

Source Documentation Tools

Numerous Tools Exist for Generating Documentation from Commented Code

- Examples:
 - javadoc for Java code
 - doxygen for C, C++, or Java code
- Doxygen uses the same comment style as javadoc.
- Please use one of the above in your projects.

Doxygen Features

- Developed on LINUX by Dimitri van Heesch; runs on other flavors of UNIX and Windows.
- Reads C, C++, and IDL source code and header files documented in the QT or Javadoc style.
- Produces output in HTML, in LaTeX, or as a man page.
- Allows just plain text or fancy structured HTML tags.
- Automatically generates class diagrams in HTML (as clickable image maps) and in LaTeX.

<http://www.stack.nl/~dimitri/doxygen/index.html>



Doxygen Features

- Includes a fast, rank based search engine to search for strings or words in the class and member documentation.
- Allows references to documentation generated for other projects (or another part of the same project) in a location independent way.
- Allows inclusion of source code examples that are automatically cross-referenced with the documentation.

Class Diagram Example

- Doxygen generates class diagrams in HTML or LaTeX, although they are not strictly UML.
- Class diagrams generated in HTML make use of color for certain class relationships.

Acknowledgment

- Doxygen feature slides from Eric Huang.

Doxygen Usage (on turing)

- First:
`alias doxygen /clinic/1999/mbl/tods/doxygen-1.1.0/bin/doxygen`
- Then cd to project directory and generate a configuration file:
`> doxygen -g configfile.cfg`
- Edit the config file to suit.
- Generate documentation:
`> doxygen configfile.cfg`

Doxygen Results: html

- File list: list of all files.
 - Documentation of members of each file, as generated from `/** ... */` comments.
 - Code listing of each file, with **hypertext links** to documentation of each method used.
- File members: list of all public methods and members.
- Class diagrams.

Other Possible Results

- Latex document
- man page formatted documents
- others.