22	47

Contents

The Challenge

The Essence of Spreadsheet Computing

Opportunities for Tools

Example 1: Type Checking

Example 2: Debugging

Challenge

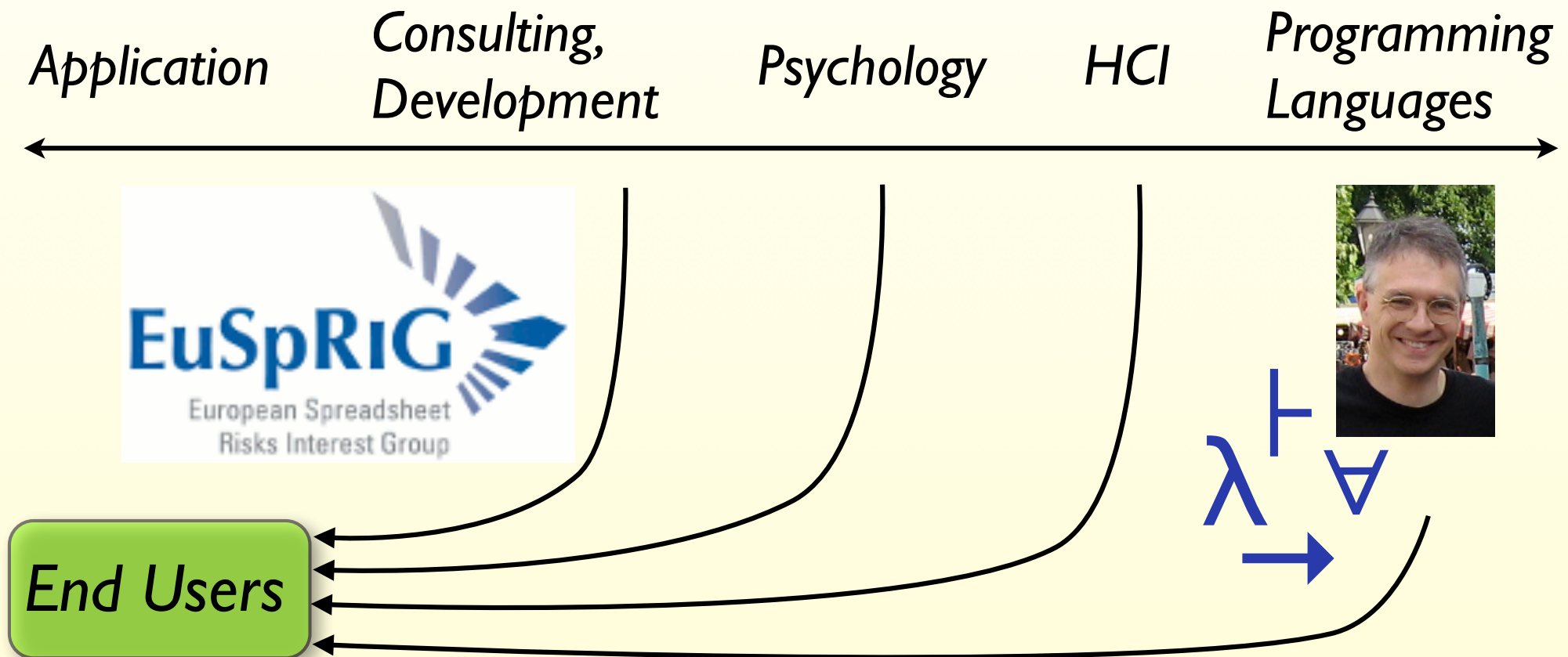
*Editing Spreadsheets $\approx$
Software Engineering*

