《二次元资源网站》关键技术文档

**版本：1.0**

**时间：2024/12/5**

**作者：林昊国、陈兴艺、刘瑾瑜**

# 一、概述

### 1. 文档目的

本文档旨在为有兴趣了解本系统的技术爱好者解释用到的相应技术。希望能和其他大佬多沟通多交流，不断完善这个系统。

### 2. 目标受众

* **研究者与学生**：对http、前后端交互、数据库的使用感兴趣的学习者。

### 3. 主要内容

本文将主要介绍本系统用到的或将会用到的关键技术，涉及的方面包括但不限于：

1. 负载均衡
2. 网页界面
3. 数据库

# 正文

### 前端三件套HTML、CSS和JavaScript

HTML：负责网页的架构；

CSS：负责网页的样式，美化；

JavaScript（JS）：负责网页的行为；

主要用vscode进行开发

### oatpp

#### **核心特点：**

* **异步编程模型**：基于协程（coroutines）实现的轻量级线程，使得开发者可以编写非阻塞代码而无需显式管理线程或回调函数。
* **高性能**：通过优化内存管理和I/O操作，确保了低延迟和高吞吐量。
* **面向对象设计**：利用C++的类和继承机制来组织代码，提高了代码的可读性和复用性。
* **模块化架构**：提供了丰富的组件库，如HTTP服务器/客户端、数据库连接池、JSON解析器等，并且支持自定义扩展。
* **跨平台兼容**：可以在Linux, macOS, Windows等多个操作系统上运行。
* **内置工具链**：包括日志记录、配置管理、健康检查等功能，方便快速搭建生产环境。

Oat++为C++开发者提供了一个强大的异步编程框架，极大地简化了复杂网络应用的开发过程。凭借其良好的性能表现、现代化的设计理念以及全面的功能集，oatpp已经成为许多高性能服务器端应用的理想选择。

### mysql

#### **主要特点：**

**1. 开源与免费**

MySQL 是一个开源软件，这意味着任何人都可以自由使用、修改和分发它。尽管有商业版本提供额外的支持和服务，但社区版对于大多数个人开发者和中小企业来说已经足够强大且免费。

**2. 跨平台支持**

MySQL 可以运行在多种操作系统上，包括但不限于 Linux、Windows、macOS 和各种 Unix 系统。这使得它非常适合多平台环境下的应用程序开发。

**3. 高性能**

MySQL 在处理大量并发连接方面表现出色，能够快速执行读写操作。它的优化器可以帮助选择最有效的查询路径，从而提高性能。

**4. 易于使用**

MySQL 提供了简单易用的命令行工具和图形用户界面（GUI），使数据库管理变得直观友好。同时，丰富的文档和支持社区也方便新手学习和解决问题。

**5. 可靠性和稳定性**

经过多年的实践检验，MySQL 拥有很高的稳定性和可靠性，特别是在互联网应用中广泛应用。它提供了事务支持（ACID特性）、备份恢复机制等功能来保证数据完整性。

**6. 可扩展性**

MySQL 支持分区表、集群解决方案（如 NDB Cluster）以及主从复制等高级特性，以满足不同规模业务的需求。

**7. 安全措施**

内置的安全特性如用户权限管理、SSL加密传输等确保了数据库的安全性。此外，还可以通过插件或外部工具进一步增强安全性。

### redis

#### **核心特点:**

**1. 高性能**

Redis 是完全基于内存的操作，因此其读写速度非常快，每秒可以处理成千上万次请求。

使用非阻塞 I/O 模型，确保了高并发下的性能稳定。

**2. 持久化支持**

提供两种持久化方式：RDB 快照（Snapshotting）和 AOF 日志（Append Only File）。RDB 在指定的时间间隔内创建数据集的时间点快照；AOF 则记录服务器接收到的所有写操作命令，重启时会重放这些命令以恢复原始数据。

