Towards Actionable Visualisation in Software Development

Leonel Merino, Mohammad Ghafari, and Oscar Nierstrasz

Software Composition Group, University of Bern

Bern, Switzerland

Background
Why is software visualisation not widely used in software development?
- Out of touch with developer needs

Prior Work
- Taxonomy and surveys
- Framework to assess tools
- Help users understand tools

Approach
- Map needs to solutions in a problem domain
- Finely-grained developer needs
Eg. Where is this method called?

Question → Problem Domain → Visualisation Tool

Research Questions
What are the goals of this literature review?
- RQ1: What are the characteristics of visualisation techniques that support developer needs?
- RQ2: How well are various problem domains supported by visualisation?

Filter 346 → 65 design study papers

Collection
http://scg.unibe.ch/research/visualisation-review

Universe
http://scg.unibe.ch/research/visualisation-review

Extraction - Need
75% stated explicit questions (need)
How were needs identified when questions weren't asked?
How many needs?

Extraction - Representation
Dense Pixel
- From Keim’s Taxonomy
- Massive sets of data

Transform
- Node links
- Explore relationships

Iconic
- Attribute values mapped to icon features

Stacked
- Hierarchical displays
- Tree maps

Data Extraction
- Evaluation: Task, Need, Audience, Data Source
- Implementation: Representation, Medium, Tool

How?
- Frequent terms
- Questions
- Goals

Extraction - Need
75% stated explicit questions (need)
How were needs identified when questions weren't asked?
How many needs?

Extraction - Representation
Dense Pixel
- From Keim’s Taxonomy
- Massive sets of data

Transform
- Node links
- Explore relationships

Iconic
- Attribute values mapped to icon features

Stacked
- Hierarchical displays
- Tree maps

Filter 346 → 65 design study papers

Table 1: VIsUALISATION TECHNIQUES SUPPORTED BY PROBLEM DOMAINS.

<table>
<thead>
<tr>
<th>Problem Domain</th>
<th>Visualisation Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>Architecture</td>
</tr>
<tr>
<td>Change History</td>
<td>Change History</td>
</tr>
<tr>
<td>Element Model</td>
<td>Element Model</td>
</tr>
<tr>
<td>System Model</td>
<td>System Model</td>
</tr>
<tr>
<td>Performance</td>
<td>Performance</td>
</tr>
</tbody>
</table>

Data Extraction Attributes

Copyright © 2023, AIIM. All Rights Reserved.

Conclusion

Covered:
- History, performance, concurrency, dependencies

Call to action:
- Rationale, Intent, Implementation, Refactoring
- Metaviz (demo at https://www.youtube.com/watch?v=qe5qiS1cmzY)

Questions?

Threats to Validity

- Bias in paper selection
- Data extraction

Others not mentioned
- ‘primary contribution’ selection
- single source for developer needs (Latoza)