*Spreadsheet Users:
End Users*

Objective:
get the job done

*How to get end users to
employ SE techniques?*

Everybody is an End User



Contents

The Challenge

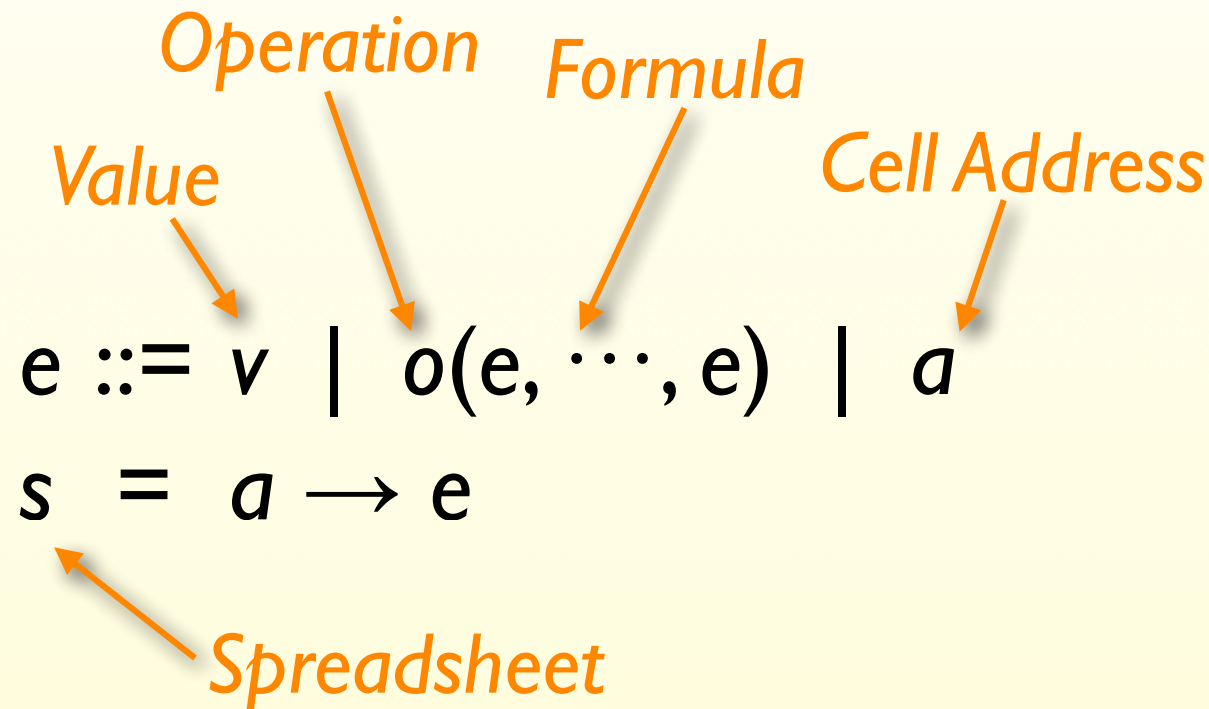
The Essence of Spreadsheet Computing

Opportunities for Tools

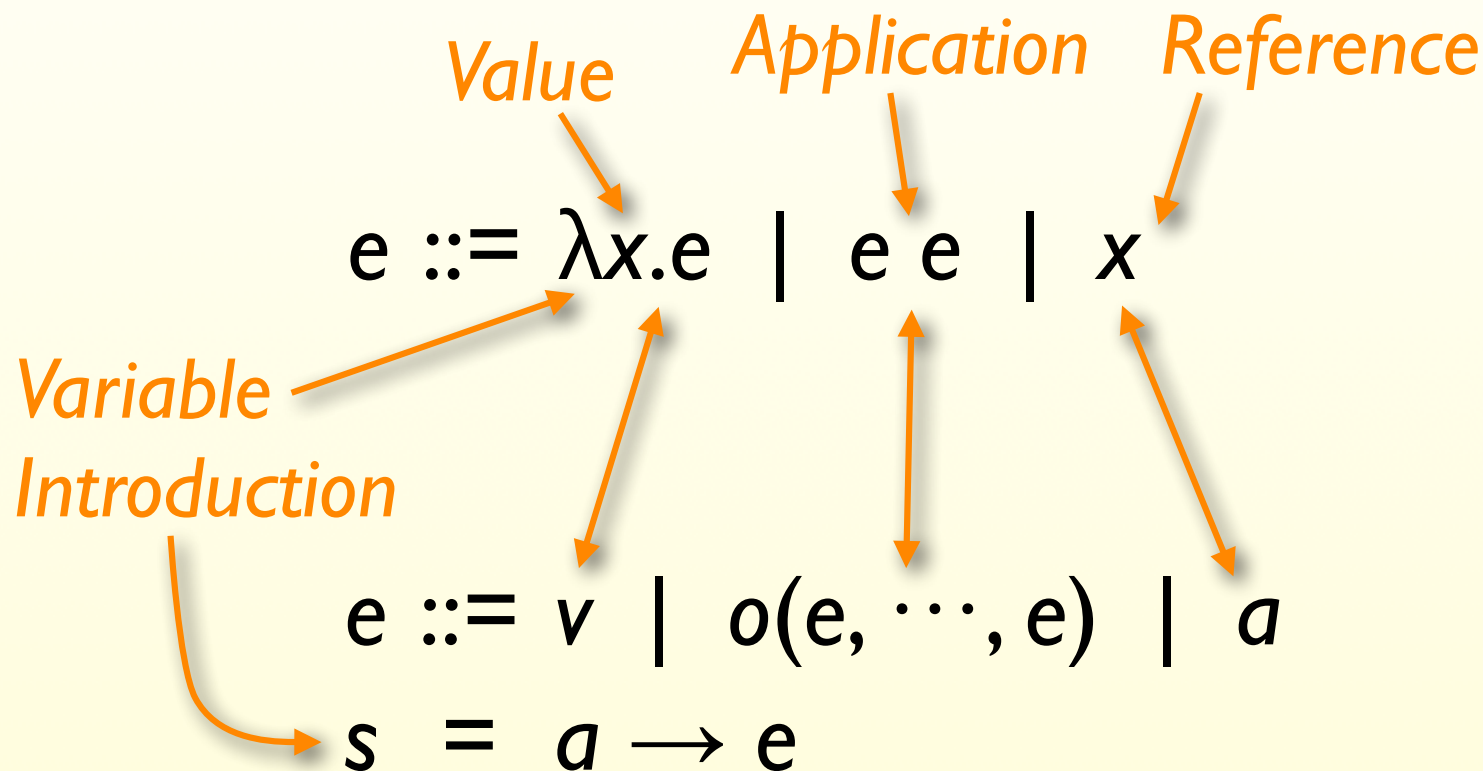
Example 1: Type Checking

Example 2: Debugging

The Essence of Spreadsheet Computing



The Essence of Spreadsheet Computing



No Local Scope

No Higher-Order Operations $\Rightarrow$ No Recursion

Contents

The Challenge

The Essence of Spreadsheet Computing

Opportunities for Tools

Example 1: Type Checking

