university of groningen

SolidSource

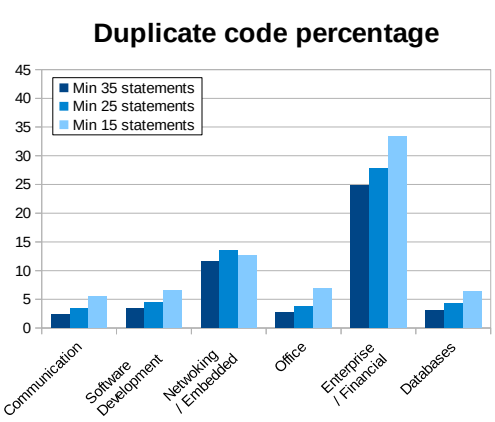
Visual Clone Analysis with SolidSDD

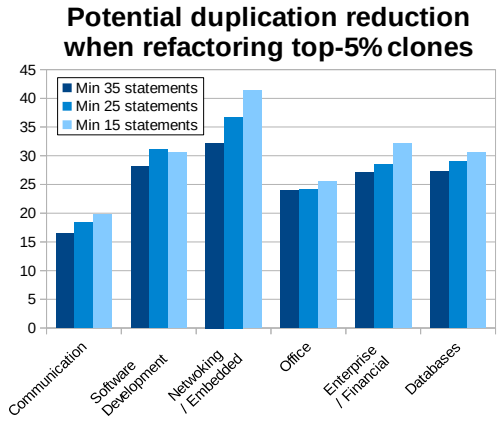
Lucian Voinea
SolidSource BV
Eindhoven, the Netherlands

Alexandru Telea
University of Groningen
the Netherlands

Presented by Avdo Hanjalić

The Situation





Software Maintenance: Over 80% of the total software life-cycle costs.
Code Reuse: Proven to reduce costs in the industry with 20-40%.
Code Clones: Between 5 - 35% of a typical code base.
Low Hanging Fruit: 15 - 40% of duplication caused by 5% of clones.

The Tool: SolidSDD

Finds structural clones:

- Supports **C**, **C++**, **C#** and **Java.**
- Shows clones vs structure (directories or syntax).
- Computes *cloned code %*, number of *distinct clones*, presence of *identifier renaming*.
- Detection configurable by *clone length*, *identifier renaming*, *gap size*, *whitespace/comments*.

Implements "overview first, zoom and filter, then details-on-demand":

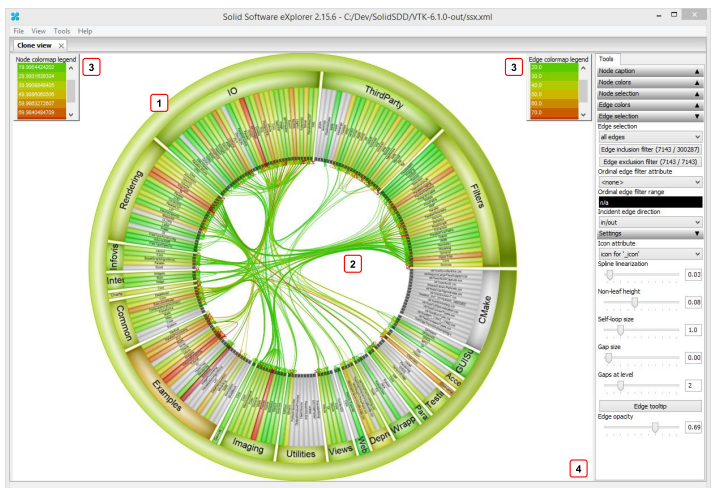
- Bundled graphs show clones vs system structure.
- Table lenses show clone and file metrics.
- Annotated text shows clones in file context and allows navigating between all pairs.
- Views are linked, allowing navigation between text, clones, and system structure.

Helps to answer questions such as:

- How are clones *distributed* vs system structure?
- Which subsystems have *high clone percentages*?
- Which files are affected by a given clone?
- How much duplication relates to the *top x% clones*?

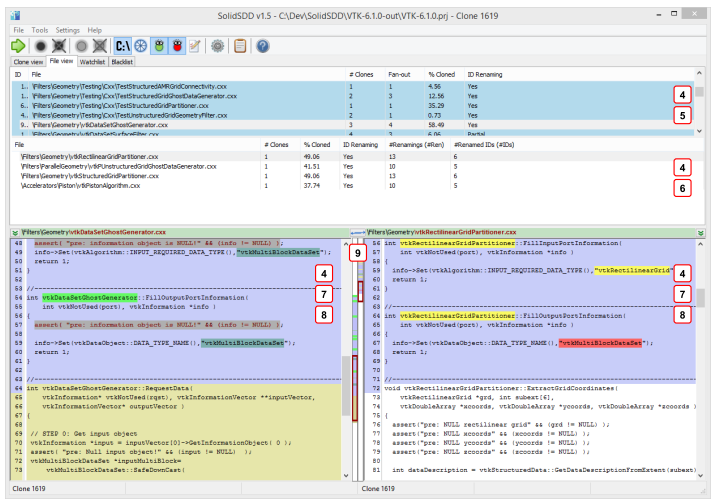
Structure View

1. **Radial tree**
   Shows system hierarchy (folders, files);
   Open/collapse subsystems by double-clicking.
2. **Hierarchical Edge Bundling**
   Aggregate clone relations atop of system structure;
   Supports finding subsystems linked by clones.
3. **Color-mapping**
   Nodes show cloned code % in a file/folder;
   Edges show cloned code % between the file-pair.