**3. 数据结构丰富**

支持多种数据类型，包括但不限于：字符串（String）：最基本的数据类型，可以是简单的键值对形式。

哈希（Hash）：用于存储对象或实体的信息。

列表（List）：实现队列和栈的功能。

集合（Set）：无序且元素唯一的集合。

有序集合（Sorted Set）：带分数排序的集合，适用于排行榜等功能。

位图（Bitmaps） 和 HyperLogLogs：用于统计分析场景。

**4. 发布/订阅模式**

内置 Pub/Sub 消息传递机制，允许客户端订阅特定频道的消息并接收推送通知。

**5. 事务支持**

虽然 Redis 不是严格意义上的 ACID 数据库，但它提供了一个简单的事务模型——MULTI/EXEC 命令组合，可以在一次执行中完成多个命令的原子性操作。

**6. Lua 脚本**

支持使用 Lua 编写的脚本直接在 Redis 服务器端运行，这有助于减少网络往返次数并提高效率。

**7. 主从复制与集群**

支持主从架构，用于数据冗余和读写分离；还提供了官方的 Redis 集群解决方案，以实现水平扩展和服务可用性。

**8. 易于部署和管理**

安装配置简单，文档详尽，社区活跃，插件丰富，便于集成到现有应用环境中。

### FastDFS

#### **核心特点:**

**1. 架构设计**

Tracker Server：跟踪服务器，负责管理所有的 Storage Server，协调客户端请求。它不直接保存文件，而是维护了所有 Storage Server 的元数据信息。

Storage Server：存储服务器，实际存放上传的文件。支持集群部署，每个集群可以包含多个组（group），每个组内又有多台 Storage Server。文件会根据策略被分配到不同的组中。

Group机制：FastDFS 支持多组（group）配置，每组之间相互独立，可以分布在不同的物理位置上。这有助于实现负载均衡和容灾备份。

**2. 高可用性**

支持主从复制（Master-Slave Replication），即一个 Group 内的 Storage Server 可以设置为主节点或从节点。当主节点发生故障时，可以从节点接管服务，保证系统的持续运行。

提供自动故障转移功能，如果某个 Storage Server 出现问题，Tracker Server 会自动将流量导向其他健康的 Storage Server。

**3. 扩展性强**

可以很容易地通过增加新的 Storage Server 或创建新的 Group 来横向扩展存储容量。

支持动态添加/删除节点，不影响现有业务的正常运作。

**4. 高效读写性能**

文件上传后会生成全局唯一的文件 ID，这个 ID 包含了文件的位置信息（如 Group 和 Storage Server），使得后续访问变得非常快速。

对于热点文件（频繁访问的文件），可以通过缓存机制进一步提升读取速度。

**5. 简单易用**

API 设计简洁明了，提供了 C/C++、Java 等多种语言的客户端库，方便集成到各种应用程序中。

配置文件参数少，易于理解和调整。

**6. 安全性**

支持基于 Token 的鉴权机制，确保只有合法用户才能进行文件操作。

数据传输过程中可启用 SSL/TLS 加密，保护文件内容的安全。

### Nginx

#### **核心特点:**

**1. 高并发性能**

Nginx 使用异步非阻塞事件驱动架构，能够高效地处理大量并发连接。与传统的基于进程或线程模型的Web服务器相比，Nginx 在内存占用和CPU使用率方面具有显著优势。

**2. 反向代理与负载均衡**

可以作为反向代理服务器，将客户端请求转发给后端应用服务器（如FastCGI、uWSGI、Java Servlet容器等），并根据配置实现简单的负载均衡策略，如轮询、IP哈希等。

支持HTTP/2协议，进一步提高了页面加载速度和服务响应时间。

**3. 静态文件服务**

Nginx 非常擅长提供静态内容（HTML、CSS、JavaScript、图片等）的服务，通过缓存机制可以极大地减少磁盘I/O操作，提升访问效率。

**4. 模块化设计**

提供了多种内置模块和第三方扩展模块，允许用户根据需要灵活定制功能。例如，可以通过安装 ngx\_http\_rewrite\_module 模块来实现URL重写；利用 ngx\_http\_ssl\_module 支持HTTPS加密传输。

**5. 安全特性**

内置了基本的身份验证机制（如HTTP Basic Auth）、XSS防护、CSRF防御等功能。

支持SSL/TLS证书管理，确保数据传输的安全性；还提供了对HTTP严格传输安全（HSTS）头的支持。

**6. 易于配置**

配置文件采用简洁明了的语法结构，易于理解和维护。官方文档详尽，社区活跃，遇到问题时很容易找到解决方案。

**7. 动态模块加载**

自1.9.11版本起，Nginx 引入了动态模块加载功能，可以在不重新编译的情况下添加或移除某些模块，增强了灵活性和可维护性。

**8. 集成其他工具**

可以与各种监控工具（Prometheus, Grafana）集成，方便进行性能监测和故障排查。

结合日志分析工具（ELK Stack, Graylog），帮助管理员更好地理解流量模式和优化系统表现。