使用 doxygen 生成C 程序文档

[一 doxygen 的主要功能和特点： 1](#_Toc223488481)

[二 适用的场景 2](#_Toc223488482)

[三 安装和使用： 2](#_Toc223488483)

[1 安装步骤： 2](#_Toc223488484)

[2.doxygen使用步骤 2](#_Toc223488485)

[3. doxygen配置文件 2](#_Toc223488486)

[4 Doxygen 注释的格式： 4](#_Toc223488487)

[5 doxygen 注释命令 5](#_Toc223488488)

[6 doxygen 的其他工具： 5](#_Toc223488489)

[图表 5](#_Toc223488490)

[列表 6](#_Toc223488491)

[四 效果示例： 7](#_Toc223488492)

[五 参考文献： 7](#_Toc223488493)

# 一 doxygen 的主要功能和特点：

Doxygen是一种开源跨平台的，以类似JavaDoc风格描述的文件系统，完全支持C、C++、Java、Objective-C和IDL语言，部分支持PHP、C#。注释的语法和Qt-Doc、KDoc和JavaDoc兼容。Doxgen能从一套归档源文件开始，生成HTML格式的在线类浏览器，或离线的LATEX、RTF参考手册。对于未归档的源文件，也能通过设置Doxygen来提取代码结构。或借助自动生成的包含依赖图（include dependency graphs）、继承图（inheritance diagram）及协作图（collaboration diagram）来可视化文件之间的关系。Doxygen生成的帮助文件的格式能是CHM、RTF、PostScript、PDF、HTML和Unix man page等。

Doxygen在Linux上研发，但也能在其他的Unix平台下运行。而且，视窗系统 9x/NT平台下也有对应的可执行版本。

下面主要介绍在linux 环境下C语言程序的文档生成。

# 二 适用的场景

复杂代码的分析：对于复杂的第三方代码，如linux kernel,module等，可以用

Doxygen 生成代码浏览web页面，以及函数调用关系图，简化代码浏览和分析。

程序原型的讨论文档生成：进行概要设计时，往往想不清楚实现的细节。可以先写

主要程序设计函数流程，忽略函数实现，用doxygen 生成结果文档，作为设计讨论

稿。少做修改后，可以形成概要设计的主体部分。

# 三 安装和使用：

## 1 安装步骤：

1) 下载安装 doxygen <http://www.stack.nl/~dimitri/doxygen/download.html> (make install)

2) 下载安装 graphviz <http://www.graphviz.org/Download_linux_rhel.php> (yum list available 'graphviz\*' yum install 'graphviz\*')

2.doxygen使用步骤  
由于只是工具的使用，这里不介绍它的原理，直接从使用步骤开始。Doxygen的使用步骤非常简单。主要可以分为：  
1）第一次使用需要安装doxygen的程序  
2）生成doxygen配置文件：

Doxygen -g 配置文件名 （生成带注释的配置文件模版）

Doxygen -s -g 配置文件名 （生成不带注释的配置文件模版）

3）编码时，按照某种格式编写注释

4）执行 doxygen:

Doxygen 配置文件名

即可生成指定形式的文档

3. doxygen配置文件  
doxygen配置文件的格式是也是通常的unix下配置文件的格式：注释'#'开始；tag = value [,value2…]；对于多值的情况可以使用 tag += value [,value2…]。

对doxygen的配置文件的修改分为两类：一种就是输出选项，控制如何解释源代码、如何输出；一种就是项目相关的信息，比如项目名称、源代码目 录、输出文档目录等。对于第一种设置好后，通常所有项目可以共用一份配置，而后一种是每个项目必须设置的。下面选择重要的，有可能需要修改的选项进行解释 说明，其他选项在配置文件都有详细解释。

TAG 缺省值 含义  
PROJECT\_NAME 项目名称  
PROJECT\_NUMBER 可以理解为版本信息  
PROJECT\_BRIEF 项目简介  
PROJECT\_LOGO 项目logo  
OUTPUT\_DIRECTORY 输出文件到的目录，相对目录（doxygen运行目录）或者绝对目录  
INPUT 代码文件或者代码所在目录，使用空格分割  
FILE\_PATTERNS \*.c \*.cc \*.cxx \*.cpp \*.c++ \*.java \*.ii \*.ixx \*.ipp \*.i++ \*.inl \*.h \*.hh \*.hxx \*.hpp \*.h++ \*.idl \*.odl 指定INPUT的目录中特定文件，如：\*.cpp \*.c \*.h  
BRIEF\_MEMBER\_DESC YES 在成员列表后显示成员简介  
REPEAT\_BRIEF YES 在详细介绍前显示简介  
CALL\_GRAPH YES 显示调用关系图  
CALLER\_GRAPH YES 被调用关系图  
REFERENCED\_BY\_RELATION YES 显示被调用的函数  
REFERENCES\_RELATION YES 显示调用的函数和变量  
SORT\_MEMBER\_DOCS NO 函数或者类成员按照字母表顺序排序  
RECURSIVE NO 是否递归INPUT中目录的子目录  
EXCLUDE 在INPUT目录中需要忽略的子目录  
EXCLUDE\_PATTERNS 明确指定的在INPUT目录中需要忽略的文件，如：FromOut\*.cpp  
OUTPUT\_LANGUAGE English 生成文档的语言，当前支持2、30种语言，国内用户可以设置为Chinese  
EXTRACT\_ALL NO 为NO，只解释有doxygen格式注释的代码；为YES，解析所有代码，即使没有注释。类的私有成员和所有的静态项由EXTRACT\_PRIVATE和 EXTRACT\_STATIC控制  
EXTRACT\_PRIVATE NO 是否解析类的私有成员  
EXTRACT\_STATIC NO 是否解析静态项  
EXTRACT\_LOCAL\_CLASSES YES 是否解析源文件（cpp文件）中定义的类  
SOURCE\_BROWSER NO 如果为YES，源代码文件会被包含在文档中，并产生交叉饮用索引  
INLINE\_SOURCES NO 如果为YES，函数和类的实现代码被包含在文档中  
ALPHABETICAL\_INDEX NO 生成一个字母序的列表，有很多类、结构等项时建议设为YES  
GENERATE\_HTML YES 是否生成HTML格式文档  
GENERATE\_HTMLHELP NO 是否生成压缩HTML格式文档（.chm）  
GENERATE\_LATEX YES 是否乘车latex格式的文档  
GENERATE\_RTF NO 是否生成RTF格式的文档  
GENERATE\_MAN NO 是否生成man格式文档  
GENERATE\_XML NO 是否生成XML格式文档