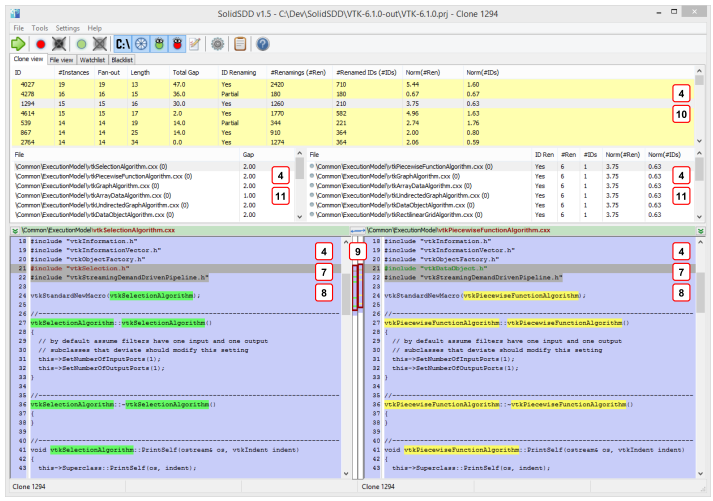


4. **Linked views**
   Select file/clone in a view to highlight it in others.
5. **File detail view**
   All files with cloned-code percentage, number of clones, and presence of identifier renaming.
6. **Clone pairs view**
   Files sharing clones with the file selected in (5).
7. **Code detail view**
   Cycle through clone-pairs by clicking a code fragment.

File & Clone Details Views

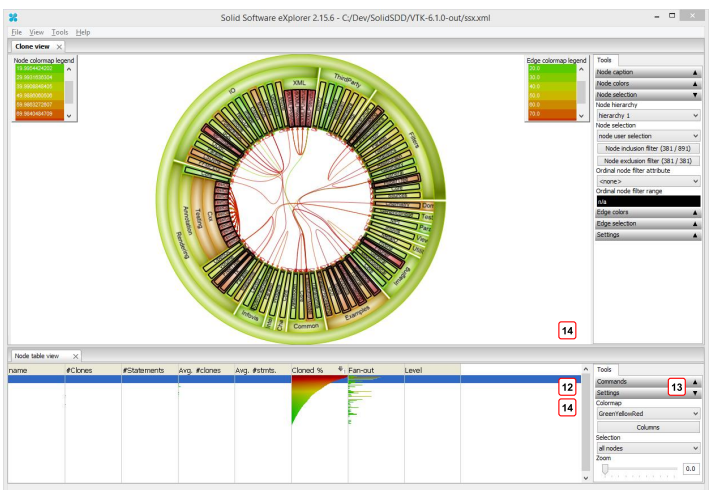


Color legend for code detail view
- non-cloned code
- clone shown in both left and right file-windows
- clone partner not shown in right file-window
- identifier renamed (shown as ■ in right window)
- abc cloned code under mouse (both file-windows)



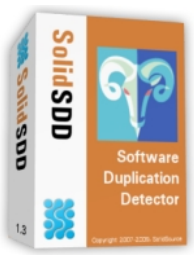
Metrics Views

8. **Color-coded text**
   Cloned code in (un)selected file-pair, renamed identifiers.
9. **Synchronized scrolling**
   Easily compare matching code fragments (6a, 6b)
10. **Clone detail view**
    All clones with number of instances, fan-out (num involved files), length, total gap, num renamed identifiers.
11. **Clone pairs view**
    Files sharing clones with the file selected in (9)



12. **Table lens**
    Files as rows with name, size, clone code %, clone fan-in (num intra-file clones), fan-out (num inter-file clones).
13. **Table lens controls**
    When zooming out, table cells are shrunk and text is replaced by colored bars showing metric values.
14. **Linked views**
    Selecting files/clones highlights them in the structure view.
    Supports investigation of e.g. top 10% largest clones.

Try it yourself!



- **Integrated** detection and exploration of code clones;
- **Free** trial and academic licenses;
- Support for **industry-sized** code bases;
- Batch-mode **report generation**, e.g. CSV and XML;
- Open API and SQLite database: easy to integrate with your own analyses or tools;
- Download, install, start using in **under 5 minutes** (Windows).

References

1. D. Reniers, L. Voinea, O. Ersoy, and A. Telea, "The Solid* toolset for software visual analytics of program structure and metrics comprehension: From research prototype to product," *Science of Computer Programming*, vol. 79, no. 1, pp. 224–240, 2014.
2. A. Telea, "Combining extended table lens and treemap techniques for visualizing tabular data," in *Proc. EuroVis*, 2006, pp. 51–58.
3. A. Hanjalic, "ClonEvol: Visualizing software evolution with code clones," in *Proc. Vissoft*, 2013, pp. 1–4.

www.cs.rug.nl/svcg

www.solidsourceit.com