

# K8s集群资源使用率统计分析工具 - 开发文档

---

## 文档信息

- \*\*版本\*\*: 1.1.0
  - \*\*创建日期\*\*: 2026-02-25
  - \*\*更新日期\*\*: 2026-02-26
  - \*\*技术栈\*\*: Python 3.10 + FastAPI + openpyxl
  - \*\*作者\*\*: 骆玉宝
  - \*\*邮箱\*\*: ybluo@alauda.io
  - \*\*GitHub\*\*: <https://github.com/Linus-cat/k8s-resource-analyzer>
  - \*\*使用AI工具\*\*: OpenCode
- 

## 1. 项目概述

### 1.1 背景与目标

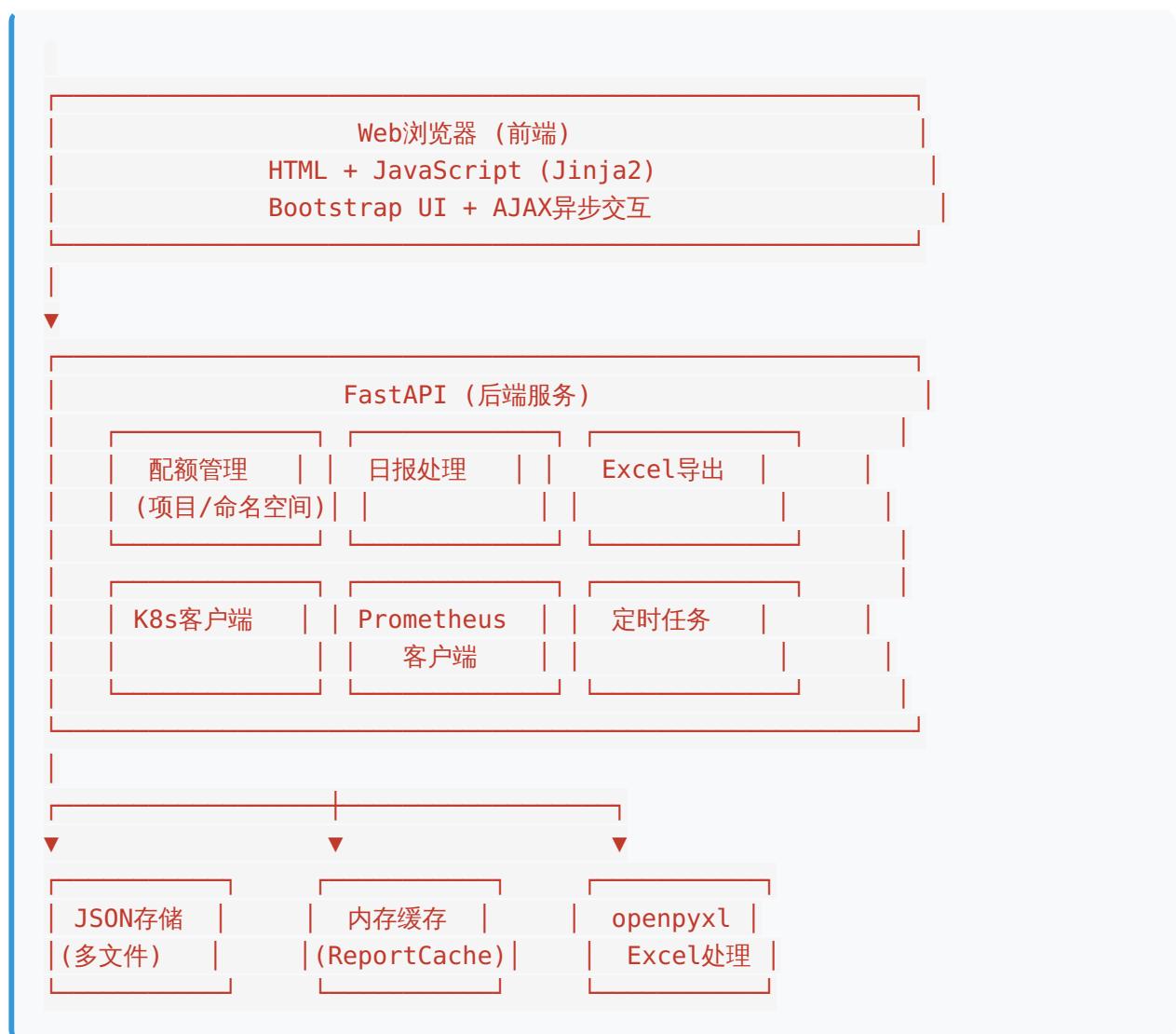
随着Kubernetes集群规模的不断扩大，对集群资源使用情况进行有效监控和分析变得尤为重要。本工具旨在实现多集群、多维度的资源使用数据自动化统计与分析，支撑项目级资源配额使用率的日常监控与管理。