Example 2: Debugging

Opportunities

No loops/recursion

*Limited Computation
Model*



*Simplified Reasoning
for Tools*

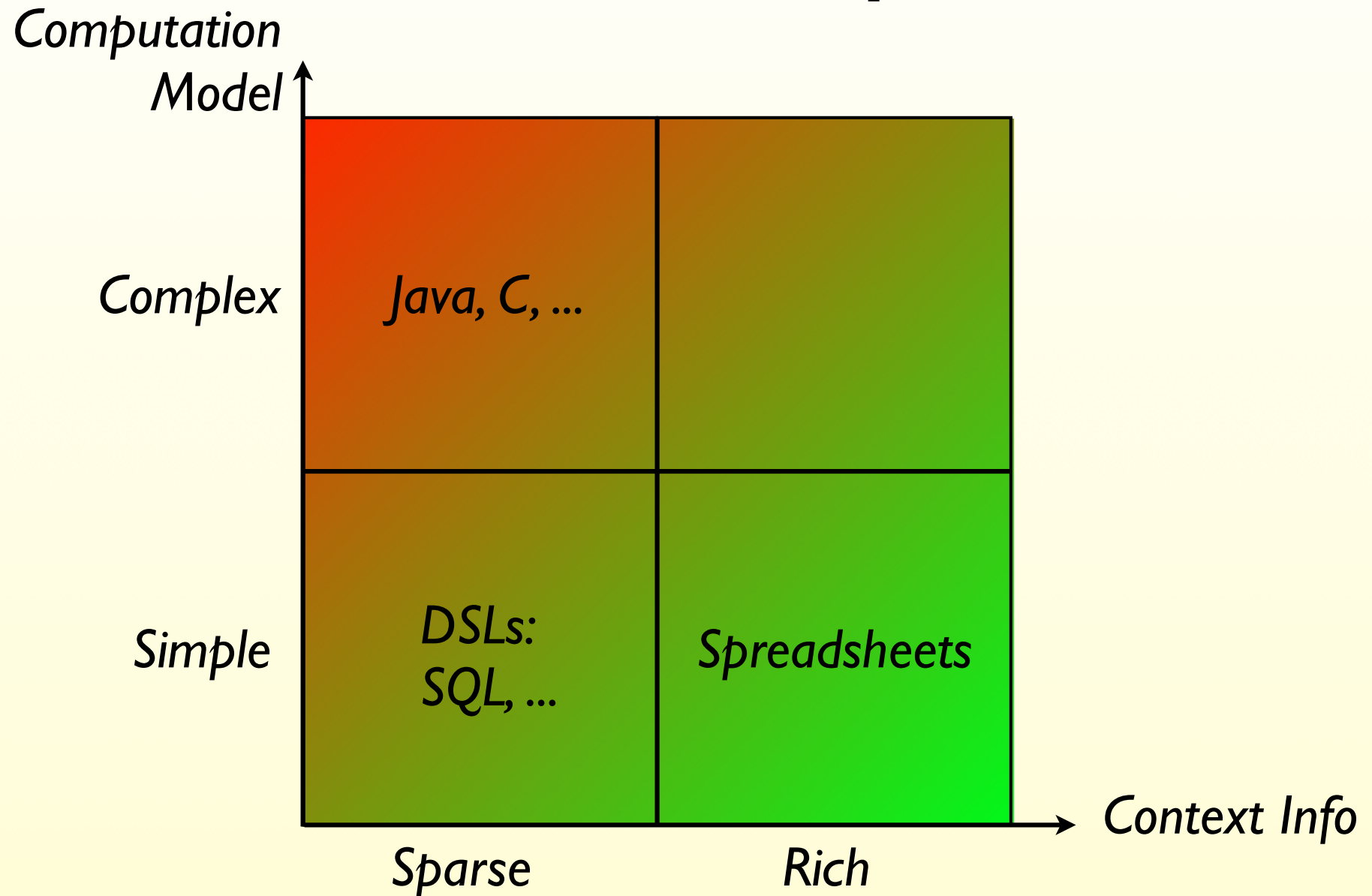
*Spatial Embedding
of Formulas*



*Availability of Rich
Contextual Information*

[IEEE Software, Sept. 2009]

A Sweet Spot



Contents

The Challenge

The Essence of Spreadsheet Computing

Opportunities for Tools

Example 1: Type Checking

Example 2: Debugging

End-User Type System?

Problem

Abstract concept of types is very difficult to convey to end users

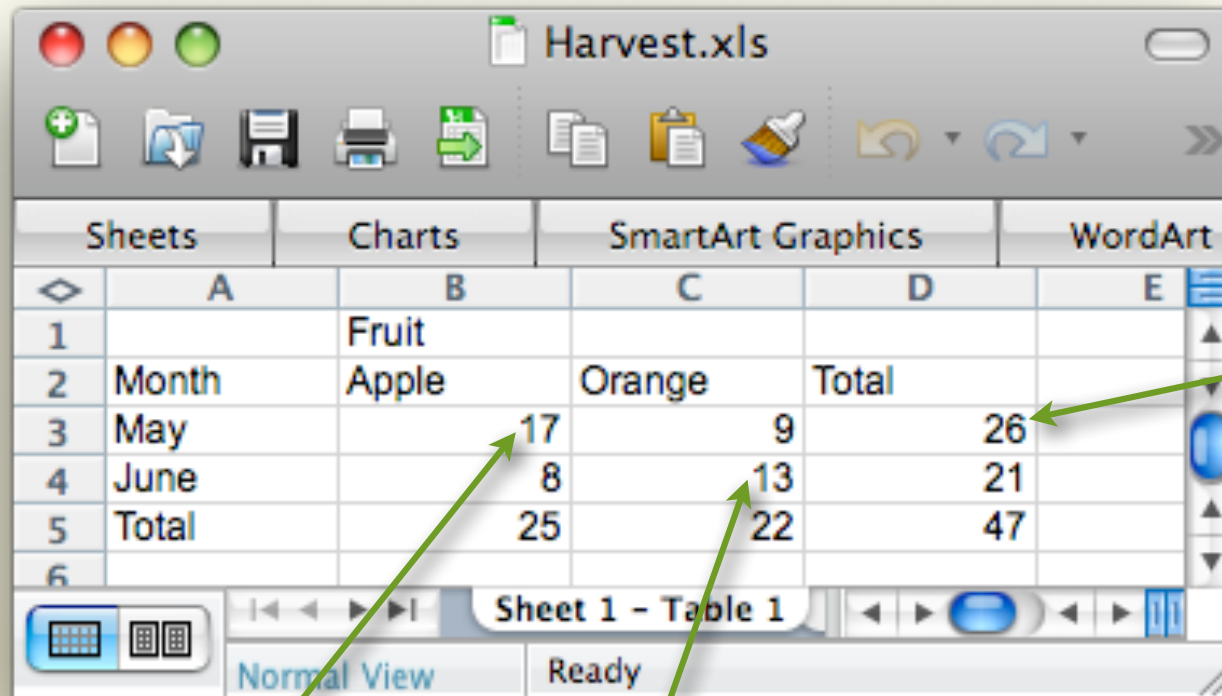
Type System

End user

Idea

Use vocabulary from the spreadsheet as type information
[PADL'02, HCC'02]

Type Checking



The screenshot shows a spreadsheet application window titled "Harvest.xls". The window has a menu bar with options: Sheets, Charts, SmartArt Graphics, and WordArt. Below the menu bar is a toolbar with various icons. The spreadsheet itself has columns labeled A through E and rows numbered 1 through 6. The data is organized as follows:

	A	B	C	D	E
1		Fruit			
2	Month	Apple	Orange	Total	
3	May	17	9	26	
4	June	8	13	21	
5	Total	25	22	47	
6					

At the bottom of the window, there is a status bar showing "Normal View" and "Ready". The sheet name "Sheet 1 - Table 1" is also visible.

