

ECCO: Extraction and Composition for Clone-and-Own

Lukas Linsbauer

Stefan Fischer

Roberto Erick Lopez-Herrejon

Alexander Egyed



Institute for Software Systems Engineering Johannes Kepler University Linz, Austria



Motivation

- Companies often build custom-tailored software
- If successful, they build products for other customers
 - Clone-and-Own
- End up with similar software products
 - Also e.g. Testing, System Engineering, ...



Clone-and-Own

- a.k.a. Copy and Paste
- Steps:
 - Extracting reusable Code (copy)
 - Composing reusable Code (paste)
 - Completing Products
 - Adding missing Code
 - Modify or Remove Code



Drawing Application





Drawing Application

Product 1 (P1)	Product 2 (P2)	Product 3 (P3)
Base Line Wipe	Base Line Color	Base Line Rect Color
<pre>class Canvas { List<line> lines; void wipe() { this.lines.clear(); } } class Line { Line(Point start) { } class Main extends JFrame{ initContentPane() { toolPanel.add(lineButton); toolPanel.add(wipeButton); } } </line></pre>	<pre>class Canvas { List<line> lines; void setColor(String c){ } class Line { Line(Color c, Point start){ } class Main extends JFrame{ initContentPane(){ toolPanel.add(lineButton); toolPanel.add(colorsPanel); } }</line></pre>	<pre>class Canvas { List<line> lines; List<rect> rects; void setColor(String c) { } class Line { Line(Color c, Point start) { } class Rect { Rect(Color c, int x, int y) { } class Main extends JFrame{ initContentPane() { toolPanel.add(rectButton); toolPanel.add(colorsPanel); toolPanel.add(colorsPanel); } } </rect></line></pre>



Clone-and-Own Example

P4 (Base, Line, Rect, Wipe)



Composite and the composite an



Clone-and-Own

- Benefits
 - Pay as you go: no upfront investment, like for developing a Software Product Line (SPL)
 - Intuitive: easier to build individual products without having to consider variability
 - Unforeseeable future: Company might not know which features will be required in the future

Problems

- Lack systematic methodology and tool support
- Extraction / Composition get significantly more complex with each new product



Goals

- Automate the Clone-and-Own approach
 - Extraction & Composition
- Support for different kinds of artifacts
 - Source code
 - Models
 - Tests





ECCO (Extraction & Composition for Clone-and-Own)





Extraction

- Traces: associations between <u>features or feature</u> <u>interactions</u> and <u>artifacts</u>
- Order: Sequence of artifacts
- Dependencies: relationships between traces





Extraction – Big Picture

Product 1 (P1)	Product 2 (P2)	Product 3 (P3)
Base Line Wipe	Base Line Color	Base Line Rect Color
<pre>class Canvas { List<line> lines; void wipe() { this.lines.clear(); } } class Line { Line(Point start) { } class Main extends JFrame{ initContentPane() { toolPanel.add(lineButton); toolPanel.add(wipeButton); } }</line></pre>	<pre>class Canvas { List<line> lines; void setColor(String c){ } class Line { Line(Color c, Point start){ } class Main extends JFrame{ initContentPane(){ toolPanel.add(lineButton); toolPanel.add(colorsPanel); } }</line></pre>	<pre>class Canvas { List<line> lines; List<rect> rects; void setColor(String c){ } class Line { Line(Color c, Point start){ } class Rect { Rect(Color c, int x, int y){ } class Main extends JFrame{ initContentPane(){ toolPanel.add(lineButton); toolPanel.add(rectButton); } }</rect></line></pre>
5/19/2015		<pre>toolPanel.add(colorsPanel); 11 }</pre>



Extraction – Big Picture





Extraction – Big Picture





Extraction – Feature Overlaps





Extraction – Artifact Overlaps



Extraction



JOHANNES KEPLER | JKU

Extraction



JOHANNES KEPLER | JKU



Composition

- <u>Input:</u> extracted information, set of selected features
- <u>Output:</u> composed product from artifacts in selected traces, hints for completion





Composition – P4





Completion – P4(Base, Line, Rect, Wipe)





Evaluation – Case Studies

Case-Study	#F	#P	LoC	#Art
Draw	5	12	287 - 473	491
ZipMe	7	32	5K - 6.2K	5.2K+
VOD	11	32	4.7K - 5.2K	5.5K+
GameOfLife	15	65	874 - 1.9K	1.3K+
ArgoUML	11	256	264K - 344K	192K+
ModelAnalyzer	13	5	35K - 59K	94K+

#F: Number of Features#P: Number of ProductsLoC: Range of Lines of Code#Art: Number of Distinct Artifacts



Evaluation – Scheme





Evaluation - Recall





Evaluation - Precision





Summary

- 10 products in the product portfolio already lead to an average precision and recall of higher than 95% for newly composed products (for our case studies).
- ECCO guidance (Hints) can help narrowing down the locations that have to be considered during the completion



Thank you!

Thank you for your attention!

Contacts:

lukas.linsbauer@jku.at stefan.fischer@jku.at



Want to learn more?

- Lukas Linsbauer, Roberto E. Lopez-Herrejon, Alexander Egyed: Recovering traceability between features and code in product variants. SPLC 2013
- Stefan Fischer, Lukas Linsbauer, Roberto Erick Lopez-Herrejon, Alexander Egyed: Enhancing Clone-and-Own with Systematic Reuse for Developing Software Variants. ICSME 2014
- Lukas Linsbauer, Stefan Fischer, Roberto Erick Lopez-Herrejon, Alexander Egyed: Using Traceability for Incremental Construction and Evolution of Software Product Portfolios. SST 2015
- Stefan Fischer, Lukas Linsbauer, Roberto Erick Lopez-Herrejon, Alexander Egyed: The ECCO Tool: Extraction and Composition for Clone-and-Own. ICSE 2015

Demonstration Video:

<u>https://www.youtube.com/watch?v=N6gPekuxU6o</u>