### 1.2 核心功能

- 项目配额管理**: 项目级CPU、内存配额配置，支持手动导入和K8s同步
- 命名空间配额管理**: K8s命名空间级资源配额 (CPU、内存、Pod、存储)
- 多K8s集群支持**: 可配置多个K8s集群，统一管理
- 多Prometheus支持**: 可配置多个Prometheus实例
- 准确数据同步**: 基于副本数去重的精确同步，适合滚动更新场景
- 日报数据解析**: 从文件读取各命名空间的CPU、内存使用数据
- 使用率计算**: 按项目汇总资源使用量，结合配额计算使用率
- Excel导出**: 生成标准化报告，支持多日期导出
- 定时任务**: 自动同步K8s配额和Prometheus数据

## 2. 系统架构

### 2.1 技术架构



### 2.2 项目结构



```
|   └── quota_manager.py      # 配额管理 (CRUD+Excel导入)
|   └── calculator.py        # 使用率计算逻辑
|   └── exporter.py          # Excel导出功能
|   └── k8s_client.py         # K8s API客户端
|   └── prometheus_client.py # Prometheus客户端
|   └── scheduler.py         # 定时任务调度器
├── templates/
|   └── index.html           # 前端页面 (多Tab界面)
└── data/
    ├── quotas.json          # 项目配额
    ├── k8s_configs.json      # K8s集群配置
    ├── prometheus_configs.json # Prometheus配置
    ├── namespace_quotas.json # 命名空间配额
    └── scheduler_config.json # 定时任务配置
└── uploads/                # 上传文件临时目录
└── requirements.txt         # Python依赖
└── README.md                # 项目说明
```

---

## 3. 安装与运行

### 3.1 环境要求

- Python 3.10+
- pip 包管理器

### 3.2 安装步骤

```
# 1. 进入项目目录
cd k8s-resource-analyzer

# 2. 安装依赖
pip install -r requirements.txt

# 3. 启动服务
python -m app.main
# 或
uvicorn app.main:app --host 0.0.0.0 --port 8000
```

### 3.3 访问方式

- 本地访问: <http://localhost:8000>
- 局域网访问: <http://<服务器IP>:8000>

---

## 4. 功能说明

### 4.1 菜单功能总览

菜单	功能说明
项目配额管理	管理项目级配额，支持K8s同步、手动导入Excel、批量操作
命名空间配额	管理命名空间级配额，显示CPU/内存/Pod/存储等资源限制和已使用量
日报上传	上传日报文件，系统解析并计算项目资源使用率
报表查看	查看已上传的日报数据，按日期筛选
导出	选择多个日期，导出Excel报告
定时任务	配置自动同步任务的时间和间隔
K8s接入	配置K8s集群连接信息，支持批量导入多个集群
Prometheus接入	配置Prometheus连接，支持批量导入多个实例

### 4.2 项目配额管理

#### #### 4.2.1 K8s同步

- 选择已接入的K8s集群（支持全选）
- 点击"同步项目配额"从K8s获取ProjectQuota数据
- 支持批量同步多个集群

#### #### 4.2.2 手动导入

- 下载模板，格式：云序号、项目名称、CPU配额、内存配额
- 支持增量导入

## 4.3 命名空间配额

- 显示所有命名空间的资源限制和已使用量
- CPU、内存、Pod数、存储使用量及使用率
- 支持按集群筛选

## 4.4 K8s接入

- 添加K8s集群配置（集群名称、Kubeconfig）
- 支持ProjectQuota或Namespace ResourceQuota
- 批量导入多个集群配置
- 下载配置模板

## 4.5 Prometheus接入

- 添加Prometheus配置（名称、URL、关联集群）
- 支持精确同步模式（根据副本数去重）
- 批量导入多个Prometheus配置

## 4.6 定时任务

- 启用/禁用定时任务
- 配置命名空间配额同步时间
- 配置Prometheus同步时间
- 配置同步间隔

## 4.7 数据导出

- 选择日期（支持全选）
- 导出格式：主键/周期/CPU利用率/内存利用率
- 使用率保留10位小数

---

## 5. API接口文档

### 5.1 项目配额管理

方法	路径	说明
GET	/api/quotas	获取所有配额列表
POST	/api/quotas	添加单个配额
PUT	/api/quotas/{cloud_id}	修改配额
DELETE	/api/quotas/{cloud_id}	删除配额
POST	/api/quotas/import	Excel导入配额

### 5.2 K8s集群管理

方法	路径	说明
GET	/api/k8s/configs	获取K8s配置列表
POST	/api/k8s/configs	添加K8s配置
DELETE	/api/k8s/configs/{id}	删除K8s配置
POST	/api/k8s/sync	同步命名空间配额
POST	/api/k8s/sync-project-quota	同步项目配额
POST	/api/k8s/configs/import	批量导入K8s配置

### 5.3 命名空间配额

方法	路径	说明
GET	/api/k8s/quotas	获取命名空间配额列表
DELETE	/api/k8s/quotas/{cluster}/{namespace}	删除命名空间配额

POST	/api/k8s/quotas/import	批量导入命名空间配额
------	------------------------	------------

## 5.4 Prometheus管理

方法	路径	说明
GET	/api/prometheus/configs	获取Prometheus配置列表
POST	/api/prometheus/configs	添加Prometheus配置
DELETE	/api/prometheus/configs/{id}	删除Prometheus配置
POST	/api/prometheus/sync	同步Prometheus数据
POST	/api/prometheus/import	批量导入Prometheus配置

