A Pragmatic Perspective on Software Visualization

Arie van Deursen
Delft University of Technology
Acknowledgements

• SoftVis organizers
  - Alexandru Telea
  - Carsten Görg
  - Steven Reiss

• TU Delft co-workers
  - Felienne Hermans
  - Martin Pinzger

• The Chisel Group, Victoria
  - Margaret-Anne Storey
Outline

1. Questions & Introduction
   Software visualization reflections

2. Zooming in:
   Visualization for end-user programmers

3. Zooming out:
   Software visualization reflections revisited

4. Discussion
   But not just at the end
Exploring Legacy Systems Using Types*

Arie van Deursen  
Leon Moonen

CWI, P.O. Box 94079  
1090 GB Amsterdam, The Netherlands  
http://www.cwi.nl/~{arie,leon}/  
{arie,leon}@cwi.nl

ABSTRACT

We show how hypertext-based program understanding tools can achieve new levels of abstraction by using inferred type information for cases where the subject software system is written in a weakly typed language. We propose TYPEEXPLORER, a tool for browsing COBOL legacy systems based on these types. The paper addresses (1) how types, an invented abstraction, can be presented meaningfully to software re-engineers; (2) the implementation techniques used to construct TYPEEXPLORER; and (3) the use of TYPEEXPLORER for understanding legacy systems, at the level of individual statements as well as at the level of the software architecture – which is illustrated by using TYPEEXPLORER to browse an industrial COBOL system of 100,000 lines of code.


Keywords and Phrases: Software maintenance, program understanding, program analysis, type inference, documentation generation, variable usage, hypertext.


Note: Work carried out under projects SEN 1.1, Software Renovation and SEN 1.5, Domain-Specific Languages.

1. Introduction

Software immigrants, employees that are added to an existing software system in order to expand the existing and their own code as time between development and maintenance goes by. As a consequence, maintenance tasks become difficult, expensive, and error prone.

To reduce these problems, organizations are interested in tools that understand and represent the hypertext for programs [18]. Within the typically thousands of thousands of lines of code, individual statements can be isolated out early. The engineer can then examine these, Figure 1, down and by having a lot of work on the "opportunities".

Such a hypertext-based understanding tool has especially those cases where the output is low-level, more manageable, in order to bring the system's functionality and, more technical, at a higher level.

The fundamental idea is to arrive at a system that is just cross referencing to that level of understanding. Our solution is based on looking at system understanding.

For type information at the level of types for programs, type inference, and type signatures, many of the existing software systems, however, are written in older languages with very weak type systems.
Ali Mesbah, Arie van Deursen, Danny Roest
Invariant-based automated testing of modern web applications. ICSE’09, TSE subm.
Bas Cornelissen, Andy Zaidman, Arie van Deursen,
A Controlled Experiment for Program Comprehension through Trace Visualization.
*IEEE Transactions on Software Engineering, 2010*
A. Zaidman, B. van Rompaey, A. van Deursen and S. Demeyer. Studying the co-evolution of production and test code in open source and industrial developer test processes through repository mining. *Empirical Software Engineering, 2010*
A. Hindle, M. Godfrey, and R. Holt.
Software Process Recovery using Recovered Unified Process Views. ICSM 2010
Observations

1. My visualizations leave room for improvement...
2. Some very cool results are never applied 😞
3. Software visualizations in context can be successful
4. Simpler might be more effective
5. What is our perspective on evaluation?
What is “Exciting” in an Engineering Field?

1. Invention of wholly new ideas and directions
2. Work of promise that illuminates #1
3. Early application of #2 showing clear prospect of benefit
4. Substantial exploitation of #3 yielding measurable societal benefits
5. Maturing of #4 with widespread adoption by practitioners
What Can We Learn From The Social Sciences?

Paradigms shaping the practice of research:

- Post-positivism
- Social constructivism
- Participatory / advocacy
- Pragmatism
Post-positivism

• *Conjectures and Refutations*: The Growth of Scientific Knowledge

• Testing of hypotheses

• A priori use of theory
Pragmatism

- Clarify meanings of intellectual concepts by tracing out their “conceivable practical consequences”.
  (Charles Peirce, 1905)

- Do not insist upon antecedent phenomena, but upon consequent phenomena;
  Not upon the precedents but upon the possibilities of action
  (John Dewey, 1931)
Pragmatic Considerations

• Not every belief that is “true” is to be acted upon

• Not committed to single research method

• Research occurs in social (and technological) context

• Research builds up “group knowledge”
The Qualitative Research Palette

• Measuring applicability?

