

Software Visualization

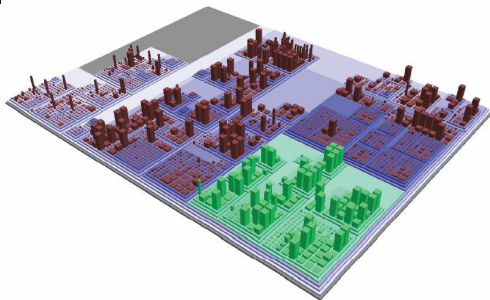
Eduardo Figueiredo

<http://www.dcc.ufmg.br/~figueiredo>

Metrics and Visualization

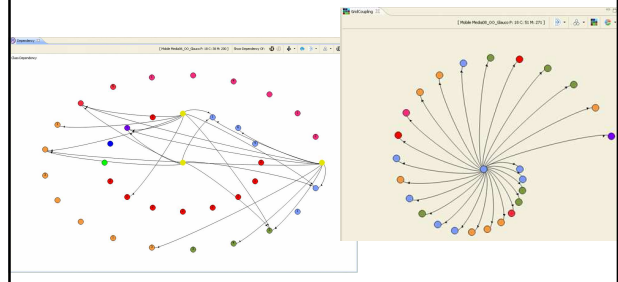
- Evaluating the design with metrics is a hard task
 - Metrics are too fine grained and generate a lot of data to analyze
- Visualization has long been used to break down the complexity of information

System as a City



R. Wetzel and M. Lanza. *Visualizing Software Systems as Cities*. In 4th IEEE International Workshop on Visualizing Software for Understanding and Analysis, pp. 92 - 99, 2007

Coupling between Classes



G. Carneiro et al. *Identifying Code Smells with Multiple Concern Views Analysis*. Brazilian Symposium on Software Engineering (SBES), Salvador, 2010.

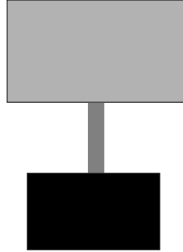
Polymetric Views

Polymetric Views

- A polymetric view is a metrics-enriched visualization of software entities and their relationships
- Notation
 - Rectangles display software entities
 - Edges represent relationships between entities

Metrics in a Polymetric View

- A single node can render up to five metrics
 - Width, Height, Position X, Position Y, and Color
- A single edge can render two metrics
 - Width and Color



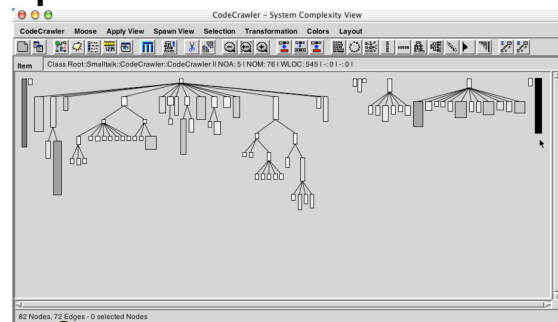
Node Size and Color

- Node Size
 - Width and height render two metrics
 - The convention is that they reflect size metrics, such as LOC and WOC
- Node Color
 - Usually, a grey scale from white to black is used to reflect a measurement
 - Higher measurement, darker node
 - It can be used for non-size metrics, such as LCOM and Cyclomatic Complexity

Node Position and Edge

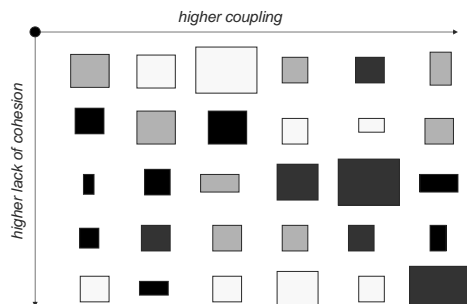
- Node Position
 - X and Y coordinates can reflect two metrics
 - This requires a fixed point as an absolute origin (gives the notion of distance)
 - It is not applicable in a tree view
- Edge Width and Color
 - Edge width and color render two metrics
 - Both metrics usually reflect the strength of a relationship, such as attribute accesses and method calls

Polymetric View as Tree View



Michele Lanza, *CodeCrawler - Polymetric Views in Action*. In 19th IEEE International Conference on Automated Software Engineering (ASE), pp. 394 - 395, 2004.

Polymetric View as Chess Board



Advantages

- A polymetric view renders numbers in a simple and effective way
- It is easy to be directly interpretable by the viewer
- It is a condensed view for scalability

Bibliography

- M. Lanza, R. Marinescu. **Object-Oriented Metrics in Practice**. Springer, 2006.
 - Section 3.2 Polymetric Views

Distribution Map

Distribution Map

- It is a visual technique composed of large rectangles containing small squares
 - Each square has a color
- Large rectangles are called **parts** of a reference partition
 - Small squares are called **elements**
 - Colors represent the **properties** we want to analyze

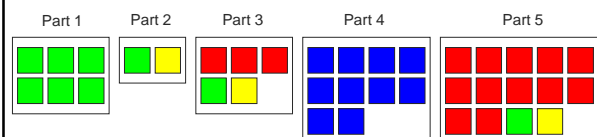
Partitions

- Distribution Map relates the reference partition (rectangles) to a comparison partition (colors)
 - Reference partition is the intrinsic structure of the system (e.g., classes and methods)
 - Comparison partition is given by a grouping or property (e.g., class owners)

Analyzing Distribution Map

- With Distribution Map, we can characterize
 - The parts with respect to the contained properties
 - The properties with respect to their distribution over the parts

Example of Distribution Map

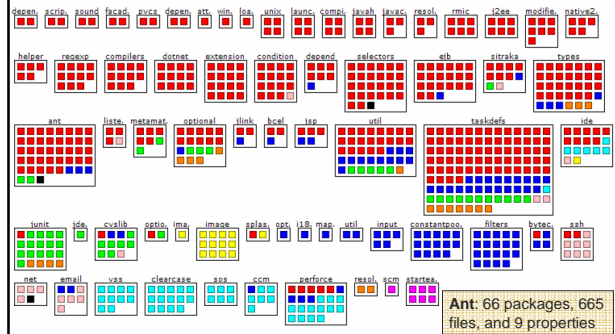


- Five parts with 6, 2, 5, 10, and 14 elements, respectively
- Four properties: Red, Blue, Green, and Yellow

Example of Statements

- Blue is well-encapsulated
 - This property is located in only one part
- Yellow is crosscutting
 - This property spread over three parts
- Part 1 and Part 4 are self-contained
 - This parts have only one property each

Scalability of Distribution Map



Bibliography

- S. Ducasse, T. Girba, A. Kuhn. **Distribution Map**. International Conference on Software Maintenance (ICSM), pp. 203-212, 2006.