## 5.5 定时任务

方法	路径	说明
GET	/api/scheduler/config	获取定时任务配置
POST	/api/scheduler/config	更新定时任务配置

## 5.6 日报与导出

方法	路径	说明
POST	/api/reports/upload	上传日报文件
GET	/api/reports	获取已上传日期列表
GET	/api/reports/{date}	获取指定日期数据
GET	/api/reports/export	导出Excel报告

## 5.7 模板下载

方法	路径	说明

GET	/api/templates/quota	下载项目配额模板
GET	/api/templates/report	下载日报模板
GET	/api/templates/namespace-quota	下载命名空间配额模板
GET	/api/templates/k8s-config	下载K8s配置模板
GET	/api/templates/prometheus-config	下载Prometheus配置模板

---

## 6. 核心模块说明

---

### 6.1 k8s\_client.py - K8s API客户端

- 解析kubeconfig配置
- 调用Kubernetes API获取命名空间、资源配额
- 获取资源
- 获取ProjectQuota自定义Workload副本数（用于精确同步）

### 6.2 prometheus\_client.py - Prometheus客户端

- 查询CPU/内存使用量
- 精确同步模式：根据副本数去重处理滚动更新
- PromQL规则：
- CPU: `sum(max by (pod)(max\_over\_time(irate(container\_cpu\_usage\_seconds\_total[5m])[1d:])))`
- 内存: `sum(max by (pod)(max\_over\_time(container\_memory\_usage\_bytes[1d])))`

### 6.3 scheduler.py - 定时任务

- 定时同步K8s命名空间配额
- 定时同步Prometheus使用数据
- 可配置同步时间

---

## 7. 部署建议

### 7.1 开发环境

```
pip install -r requirements.txt  
python -m app.main
```

### 7.2 生产环境

```
# 使用Gunicorn替代uvicorn  
pip install gunicorn  
gunicorn -w 4 -k uvicorn.workers.UvicornWorker app.main:app --bind 0.0.0.0:8000
```

### 7.3 Docker部署

```
FROM python:3.10-slim  
  
WORKDIR /app  
  
COPY requirements.txt .  
RUN pip install --no-cache-dir -r requirements.txt  
  
COPY . .  
  
EXPOSE 8000  
  
CMD ["uvicorn", "app.main:app", "--host", "0.0.0.0", "--port", "8000"]
```

```
docker build -t k8s-resource-analyzer .  
docker run -d -p 8000:8000 -v ./data:/app/data k8s-resource-analyzer
```

## 8. 后续优化与功能添加计划

---

### 8.1 数据持久化

**优先级:** 高

**目标:** 从JSON文件存储改为SQLite/PostgreSQL数据库

**收益:**

- 支持更大数据量
- 提高并发性能
- 支持数据备份和迁移

### 8.2 用户认证与权限管理

**优先级:** 中

**目标:** 增加登录功能，支持多用户和角色权限

**功能:**

- 用户注册/登录（JWT认证）
- 角色：管理员、普通用户
- 管理员可管理所有配置
- 普通用户只能查看和导出

### 8.3 告警功能

**优先级:** 中

**目标:** 使用率超过阈值时发送通知

**通知方式:**

- 邮件通知
- 钉钉/企业微信机器人
- Slack Webhook

### 8.4 历史趋势与可视化

**优先级:** 中

**目标:** 展示历史使用率变化趋势

**功能:**

- 使用ECharts绘制趋势图
- 按周/月/年汇总
- 对比不同项目/集群的使用率

## 8.5 多租户支持

**优先级:** 低

**目标:** 支持多个独立租户使用

**功能:**

- 租户隔离
- 独立配额和配置
- 租户管理员

## 8.6 API开放

**优先级:** 低

**目标:** 提供RESTful API供其他系统集成

**功能:**

- 配额查询API
- 使用率查询API
- 导出API
- 集成文档 (Swagger)

## 8.7 监控与日志

**优先级:** 低

**目标:** 完善系统监控和日志记录

**功能:**

- 请求日志
- 错误追踪
- 性能监控 (Prometheus metrics)
- 健康检查接口

---

# 9. 附录

## 9.1 依赖清单

```
fastapi==0.109.0
uvicorn==0.27.0
openpyxl==3.1.2
```

```
python-multipart==0.0.6  
pydantic==2.5.3  
jinja2==3.1.3  
pyyaml==6.0.1
```

## 9.2 常见问题

### Q1: 导出时提示"Report not found"

A: 确保已上传日报文件，且该日期有匹配配额的项目。

### Q2: 上传文件解析失败

A: 检查文件格式是否符合要求，每行格式为：`项目名称;命名空间;CPU;内存(bytes)`

### Q3: 使用率显示为空

A: 确保配额中的项目名称与日报中的项目名称完全匹配。

### Q4: K8s同步失败

A: 检查kubeconfig是否正确，确保API Server可访问。

### Q5: Prometheus同步失败

A: 检查Prometheus URL是否可访问，确认查询语句正确。

---

**文档结束**