ZSUMMER说明文档

目录

[ZSUMMER说明文档 1](#_Toc351324041)

[ZSUMMER整体架构 2](#_Toc351324042)

[架构图 2](#_Toc351324043)

[架构说明 2](#_Toc351324044)

[简介 3](#_Toc351324045)

[LOG4Z架构 4](#_Toc351324046)

[架构图 4](#_Toc351324047)

[简介 4](#_Toc351324048)

[日志记录器说明 5](#_Toc351324049)

[日志输出文件路径与文件名命名规则 5](#_Toc351324050)

[示例程序 6](#_Toc351324051)

# ZSUMMER

## introduction:

ZSUMMER是一款C++ 跨平台的 高性能的 轻量级的网络底层库, 支持TCP, UDP, 拥有完善的定时器机制与日志系统.  
ZSUMMER is a cross-platform C++ high performance lightweight network library.

## feature:

MIT开源 代码可以任意使用在任何场合.  
MIT source code can be used in any occasions.

纯原生 不依赖任何第三方库, C++98标准.  
does not rely on any third party libraries

所有代码全手工打造, 代码简洁清爽 每一句都经过仔细斟酌, 结构清晰顺畅, 命名规范易读, 性能稳定强大 .  
All of the code is all hand-made, code simple and refreshing every word carefully, structure is clear and smooth, easy name specification, performance is stable and powerful.

使用EPOLL/IOCP纯手工实现,采用优雅简洁的设计模式, 跨LINUX/WINDOWS 32/64平台而几乎不因此损耗任何性能.  
Realize the use of EPOLL/IOCP manual, use design pattern elegant, across the LINUX/WINDOWS 32/64 platform and almost no loss of any property.

上层接口采用规范的proactor一致的接口设计, 轻松应对绝大多数服务端的使用场景. 包括高并发高吞吐要求的前端 以及后端DB代理服务 逻辑业务服务等等.  
The upper interface with interface design specification of Proactor consistent, easy to deal with the vast majority of server usage scenarios.  
High throughput requirements including high concurrent front and rear DB agent service logic service and so on.

每个IOSERVER都可以无限制挂靠的connecter角色, accepter角色与clienter角色 充分发挥多核性能.  
Each IOSERVER can limit the link the role of connecter, accepter and clienter role into full play the role of multi-core performance.

完全异步的接口设计 排除业务逻辑中可能存在的底层重入问题  
Interface design fully asynchronous rule out the possibility of the underlying problem of re-entry business logic.

## 架构图

UTILITY

THREAD4Z

MER

LOG4Z

PROTOCOL4Z

ZSUMMER

EPOLL

IOCP

TCP\_ACCEPTER

TCP\_SOCKET

UDP\_SOCKET

TIMMER

## 架构说明

utility为跨平台的通用小部件封装.

thread4z为跨平台的线程封装 包含多线程, 互斥锁 信号量 原子操作 等封装.

LOG4Z 为日志记录库, 已经独立开源, 支持多日志分流 多线程 多级过滤 多种颜色显示, 支持文件配置与配置热加载.

PROTOCOL4Z 协议流封装.

ZSUMMER为EPOLL/IOCP实现的网络核心, 上层接口采用规范的proactor模式封装.

## 压测数据 stress report

WINDOWS: 60K并发 吞吐量120M 服务端每秒120,000次的消息包处理与收发 CPU总占用小于1%.  
WINDOWS: 60K concurrent, throughput of 120M, server 120000 times per second, the packet processing and transceiver CPU total occupancy of less than 1%.

LINUX, 因虚拟机性能问题: 40K并发 吞吐量80M 服务端每秒80,000次的消息包处理与收发, CPU总占用小于30%.  
LINUX, for virtual machine performance data than windows almost: 40K concurrent throughput of 80M server 80000 times per second, the packet processing and transceiver, CPU total occupancy of less than 30%.

详细见报告页:[stress-report](https://github.com/zsummer/zsummer/wiki/stress-report)  
For details see the report page: [stress-report](https://github.com/zsummer/zsummer/wiki/stress-report)

## build server & client

in linux:  
cd zsummer/example  
cmake .  
make

in windows:  
cd zsummer/example  
use vs2010 open zsummer.sln