文档编号：

fmt使用说明

fmt Instructions

|  |  |  |  |
| --- | --- | --- | --- |
| **文档状态** | [√]初稿  [ ] 评审通过  [ ] 修改  [ ] 发布  [ ] 作废 | **密级** |  |
| **当前版本** | V0.1 |

中国外汇交易中心

全国银行间同业拆借中心

变更履历

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **版本号** | **变更时间** | **修改人** | **变更简要说明** | **变更**  **类别** |
| 1 | 0.1 |  |  | 初始化创建文档 | N |

**fmt使用说明**

1. **fmt配置**
   1. **安装fmt库到环境中**

下载fmt库：

|  |
| --- |
| git clone https://github.com/fmtlib/fmt.git |

编译fmt：

|  |
| --- |
| cmake . |

安装fmt：

|  |
| --- |
| make && make install |

默认生成静态库，改为动态链接库：

|  |
| --- |
| cmake -DBUILD\_SHARED\_LIBS=TRUE .. |

默认安装路径为系统库，修改安装路径：

|  |
| --- |
| cmake -DCMAKE\_INSTALL\_PREFIX=<your path> .. |

使用时编译选项和依赖：

|  |
| --- |
| -std=c++11 -lfmt |

其他设置：

需要指定链接库和头文件的位置，将make install生成的静态库或动态库文件链接到工程中。

注意事项：

fmt库中使用的部分头文件名和系统库中头文件名，可以将所有头文件名增加fmt前缀区分系统库中头文件。修改了头文件名的版本获取地址：

|  |
| --- |
| https://github.com/MrUnbreakable/mylinuxgit/tree/master/fmtlib |

* 1. **CMAKE使用fmt库**

将fmt下载到项目中并在Cmake中添加：

|  |
| --- |
| link\_directories(<your path to libfmt>)  target\_link\_libraries( <your-target> fmt)//将整个库都包含其中  target\_link\_libraries( <your-target> fmt-header-only)//只包含头文件版本 |

官方API文档链接：

|  |
| --- |
| https://fmt.dev/dev/api.html |

1. **fmt使用**
   1. **头文件包含**

1. 基本包含：

|  |
| --- |
| #include "fmt/format.h" |

2. 其他包含（未修改头文件名的版本）：

|  |
| --- |
| #include "fmt/printf.h"  #include "fmt/color.h"  #include "fmt/chrono.h"  … |

备注：format.h文件必须包含在文件中，而其他包含文件根据是否使用具体功能而定。如：使用fmt::printf 需要包含printf.h。

* 1. **使用示例**

1. 最简单用法，类似于python：

|  |
| --- |
| fmt::print("Hello, {}!", "world");  fmt::print(stderr, "Don't {}!", "panic"); |

2. 多参数用法：

|  |
| --- |
| fmt::print("Elapsed time: {s:.2f} seconds", fmt::arg("s", 1.23)); |

3. 类似std中函数用法：

|  |
| --- |
| fmt::printf("Hello, %s!", "world");  std::string message = fmt::sprintf("The answer is %d", 42); |

4. 简单的字符串format方法：

|  |
| --- |
| std::string s = fmt::format("I'd rather be {1} than {0}.", "right", "happy"); |

5. 另一种format方法：

|  |
| --- |
| fmt::memory\_buffer buf;  fmt::format\_to(buf, "{}", 42);  fmt::format\_to(buf, "{:x}", 42); // 十六进制表示  std::cout << buf.data(); |

6. 日期格式化：

|  |
| --- |
| std::time\_t t = std::time(nullptr);  fmt::print("The date is {:%Y-%m-%d}.\n", \*std::localtime(&t)); |

7. stl容器中使用：

|  |
| --- |
| std::vector<int> v = {1, 2, 3};  fmt::print("{}\n", fmt::join(v, ", ")); |