建立好的模板后，要根据需要修改tag

一个修改过的配置文件：<http://svn.erc.ict.ac.cn/svn/bwfs/bwfsprogs/trunk/bwfs_tune/tune.ctl>

## 4 Doxygen 注释的格式：

Doxygen规定了进行注释的一些格式，正确的注释才能使Doxygen生成文件。第一个代码条目，都有两种描述：简要描述和周详描述，两者都是可选的。简要描述只有一行，而周详描述则提供更长、更仔细的描述，Doxygen只允许有一个简要描述和周详描述。

在Doxygen中，一般只会处理和程式结构相关的注释，函数内部的注释通常不做处理。对于周详描述来说，有下面几种表示方式。

|  |
| --- |
| JavaDoc风格，中间的"\*"号可选。  /\*\*  \* 注释  \* 注释  \*/  Qt风格，中间的"\*"号可选。  /\*!  \* 注释  \*/  C++风格的变体，或最后一个"/"改为"！"也能。  /// 单行注释  /// 注释  ///  更加显著的表示。  ///////////////////////////////////////////  /// 注释  ///////////////////////////////////////////  简要描述亦有多种表示方式。  在上述注释块中使用@brief命令，周详注释在空行之后开始。  /\*\*@brief 简要描述  \* 继续  \*  \* 周详注释  \*/    JAVADOC\_AUTOBRIEF设置为YES后，在JavaDoc风格的注释中，第一个点号之前的内容被自动设置为简要描述。  对于多行C++变体，这个选项亦会起到相同的作用。  /\*\* 简要描述.周详描述  \* 注释  \*/  C++变体风格。  /// 简要描述  /\* 周详描述 \*/ |

## 5 doxygen 注释命令

@param [out/in] 参数描述

@return返回值描述

@brief 简要描述，简要描述以空行结束，出现空行后，下一行为为详细描

述。注意，生成的文档中，有些地方只列出简要描述。

@warning 警告描述

## 6 doxygen 的其他工具：

### 图表

Doxygen里有内置生成C++类层次图的功能。他使用贝尔实验室研发的graphviz 1.5中的工具"dot"来生成更高级的图表。使用这个工具时，要将设置选项HAVE\_DOT设为YES。

当GRAPHICAL\_HIERARCHY设置为YES时，将会绘制一个图像表示的类图结构。

当CLASS\_GRAPH设置为YES时，会为每个归档的类创建一张图表示其直接或间接的继承关系。

当INCLUDE\_GRAPH设置为YES时，会为每个归档文件创建一幅包含依赖图，此功能目前仅有HTML和RTF格式支持。

当COLLABORATION\_GRAPH设置为YES时，会为每个归档类或结构绘制基类继承关系图和使用关系图。

当CALL\_GRAPH设置为YES时，会为每个函数显示一幅直接或间接调用关系图。

当CALLER\_GRAPH设置为YES时，会为每个函数显示一幅直接或间接被调用关系图，即列出所有调用此函数的函数。

注意，如果要实现上述功能，需要将HAVE\_DOT 设置为YES.

### 列表

Doxygen有许多方法能创建项目列表。

|  |
| --- |
| 使用"-"在每行开始之前打头，使用"."能结束一个列表，开始新的段落。使用这种方法要严格对齐。  /\*\*  \* - 表项一  \* - 子项一  \* - 子项二  \* .  \* .  \*/  在文件块中使用HTML命令。这种方法不必严格对齐。  /\*!  \* <ul>  \* <li> 表项一  \* <ol>  \* <li> 子项一  \* <li> 子项二  \* </ol>  \* <li> 表项二  \* </ul>  \*/ |

# 四 效果示例：

见具体演示

# 五 参考文献：

更详细的信息可以参考doxygen 官方网址：<http://www.stack.nl/~dimitri/doxygen/>

或 <http://www.doxygen.org/> 下载doxgen manual, pdf 格式或chm 格式。