ODD Documentation Tool Tutorial

For

TRAMP

Bernhard Zaun
December 10th 2001

Agenda

- Overview of ODD

- Why to document source code

- Doxygen
  - What is Doxygen
  - Config file
  - Documentation blocks
  - Special commands
  - Examples

- Bibliography

Object Design Document

- Introduction
    - Object design trade-offs
    - Interface documentation guidelines
    - Definitions, acronyms and abbreviations
    - References

- Packages

- Class interfaces

- Classdiagram (Note: Not mentioned in [1])

- Glossary

Why to Document Code

- For yourself
- For your team members
- For other teams
- For successors

- To keep an overview of the code
- To understand code as quickly as possible

- To acquire a Schein :o)

What is Doxygen ?

/ Documentation system for documented source code in

" C++ / C

" Java

" IDL

/ To keep documentation consistent with source code

/ Useful for quickly finding your way in large source distributions

Doxygen

- Can create several formats like
  - HTML and compressed HTML
  - Latex, so PS and hyperlinked PDF
  - RTF
  - unix manpages

- Code structures / classdiagrams
  - dependency graphs
  - inheritance diagrams
  - collaboration diagrams

- Can also be used for any other

Doxygen

- Portable
  - Was developed under Linux
  - Runs on most unix systems
  - Also available for M$ Windows

- Can be integrated into ANT (see [2])

- Consists of
  - `doxygen` : generates the documentation
  - `doxytag` : for external references
  - `doxysearch` : can use search engine
  - `doxywizard` : GUI to edit config files

Doxygen Howto

1. Generate template configuration file with
   ```
   doxygen -g <config_file>
   ```
2. Configure it with
   ```
   vi <config_file>
   ```
3. Document source code with special documentation blocks
4. Run doxygen
   ```
   doxygen <config_file>
   ```

5. Enjoy the result

Doxytag Howto

/   Can create *tag files to be used for* `doxygen`

     /   *tag file contains information about internal or external HTTP documents, formerly created by* `doxygen`*, even if the source code is not there*

Shell examlpe:

```
doxytag -t <tag_file>.tag
                  <HTML_source_dir>/html
```

Doxytag Howto

- *Can create a search index which can be used for* `doxysearch`
    - *search index contains information of all the words of the documentation*

        Shell examlpe:
        `doxytag -s <search_index_file>.idx`

- *Can also be run by* `doxygen`
  (see configuration file)

Doxysearch Howto

- Fast portable search engine, run as CGI
  - HTTP deamon is necessary with CGI support

- Configured in the `doxygen` config file

- `doxygen <config_file>` creates
  - CGI script (see CGI_NAME), which has to be copied to the `cgi-bin` directory
  - `search.cfg` containing the absolute URL of the documentation and the CGI script

Configuration File

- / One configuration file (`Doxyfile` by default) for one project or a source tree, scanned recursively

- / Containing tags (like ANT or make) which can be categorized

- / Free-form ASCII text
  - / Comments start with `#` , end with line
  - / Environment variables: `$(ENV_NAME)`
  - / Include meta config file:
    `@INCLUDE = meta config file name`

Config Tags Categories

/ ## General options:
  - / PROJECT_NAME, PROJECT_NUMBER,OUTPUT_DIRECTORY, ...

/ ## Input related options:
  - / INPUT, FILE_PATTERNS, RECURSIVE, EXCLUDE, ...

/ ## Dot options:
  - / CLASS_DIAGRAMS, CLASS_GRAPH, GRAPHICAL_HIERARCHY,..

/ ## LaTeX related options:
  - / PAPER_TYPE, LATEX_HEADER, EXTRA_PACKAGES, ...

/ ## External reference options:
  - / PERL_PATH, TAGFILES, ...