May Fruit

May Apple

June Orange

Labeling Rules

$e : L$ and $L : K \Rightarrow e : K \triangleright L$

Fruit $\triangleright$ Apple

$e : L$ and $e : K \Rightarrow e : L \& K$

Month $\triangleright$ May & Fruit $\triangleright$ Apple

$e : L$ and $e' : K \Rightarrow e + e' : L \mid K$

May Apple $\mid$ May Orange

May (Apple $\mid$ Orange)

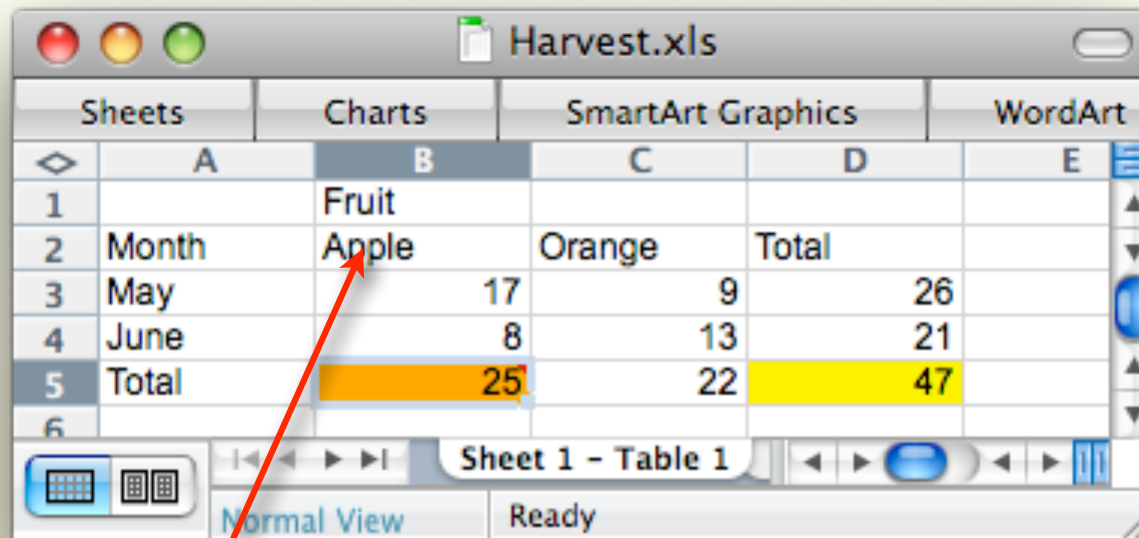
May Fruit

Factoring

Generalization

	A	B	C	D	E
1		Fruit			
2	Month	Apple	Orange	Total	
3	May	17	9	26	
4	June	8	13	21	
5	Total	25	22	47	
6					

A Summation Error



	A	B	C	D	E
1		Fruit			
2	Month	Apple	Orange	Total	
3	May	17	9	26	
4	June	8	13	21	
5	Total	25	22	47	
6					

B5 = SUM(B2:B4)