• The outcome as a narrative

• Multi-facetted validity

• Case studies
• Ethnography
• Participant observation
• Grounded theory
• Phenomenology
• Narrative studies
• Participative inquiry
• Interviewing
• Document analysis
• ...

C. B. Seaman. Qualitative methods in empirical studies of software engineering. *IEEE TSE, 1999*
Part II: Zooming In

Felienne Hermans, Martin Pinzger, Arie van Deursen
Supporting Professional Spreadsheet Users by Generating Leveled Dataflow Diagrams.
Corporate Spreadsheets

- Decision making
- Financial reporting
- Forecasting
- Business data
Spreadsheet Research

• Spreadsheet Risks Interest Groups
  – Managing & identifying spreadsheet errors

• “End Users Shaping Effective Software” (2005…)
  – Spreadsheet corpus, testing, debugging, surveys
  – ICSE, TOSEM, TSE, Comp. Surveys, VL/HCC, CHI,…
  – Nebraska, CMU, Oregon State, Washington, …

F. Hermans, M. Pinzger, and A. van Deursen.
Automatically Extracting Class Diagrams from Spreadsheets. ECOOP 2010.
130 billion Euro in “assets under management”
1600+ employees
Excel #1 software system
3 hours per day
On average > 5 years old
On average 13 users each
Objectives and Approach

Objective:
• Assist end-user programmers in spreadsheet comprehension

Approach:
• Collect information needs in interviews
• Provide tool addressing key information needs
• Evaluate tool strengths and weaknesses in concrete Robeco setting
Information Need Identification

**Interview** 27 people:
- Bring a typical spreadsheet
- Maximize variance in knowledge, experience, departments

**Qualitative** data collection
- Discover needs through open-ended questions
- “Tell us about your spreadsheet”
Grounded Theory

- Systematic procedure to discover theory from (qualitative) data
- Theoretical sensitivity
- Theoretical coding
- Open coding
- Theoretical sampling
- Constant comparative method
- Selective coding
- Memoing


Intermezzo: Eclipse Testing

Interested in the testing culture of the Eclipse community?

Participate in our study on testing practices for plug-in systems, and get free access to the study findings anticipating to identify testing best practices and pitfalls!

http://the-eclipse-study.blogspot.com/
Result I: Transfer Scenarios

- **S1: Transfer to colleague (50%)**
  - new colleague; employee leaves; additional users.
- **S2: Check by auditor (25%)**
  - Assess risks, design, documentation.
- **S3: To IT department (25%)**
  - Replace by custom software
  - Increased importance / complexity, multiple people, ...
Result II: Information Needs

• N1: How are worksheets related? (45%)

• N2: Where do formulas refer to (40%)

• N3: What cells are meant for input (20%)

• N4: What cells are meant for output (20%)
Observation: Top information needs related to “flow of data” through the spreadsheet

Research Question: How can we leverage dataflow diagrams to address information needs?
<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Wk 13</td>
</tr>
<tr>
<td>2</td>
<td>Wk 14</td>
</tr>
<tr>
<td>3</td>
<td>Difference</td>
</tr>
<tr>
<td>4</td>
<td>Percentage</td>
</tr>
</tbody>
</table>

**Cell Classification**

**Data Block Identification**

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Wk 13</td>
</tr>
<tr>
<td>2</td>
<td>Wk 14</td>
</tr>
<tr>
<td>3</td>
<td>Difference</td>
</tr>
<tr>
<td>4</td>
<td>Percentage</td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

**Graph Construction**

**Name Resolution & Replacement**

**Grouping**

```
B2
  /-
  | B3
  / /100
  | B4

Wk13
  /-
  | Wk14
  / Difference
  / /100
  | Percentage

Wk13
  /-
  | Wk14
  / Difference
  / /100
  | Percentage
```
Directed Graph Markup Language

- Visual Studio 2010 DGML graph browser
- Zooming
- Collapsing / expanding levels
- Grouping of multiple arrows
- Butterfly mode / slicing
- Graph editing (deletion, coloring, leveling)

Create prototype (GyroSAT) aimed at collecting initial user feedback
Example Grading Sheet

<table>
<thead>
<tr>
<th>Questions</th>
<th>1a</th>
<th>1b</th>
<th>1c</th>
<th>2a</th>
<th>2b</th>
<th>3a</th>
<th>3b</th>
<th>points</th>
<th>grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>max score</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td>18</td>
<td>10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Results</th>
<th>ID</th>
<th>Presence</th>
<th>1a</th>
<th>1b</th>
<th>1c</th>
<th>2a</th>
<th>2b</th>
<th>3a</th>
<th>3b</th>
<th>points</th>
<th>grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laura Chase</td>
<td>1234</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>4.5</td>
</tr>
<tr>
<td>Richard Griffen</td>
<td>1235</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>13</td>
<td>7.5</td>
</tr>
<tr>
<td>Winifred Prior</td>
<td>1236</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>12</td>
<td>7</td>
</tr>
<tr>
<td>Alex Thomas</td>
<td>1237</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Iris Chase</td>
<td>1238</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>15</td>
<td>8.5</td>
</tr>
</tbody>
</table>
Evaluation

• Is this actually useful for spreadsheet professionals?

