Praktische Aspekte der Informatik

Thomas Löwe
Prof. Marcus Magnor
Documentation

Getting started with Doxygen
Warning!

The following slides are meant to give you a very superficial introduction.

If you want to learn more, have a look at:
http://www.stack.nl/~dimitri/doxygen/starting.html
• Why use Automatic Documentation?
• Doxygen – Basic Usage
• Doxygen – Advanced Usage
Why use Automatic Documentation?

• Help others (and future-you) understand your code

• Comment once, use several output formats
  ▪ HTML
  ▪ LaTeX
  ▪ Custom output
  ▪ … more, e.g. Qt Assistant

• Create different views of software
  ▪ Automatically generate documentation for a user group
  ▪ No need to maintain documentation multiple times
Why use Automatic Documentation?

**Top-down**
- Create model first
- Generate code stub from model
- e.g. MS Visio, ...

**Bottom-up**
- Write code first
- Update documentation while coding
- e.g. Doxygen, ...
• You can easily generate a basic doxygen file
  doxygen -g <config-file>

• Modify in text editor and run
  doxygen <config-file>

• Alternatively, you could use a GUI, e.g. doxywizard
# This tag specifies the encoding used for all characters in the config file
# that follow. The default is UTF-8 which is also the encoding used for all
# text before the first occurrence of this tag.

DOXYFILE_ENCODING = UTF-8

# The PROJECT_NAME tag is a single word (or a sequence of words surrounded
# by quotes) that should identify the project.

PROJECT_NAME = My Project

# The PROJECT_NUMBER tag can be used to enter a project or revision number.
# This could be handy for archiving the generated documentation or
# if some version control system is used.

PROJECT_NUMBER =

# The OUTPUT_DIRECTORY tag is used to specify the (relative or absolute)
# base path where the generated documentation will be put.

OUTPUT_DIRECTORY =

# The OUTPUT_LANGUAGE tag is used to specify the language in which all
# documentation generated by doxygen is written. Doxygen will use this
# information to generate all constant output in the proper language.
# The default language is English, other supported languages are:
# Afrikaans, Arabic, Brazilian, Catalan, Chinese, Chinese-Traditional,
# Croatian, Czech, Danish, Dutch, Esperanto, Farsi

OUTPUT_LANGUAGE = English

...
• Document your code!

• Several options:
  ▪ Use /*!  Or /***  instead of /*
  ▪ Use //!!  or ///!  Instead of //
  ▪ And many more…

• Example:
  
  /*! \brief Brief description. 
   * Brief description continued. 
   *
   * Detailed description starts here.
   */
There are a lot of special commands:

\texttt{\textbf{\texttt{struct}}} to document a struct.
\texttt{\textbf{\texttt{union}}} to document a union.
\texttt{\textbf{\texttt{enum}}} to document an enumeration type.
\texttt{\textbf{\texttt{fn}}} to document a function.
\texttt{\textbf{\texttt{var}}} to document a variable or typedef or enum value.
\texttt{\textbf{\texttt{def}}} to document a #define.
\texttt{\textbf{\texttt{typedef}}} to document a type definition.
\texttt{\textbf{\texttt{file}}} to document a file.
\texttt{\textbf{\texttt{namespace}}} to document a namespace.
\texttt{\textbf{\texttt{package}}} to document a Java package.
\texttt{\textbf{\texttt{interface}}} to document an IDL interface.
...
/*! \file structcmd.h
 * \brief A Documented file.
 * Details.
 */

class Test {

public:
/** An enum type.
 * The documentation block cannot be put after the enum!
 */
enum EnumType {
    ValueA, /**< enum value 1 */
    ValueB /**< enum value 2 */
};

protected:
    void member(); /**< A protected member function.

private:
    int value; /**< An integer value */

};
• Create different output
  ▪ HTML pages
  ▪ Latex files

• Graphs/Diagrams ([www.graphviz.org](http://www.graphviz.org))
• Additional options
  ▪ Include formulas (LaTeX style)
  ▪ Include graphics
  ▪ Change formatting
  ▪ Create your own styles

• Example: Images
  \image <format> <file> ["caption"] [<sizeindication>=<size>]

Doxygen – Advanced Output