B2 : Fruit

B3 : Fruit▷ Apple

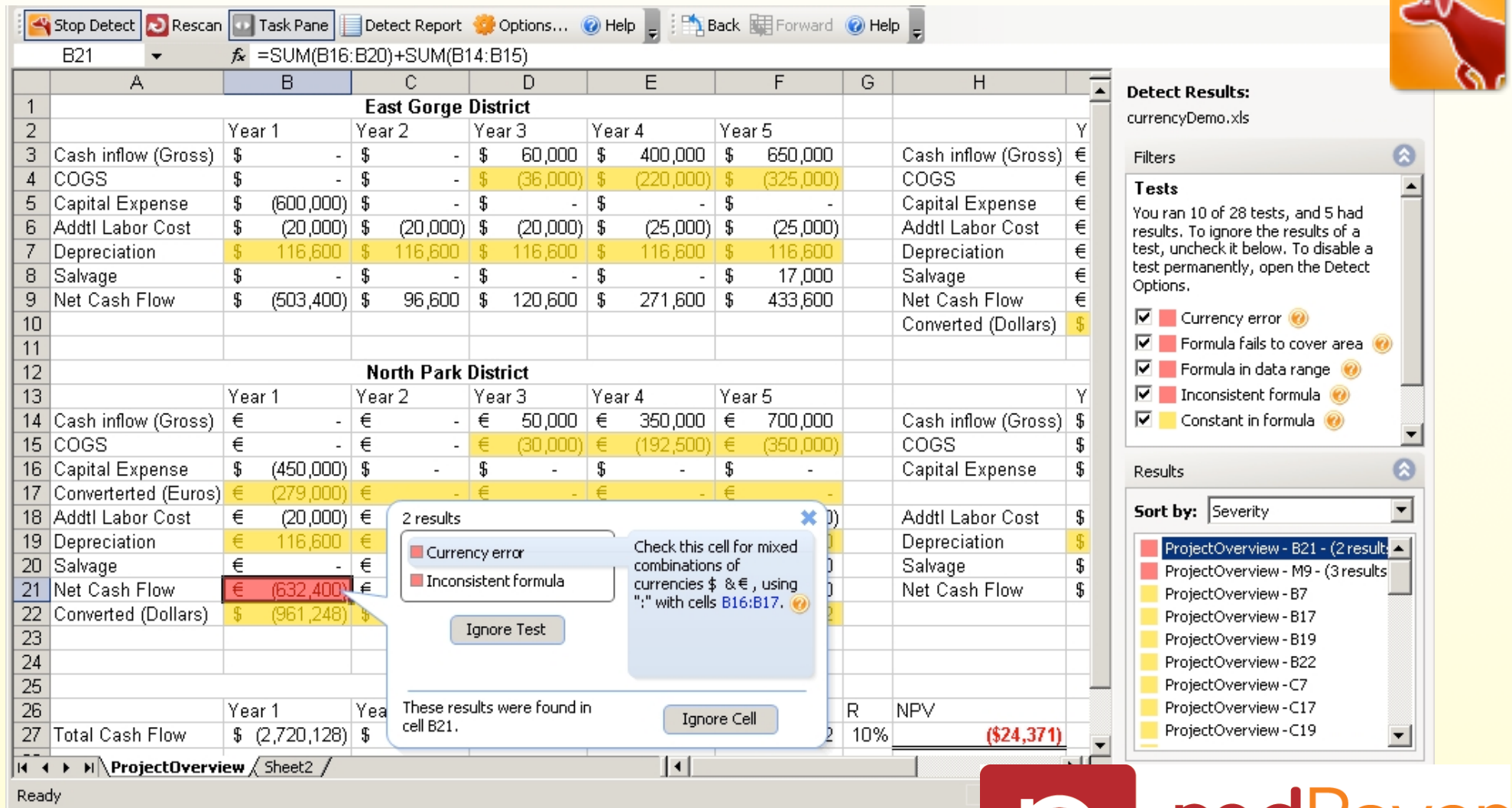
B4 : Fruit▷ Apple

B2's label cannot be factored $\Rightarrow$ prevents generalization step

Reflections

- We can infer label relationships automatically [VL/HCC'04, JVLC'07]
- Users *do* use labels to reason about formulas in spreadsheets to debug effectively [VL/HCC'07]
- Combining syntactic label checking with semantic dimension analysis finds even more faults [VL/HCC'08a, JVLC'09, VL/HCC'09, VL/HCC'10]
- Is it actually practical?

Part of Real-World Tool



Stop Detect **Rescan** **Task Pane** **Detect Report** **Options...** **Help** **Back** **Forward** **Help**

B21 $=SUM(B16:B20)+SUM(B14:B15)$

East Gorge District							
	Year 1	Year 2	Year 3	Year 4	Year 5		Y
Cash inflow (Gross)	\$ -	\$ -	\$ 60,000	\$ 400,000	\$ 650,000	Cash inflow (Gross)	€
COGS	\$ -	\$ -	\$ (36,000)	\$ (220,000)	\$ (325,000)	COGS	€
Capital Expense	\$ (600,000)	\$ -	\$ -	\$ -	\$ -	Capital Expense	€
Addtl Labor Cost	\$ (20,000)	\$ (20,000)	\$ (20,000)	\$ (25,000)	\$ (25,000)	Addtl Labor Cost	€
Depreciation	\$ 116,600	\$ 116,600	\$ 116,600	\$ 116,600	\$ 116,600	Depreciation	€
Salvage	\$ -	\$ -	\$ -	\$ -	\$ 17,000	Salvage	€
Net Cash Flow	\$ (503,400)	\$ 96,600	\$ 120,600	\$ 271,600	\$ 433,600	Net Cash Flow	€
						Converted (Dollars)	\$