• Evaluation I: *Interviews*

• Evaluation II: *Actual transfer tasks*

McGrath: “Credible empirical knowledge requires consistency or convergence of evidence across studies based on different methods.”
27 Interviews
They really outperform the Excel Audit Toolbar, because they show the entire view of the sheet.
As analysts we are used to thinking in processes and therefore this kind of diagrams is very natural to us.
This diagram is very complex, I’m not sure it can help me
I would prefer to have the visualization inside Excel
Case Studies: Experimental design

• Monitor 9 *actual* transfer tasks:
  – 3 for each category

• Each task involved:
  – Two participants: *expert* to *receiver*
  – ~1 hour
  – Laptop with GyroSAT; desktop with Excel
  – Participant observation & reflective questions

• Only helped if participants got stuck
# Spreadsheets Involved

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Spreadsheet Description</th>
<th>Worksh.</th>
<th># Rows</th>
<th># Col.</th>
<th># Cells</th>
<th># Form.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Avg</td>
<td>Min</td>
<td>Max</td>
<td>Avg</td>
<td>Min</td>
<td>Max</td>
</tr>
<tr>
<td>S1a</td>
<td>9</td>
<td>393</td>
<td>3</td>
<td>11</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>S1b</td>
<td>5</td>
<td>55</td>
<td>31</td>
<td>14</td>
<td>14</td>
<td>16</td>
</tr>
<tr>
<td>S1c</td>
<td>16</td>
<td>88</td>
<td>19</td>
<td>29</td>
<td>6</td>
<td>73</td>
</tr>
<tr>
<td>S2a</td>
<td>42</td>
<td>272</td>
<td>15</td>
<td>12</td>
<td>4</td>
<td>27</td>
</tr>
<tr>
<td>S2b</td>
<td>10</td>
<td>3269</td>
<td>9</td>
<td>17</td>
<td>7</td>
<td>28</td>
</tr>
<tr>
<td>S2c</td>
<td>6</td>
<td>376</td>
<td>38</td>
<td>9</td>
<td>2</td>
<td>28</td>
</tr>
<tr>
<td>S3a</td>
<td>4</td>
<td>48</td>
<td>3</td>
<td>14</td>
<td>3</td>
<td>26</td>
</tr>
<tr>
<td>S3b</td>
<td>5</td>
<td>100</td>
<td>5</td>
<td>26</td>
<td>14</td>
<td>32</td>
</tr>
<tr>
<td>S3c</td>
<td>18</td>
<td>62</td>
<td>1</td>
<td>16</td>
<td>5</td>
<td>40</td>
</tr>
</tbody>
</table>
Spreadsheets Involved

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Spreadsheet Description</th>
<th>Worksh.</th>
<th># Rows</th>
<th># Col.</th>
<th># Cells</th>
<th># Form.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Avg</td>
<td>Min</td>
<td>Max</td>
<td>Avg</td>
<td>Min</td>
</tr>
<tr>
<td>S1a</td>
<td>Shares risk management</td>
<td>9</td>
<td>393</td>
<td>3</td>
<td>1989</td>
<td>11</td>
</tr>
<tr>
<td>S1b</td>
<td>Top and bottom 5 stock performance</td>
<td>5</td>
<td>55</td>
<td>31</td>
<td>74</td>
<td>14</td>
</tr>
<tr>
<td>S1c</td>
<td>Combining data from different sources for weekly reports</td>
<td>16</td>
<td>88</td>
<td>19</td>
<td>294</td>
<td>29</td>
</tr>
<tr>
<td>S2a</td>
<td>Overview of portfolio data</td>
<td>42</td>
<td>272</td>
<td>15</td>
<td>611</td>
<td>12</td>
</tr>
<tr>
<td>S2a</td>
<td>Overview of gain and loss of all trades for one week</td>
<td>10</td>
<td>3269</td>
<td>9</td>
<td>32442</td>
<td>17</td>
</tr>
<tr>
<td>S2a</td>
<td>Constructing a stock portfolio</td>
<td>6</td>
<td>376</td>
<td>38</td>
<td>1000</td>
<td>9</td>
</tr>
<tr>
<td>S3a</td>
<td>Comparison of stock data from two sources</td>
<td>4</td>
<td>48</td>
<td>3</td>
<td>26</td>
<td>14</td>
</tr>
<tr>
<td>S3b</td>
<td>Data on loaned stocks and their dividend date</td>
<td>5</td>
<td>100</td>
<td>5</td>
<td>374</td>
<td>26</td>
</tr>
<tr>
<td>S3c</td>
<td>Calculating which trades to perform in the future</td>
<td>18</td>
<td>62</td>
<td>1</td>
<td>158</td>
<td>16</td>
</tr>
</tbody>
</table>
Case Studies S1a,b,c
Transfer to Colleague

