Assisted Software Exploration using Formal Concept Analysis

Paul Heckmann
Daniel Speicher

{heckmann, dsp}@cs.uni-bonn.de
Introduction

Structure “Catalog”

1. Formal Concept Analysis
2. Exploration Support
## Background

**Formal Concept Analysis**  • B. Ganter, R. Wille; 1996

**Structural Software Decomposition**  • P. Tonella, G. Antonioli; 1998  
• G. Arévalo; 2004
FCA ➤ Formal Context

n = 3
(A, D, E)
(A, C, D)
(A, B, D)
(A, B, C)
## FCA \quad Formal Context

\[
\begin{align*}
\text{n} & = 3 \\
(A, D, E) & \quad \text{data} \\
(A, C, D) & \quad \text{attributes} \\
(A, B, D) & \quad \text{inheritance} \\
(A, B, C) & \quad \text{association}
\end{align*}
\]
**FCA**  
**Formal Context**

<table>
<thead>
<tr>
<th></th>
<th>inh(2, 1)</th>
<th>inh(3, 1)</th>
<th>assoc(3, 1)</th>
<th>inh(3, 2)</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A, B, C)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(A, B, D)</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(A, C, D)</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(A, D, E)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[(A, B, C), \text{ inh}(2, 1) \Rightarrow B \text{ inherits from A}\]
FCA  Formal Concept

Intent: \{\text{inh}(2,1), \text{inh}(3,1), \text{assoc}(3,1)\}

Extent: \{(A,B,C)\}

+ (A, B, D)
FCA Role Model Transformation

(A, B, C)  (A, B)  (A, C)

(A, B)  (A, C)
# Runtime

<table>
<thead>
<tr>
<th></th>
<th>JUnit 4.7</th>
<th>Cultivate</th>
<th>JHotDraw 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>#classes</td>
<td>143</td>
<td>607</td>
<td>625</td>
</tr>
<tr>
<td>n = 2</td>
<td>&lt;1s</td>
<td>2s</td>
<td>3s</td>
</tr>
<tr>
<td>n = 3</td>
<td>&lt;1s</td>
<td>3s</td>
<td>14s</td>
</tr>
<tr>
<td>n = 4</td>
<td>1s</td>
<td>187s</td>
<td>130s</td>
</tr>
<tr>
<td>n = 5</td>
<td>13s</td>
<td>x</td>
<td>2901s</td>
</tr>
</tbody>
</table>

Quad @2.83GHz, 4GB RAM
Concept Assessment

- Specificity
- Prevalence
- Importance
- ...

Exploration Support
Exploration Support  
Class Prominence
Class Prominence

Exploration Support

0.75
(A, D, E)
(A, C, D)
(A, B, D)
(A, B, C)

0.50
0.50
0.50

0.25

1.00
0.50

13
Core Concepts

Roles

1. (Test, TestSuite, JUnit4TestAdapter)
2. (Test, TestSuite, TestCase)
3. (Test, TestSuite, TestDecorator)

$r \in \{1, 2, 3\} \equiv \text{roles}$

$P_{c,r} \equiv \text{set of role players}$
Core Concepts

Roles

1

2

3

(Test, TestSuite, JUnit4TestAdapter)

(Test, TestSuite, TestCase)

(Test, TestSuite, TestDecorator)

\[ p_{c,1} = \{\text{Test}\} \]

\[ p_{c,2} = \{\text{TestSuite}\} \]

\[ p_{c,3} = \{\text{JUnit4TestAdapter, TestCase, TestDecorator}\} \]
Core Concepts

\[ core_c = \max_{r \in \{1, \ldots, n\}} \frac{\sum_{i \in p_{c,r}} prom_{n,i}}{|p_{c,r}|} \]

↑ prominence of role players
↓ # locations in the code
Filtering

(A, D, E)
(A, C, D)
(A, B, D)
(A, B, C)
Future Work

• Comparison of ways to compute class prominence
• Evaluation
• Further concept metrics
• Extension/Refinement of Relationship Model
• Automatic name suggestion for concepts/roles
Contributions

Summary

• Formal Concept Analysis:
  ✓ Usable even for classes > 600.

• Concept Model
  ✓ Structural + Behavioral Relationships.
  ✓ Engaged FCA complexity by Iterative Analysis.
  ✓ Abstraction from Structures of Classes to Concepts with Roles and -Players.

• Exploration Support
  ✓ Filtering by Classes.
  ✓ Efficient Filter Suggestion through Class Prominence.
  ✓ Structure Importance Assessment
public class ClassA extends SuperType {
    public class ClassA implements Abstract {
        private ClassB scalarClassB;
        private List<ClassB> vectorClassB;
        private ClassB[] arrayClassB;
    }
}
public ClassA() {
    scalarClassB = new ClassB();
}
public void forwardsCall() {
    scalarClassB.forwardsCall();
}
public ClassB producesClassB() {
    return new ClassB();
}
FCA Complexity

\[ O(|G|^2 \cdot |M| \cdot |C|) \]
Iterative Analysis

1. Iteration (Structural)

2. Iteration (Structural + Behavioral)
FCA  Unused Generalization

Function?
### Appendix ➤ Number Concepts

<table>
<thead>
<tr>
<th></th>
<th>JUnit 4.7</th>
<th>Cultivate</th>
<th>JHotDraw 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>#classes</td>
<td>143</td>
<td>607</td>
<td>625</td>
</tr>
<tr>
<td>n = 2</td>
<td>5</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>n = 3</td>
<td>16</td>
<td>45</td>
<td>38</td>
</tr>
<tr>
<td>n = 4</td>
<td>24</td>
<td>154</td>
<td>141</td>
</tr>
</tbody>
</table>