North Park District							
	Year 1	Year 2	Year 3	Year 4	Year 5		Y
Cash inflow (Gross)	€ -	€ -	€ 50,000	€ 350,000	€ 700,000	Cash inflow (Gross)	\$
COGS	€ -	€ -	€ (30,000)	€ (192,500)	€ (350,000)	COGS	\$
Capital Expense	\$ (450,000)	\$ -	\$ -	\$ -	\$ -	Capital Expense	\$
Converted (Euros)	€ (279,000)	€ -	€ -	€ -	€ -		
Addtl Labor Cost	€ (20,000)	€ -	€ -	€ -	€ -	Addtl Labor Cost	\$
Depreciation	€ 116,600	€ -	€ -	€ -	€ -	Depreciation	\$
Salvage	€ -	€ -	€ -	€ -	€ -	Salvage	\$
Net Cash Flow	€ (632,400)	€ -	€ -	€ -	€ -	Net Cash Flow	\$
Converted (Dollars)	\$ (961,248)	\$ -	\$ -	\$ -	\$ -		

	Year 1	Year 2	Year 3	Year 4	Year 5	NPV
Total Cash Flow	\$ (2,720,128)	\$ -	\$ -	\$ -	\$ -	10% (\$24,371)

Detect Results: currencyDemo.xls

Filters

Tests

You ran 10 of 28 tests, and 5 had results. To ignore the results of a test, uncheck it below. To disable a test permanently, open the Detect Options.

- ☒ Currency error
- ☒ Formula fails to cover area
- ☒ Formula in data range
- ☒ Inconsistent formula
- ☒ Constant in formula

Results

Sort by: Severity

- ProjectOverview - B21 - (2 results)
- ProjectOverview - M9 - (3 results)
- ProjectOverview - B7
- ProjectOverview - B17
- ProjectOverview - B19
- ProjectOverview - B22
- ProjectOverview - C7
- ProjectOverview - C17
- ProjectOverview - C19

2 results

- Currency error
- Inconsistent formula

Check this cell for mixed combinations of currencies \$ & €, using ":" with cells B16:B17.

Ignore Test

These results were found in cell B21.

Ignore Cell

ProjectOverview Sheet2

Ready



redRover
SOFTWARE

Contents

The Challenge

The Essence of Spreadsheet Computing

Opportunities for Tools

Example 1: Type Checking

Example 2: Debugging

Goal-Directed Debugging

Key Ideas

- Reason directly from failure to fault
- Propagate change expectations to derive change suggestions
- Rank suggestions employing heuristics

[VL/HCC'05]