Config Tags Categories

/ Source browsing related options

/ Alphabetical index options

/ Preprocessor related options

/ Options related to warning and progress messages

/ Search engine options

/ HTML related options

/ RTF related options

/ Man page related options

Config File Example

/ **Doxyfile:**

```
PROJECT_NAME            = TRAMP
PROJECT_NUMBER          = 0.1
OUTPUT_DIRECTORY        = doc
EXTRACT_ALL             = YES
HIDE_UNDOC_CLASSES      = NO
WARNINGS                = YES
INPUT                   = tramp.java
FILE_PATTERNS           = *.java
RECURSIVE               = YES
EXCLUDE_PATTERNS        = *bak.java
ALPHABETICAL_INDEX      = YES
PERL_PATH               = /usr/bin/perl
CLASS_DIAGRAMS          = YES
SEARCHENGINE            = YES
LATEX_HEADER            = <latex header file>
GENERATE_RTF            = NO
```

Special Doc Blocks

- Can be put anywhere in the source **except**:
  - inside the body of a function
  - inside a normal C style comment block
    /* ... */

- Usually in front of the declaration, class, ... which is documented

- Else *structural commands* or *compound comments* are needed

Documentation Examples

- Examples:
  - ```
    /*!
       ... some text ...
    */
    ```

  - ```
    /**          (Note: do not forget the 2nd * here !)
      * ... some more text ...
    */
    ```

  - ```
    //! ... one line of text ...
    ```

  - ```
    /// ... one more line of text ...
    ```

Compound Comments

- **Blocks:** `/*!< ... */` **or** `/**< ... */`
- **One line:** `//!< ...` **or** `///< ...`

- `/*! An example class with compound comments */`

```
class CompoundExample {
  public:
    /** An enum type.
     *  This block cannot be put after enum!
     */
    enum EnumType
    {
      int EVal1,      /**< enum value 1. */
      int EVal2       //!< enum value 2.
    };
};
```

Parsing the Code

/ Special commands are executed

/ Even **some** HTML commands are interpretet (see [3] for a detailed list)

/ Whitespace and * are removed from the beginning of lines

/ Blank lines are interpreted as paragraphs

/ Links are created (partly automatically)

Special Commands

- Structural commands start with '@' or ' \'

- Some have one or more arguments
  - `<single_word>`
  - `(rest of the line)`
  - `{rest of the paragraph}`
  - `[optional arguments]`

- Unrecognized commands like normal text

- Special caracters escaped with \
  `\, @, &, $, #`

- Some even with \\
  `<, >` would look like `\\<, \\>` in docu

Special Commands

/ **Structural indicators**
/ `\class <name> [<header-file>]`
`[<header-name>]`
/ `\fn (function declaration)`

/ **Section indicators**
/ `\author { list of authors }`
/ `\if <section-label>`
/ `\return { description of the return`
`value }`

/ **Commands to create links**
/ `\section <sec-name> (sec title)`
/ `\link <link-object>`
/ `\endlink`

Special Commands

- Commands for displaying examples
  - `\htmlinclude <file-name>`
  - `\line ( pattern )`

- Commands for visual enhancements
  - `\a <word>      HTML: <I><word></I>`
  - `\b <word>      HTML: <B><word></B>`
  - `\c <word>      HTML: <TT><word></TT>`
  - `\arg { item-description }`
  - `\image <format> <file> ["caption"]`
    `[<sizeindication>=<size>]`

Docu Examples

- Source code documentation:

```
The distance between \f$(x_1,y_1)\f$
and \f$(x_2,y_2)\f$ is
\f$\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}\f$.
```

- Results in (Latex, HTML, ...):

The distance between (x_1, y_1) and (x_2, y_2) Is $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$.

Docu Examples

```
/    class Test
/    {
/      Public:
/        const char *member(char,int) throw(std::out_of_range);
/    };

/    const char *Test::member(char c,int n) throw(std::
                                    out_of_range) {}
/    /*! \class Test
/      * \brief Test class.
/      *
/      * Details about Test.
/      */

/    /*! \fn const char *Test::member(char c,int n)
/      *  \brief A member function.
/      *  \param c a character.
/      *  \param n an integer.
/      *  \exception std::out_of_range param is out of range.
/      *  \return a character pointer.
/      */
```

Docu Examples

/ **Last example (-:**

```
/*! \class Tramp
     \brief The TRAMP class.

     Detailed information here should be added by
         -# \author Application team
         -# \author GSE team
  */
```

/ **Versus the usual way without (Note: there is an error in this block)**

```
/**
  *   The TRAMP class. Maybe the <B>Doc</B> team or
      any other is even better <BLINK>:-)</BLINK>.
  */

  class Tramp
    { ...
```

Bibliography

1. **Bernd Bruegge / Allen Dutoit:** *Object-Oriented Software Engineering*, **Prentice Hall**

2. *jakarta.apache.org/ant/external.html*

3. *=> www.doxygen.org <=*