Code Clone Analysis Tool: ICCA

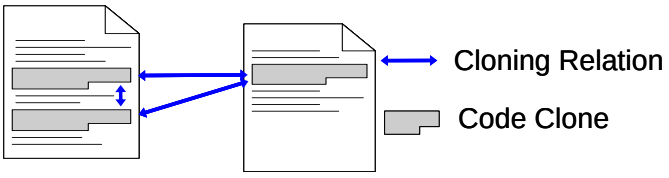
Yoshiki Higo†, Norihiro Yoshida†, Toshihiro Kamiya‡, Shinji Kusumoto†, Katsuro Inoue†

†Graduate School of Information Science and Technology, Osaka University,

‡National Institute of Advanced Industrial Science and Technology

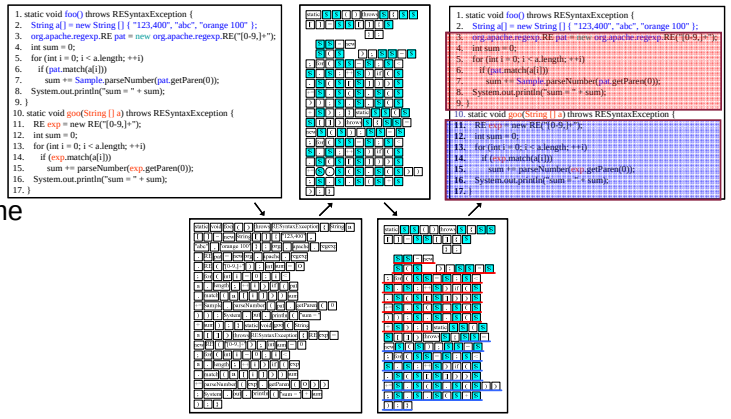
Code Clone

- A code fragment that has identical or similar fragments in source files
- Makes software maintenance more difficult
  - If we modify a code fragment and it has code clones, it is necessary to consider whether or not we have to modify each of the code clones



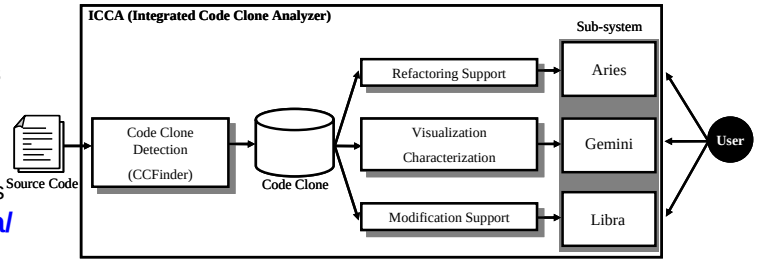
Code Clone Detection Tool: CCFinder

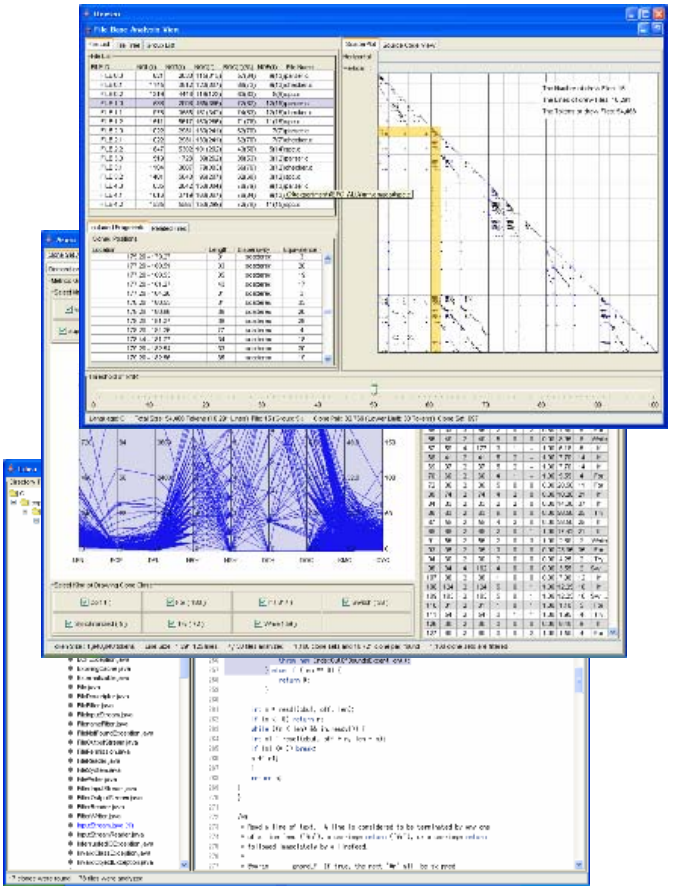
- Directly compares source code on token unit, and detects code clones
  - Normalization of name space
  - Replacement of name defined by the user
  - Removal of table initialization
- Outputs code clone information in text format
- Analyzes the system of millions line scale in practical use time
- CCFinder Official Web Page: **http://www.ccfinder.net/**



Code Clone Analysis Tool: ICCA

- Supports various scenarios on software maintenance
- There are three sub-systems for each objective
  - Gemini: Visualizes code clone information for understanding
  - Aries: Supports refactorings of code clones
  - Libra: Supports modification of source code including code clones
- Distributes our tools (ICCA/CCFinder package) to individuals and organization in different contories
  - Studies of code clones in research institute
  - Introductions into commercial software development in companies
- ICCA Official Web Page: **http://sel.ist.osaka-u.ac.jp/icca/**





Gemini Sub-system

- Visualizes code clone information
  - The user can understand the state of code clones at a glance
- Characterizes code clones and files using several metrics
  - Code clones appearing in a log of places of source code
  - Files including a lot of code clones
- Filtering out code clones which is not important

Aries Sub-system

- Identifies where should be refactored
  - detects refactoring-oriented code clones
- Suggests how each code clone can be refactored
  - Uses existing refactoring patterns
    - “Extract Method”, ”Pull Up Method”, …
  - Characterizes detected code clones using some metrics
    - The degree of code clone dispersion in the class hierarchy
    - The coupling between a code clone and its surroundings

Libra Sub-system

- Prevents the user overlooking some code fragments in the modification process
- Detects only code clones that are identical or similar to the fragment input by the user
  - For shortening detection time

contact address: **y-higo@ist.osaka-u.ac.jp**