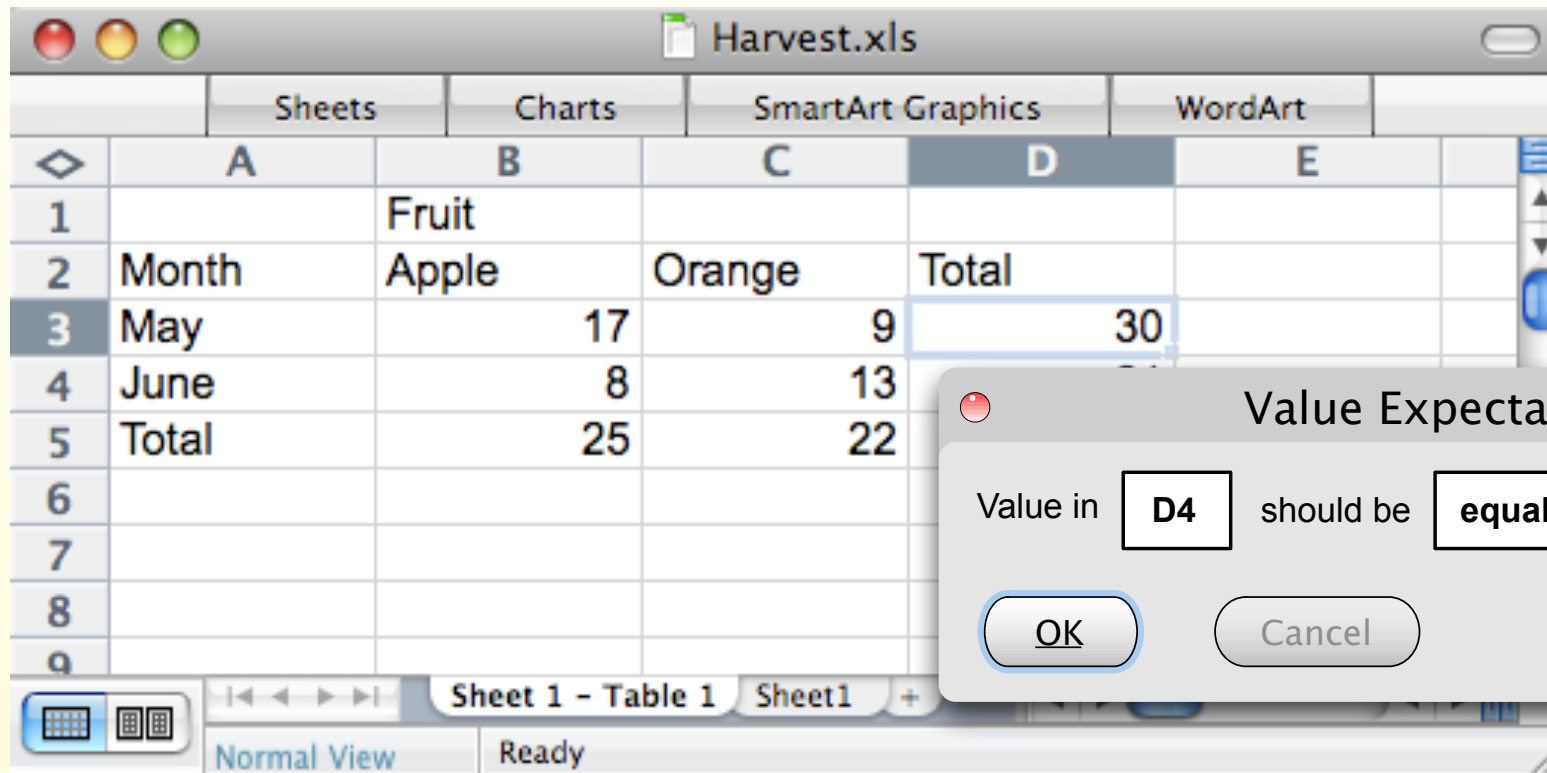
	A	B
1	6	$A1+2$
2	5	

Computed **8**
Expected **7**

*Change
Suggestions*

$A1+2 \leadsto A1+1$
 $A1+2 \leadsto A2+2$
 $A1+2 \leadsto 7$

Example



The screenshot shows a spreadsheet application window titled 'Harvest.xls'. The spreadsheet has columns A through E and rows 1 through 9. The data is as follows:

	A	B	C	D	E
1		Fruit			
2	Month	Apple	Orange	Total	
3	May	17	9	30	
4	June	8	13		
5	Total	25	22		
6					
7					
8					
9					

A 'Value Expectation' dialog box is open in the foreground. It contains the following text and controls:

- Title: Value Expectation
- Value in: D4
- should be: equal to (with a dropdown arrow)
- Value: 26
- Buttons: OK, Cancel

The dialog box is positioned over the spreadsheet, specifically over the cell D4 (June, Total) which contains the value 13.

Example

$$D4 = B3 + C4$$

The screenshot shows a spreadsheet window titled "Harvest.xls". The spreadsheet has columns A through E and rows 1 through 9. The data is as follows:

	A	B	C	D	E
1		Fruit			
2	Month	Apple	Orange	Total	
3	May	17	9	30	
4	June	8	13		
5	Total	25	22		
6					
7					
8					
9					

An error message box is displayed over cell D3, titled "Cell D3". The message reads: "In formula: =B3+C4 change B3 to C4 change C4 to C3". The box has "Ignore" and "More ..." buttons.

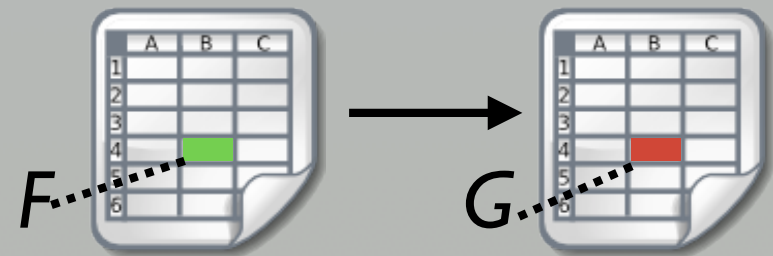
Evaluation of GoalDebug

- Corrects 97% of faults
- Ranking of correct change suggestion

First	59%
Top 2	71%
Top 5	80%

[ICSE'07, VL/HCC'08b]

1. Introduce Mutation



2. Run GoalDebug

3. Find rank of correct change suggestions

...

$G \leadsto \dots$

$k: G \leadsto F$

...

Conclusions

*Simplified Computation Model & Spatial Embedding
Facilitate
Effective, Easy-to-Use Software Engineering Tools*

More Examples:

- *Generation of Correct Spreadsheet:*
Gencel [[VL/HCC'05](#), [ICSE'05](#), [ICSE'06](#), [JFP'06](#)]
ClassSheets [[ASE'05](#), [JOT'07](#), [VL/HCC'10a](#)]
- *Testing* [[VL/HCC'06](#), [VL/HCC'08](#), [TSE'09](#)]
- *Pattern Inference* [[PPDP'06](#), [ICSE'06](#), [VL/HCC'10b](#)]