All: Surprised by (visualized) complexity of own spreadsheets

S1a: Visualization gives expert a story line

S1c: Visualization helps receiver to understand overview

S1a: Missed support for VBA
Case Studies S2a,b,c
Audit

S2a: Picture leads me straight to the fishy parts
S2b: More helpful than old approach (click all cells)
S2b: Helps to spot errors on a new level
S2a/S2b: Missing connections with environment
Case Studies S3a,b,c
Transfer to IT Dept

All: Receivers understood experts much better with the use of dataflow diagrams

S3b: Top level diagram basis for architecture

All: Storyline, zooming into details

All: Used node names from diagrams to explain excel sheet

S3a: Multiple calculations in sheets not separated
Spreadsheet Visualization

• Threats to validity:
  – Tradeoff realism versus repeatability
  – Robeco spreadsheets only
  – Non-random group of participants

• Simple but effective:
  – Helps to tell the spreadsheet story
  – Works for complex, realistic spreadsheets
  – VBA + Excel integration high on wish list
Methodological Pride!

• What are our knowledge claims?

• What are the corresponding research methods?

# Design Science

<table>
<thead>
<tr>
<th>Type</th>
<th>Conditions</th>
<th>Control of Practice</th>
<th>Example</th>
<th>Users</th>
<th>Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Illustration</td>
<td>no</td>
<td>yes</td>
<td>small</td>
<td>designer</td>
<td>illustration</td>
</tr>
<tr>
<td>Opinion</td>
<td>imagined</td>
<td>yes</td>
<td>any</td>
<td>stakeholder</td>
<td>support</td>
</tr>
<tr>
<td>Lab demo</td>
<td>no</td>
<td>yes</td>
<td>realistic</td>
<td>designer</td>
<td>knowledge</td>
</tr>
<tr>
<td>Lab experiment</td>
<td>no</td>
<td>yes</td>
<td>artificial</td>
<td>subjects</td>
<td>knowledge</td>
</tr>
<tr>
<td>Benchmark</td>
<td>no</td>
<td>yes</td>
<td>standard</td>
<td>designer</td>
<td>knowledge</td>
</tr>
<tr>
<td>Field trial</td>
<td>yes</td>
<td>yes</td>
<td>realistic</td>
<td>designer</td>
<td>knowledge</td>
</tr>
<tr>
<td>Field experiment</td>
<td>yes</td>
<td>yes</td>
<td>realistic</td>
<td>stakeholder</td>
<td>knowledge</td>
</tr>
<tr>
<td>Action case</td>
<td>yes</td>
<td>no</td>
<td>real</td>
<td>designer</td>
<td>knowledge and change</td>
</tr>
<tr>
<td>Pilot project</td>
<td>yes</td>
<td>no</td>
<td>realistic</td>
<td>stakeholder</td>
<td>knowledge and change</td>
</tr>
<tr>
<td>Case study</td>
<td>yes</td>
<td>no</td>
<td>real</td>
<td>stakeholder</td>
<td>knowledge and change</td>
</tr>
</tbody>
</table>

Roel Wieringa. Design Science Methodology. Presentation for Deutsche Telekom, 2010 (also ICSE, RE tutorial)
A Priori Engagement with Users

- Understand existing way of working
- Identify problems
- Embed solutions
Software = Peopleware

Evaluations are
• qualitative
• incomplete
• subjective

Evidence must
• grow
• and be criticized
 Visualization = Communication

- Beyond individual comprehension
- Evaluate team interaction & collaboration
End-User Programming
Start a Company!

Around 70% of all your company’s data is most likely contained in Excel sheets, not in software or paper.

PROFESSIONALIZING SPREADSHEET USE...
...BY USING INFOTRON’S DATAFLOW VISUALIZATION

INFOTRON.NL