

研发赛组：智能发布系统

Z³Labs：李正乾，郑祥林，赵英杰

2025-10-30

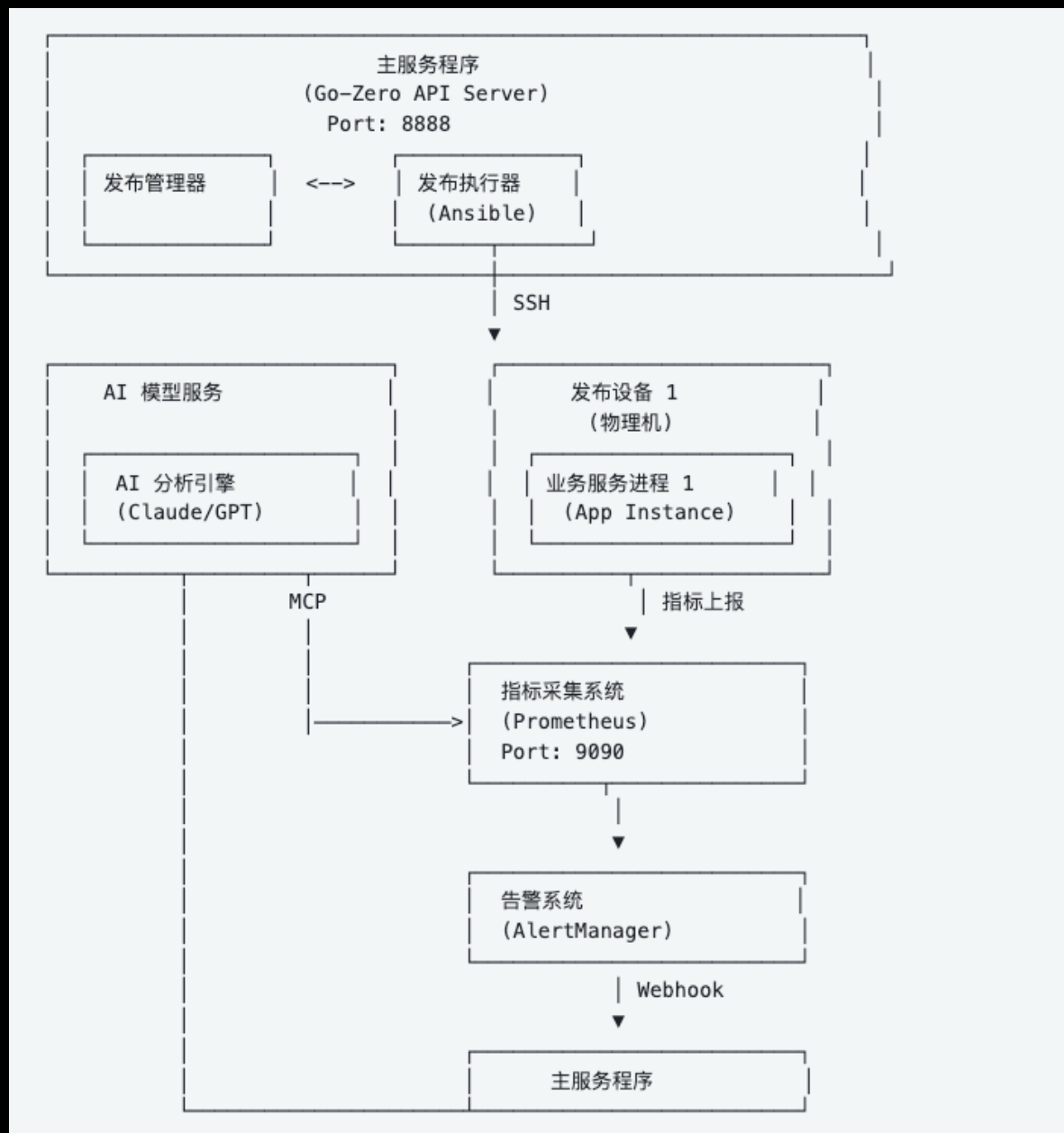
用户痛点

- 线上80%故障都是发布变更导致的异常
- 异常之后无法及时回滚，需要人为干预
- 告警之后没有自动的分析，容易手忙脚乱

用户故事

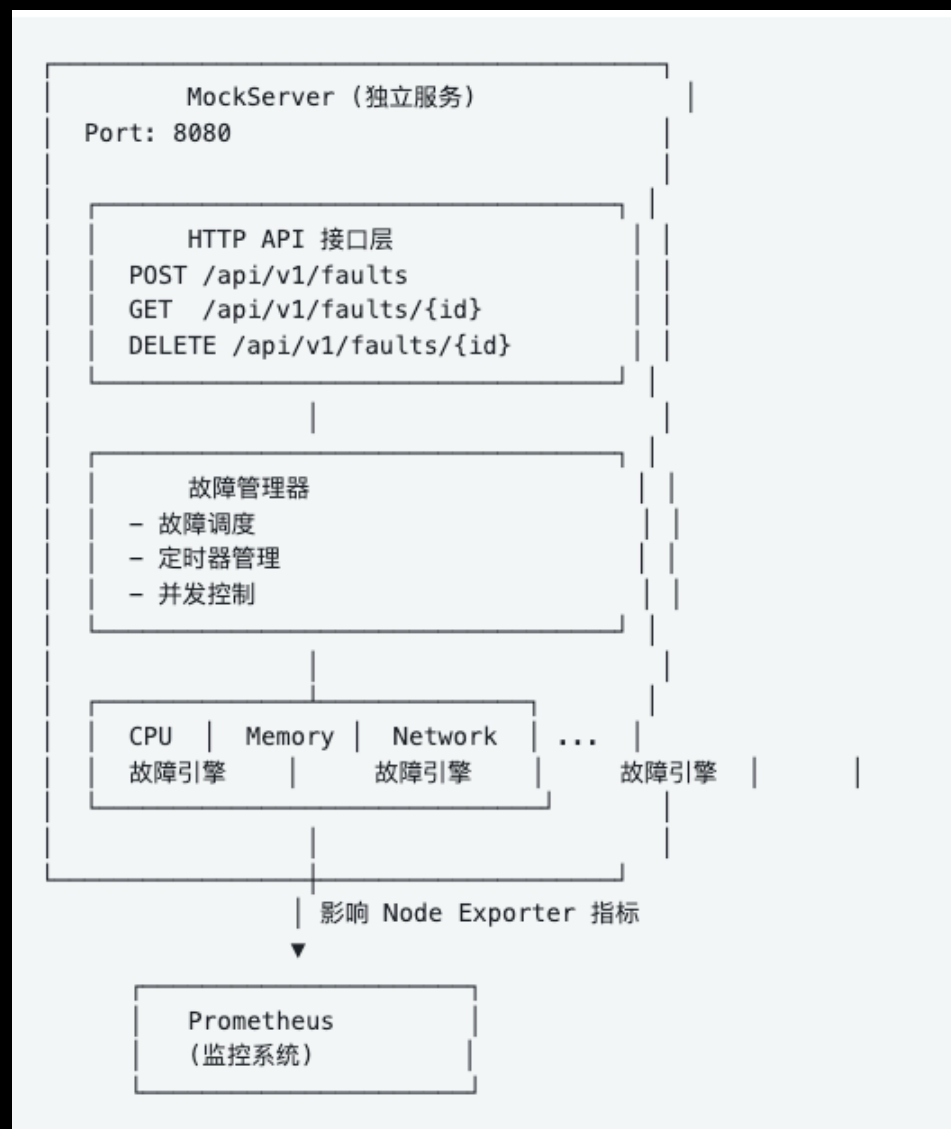
- us1: github 打 tag 触发打包上传到 kodo
- us2: 用户创建发布任务
- us3: 期间回滚指标异常，触发回滚策略
 - 自动回滚
 - 触发告警
 - 触发AI 自动分析

架构设计



mockServer 设计

- MockServer 是一个极简的故障注入服务，核心设计精髓：
- - curl 即可触发：无需复杂配置，一条命令模拟故障
- - 多类型并发：同时触发 CPU、内存、网络等多种故障
- - 持续时间控制：精确控制故障持续时间
- - 用于 AI 测试：验证 AI 诊断系统能否识别和分析故障



AI 智能分析

- 普罗米修斯mcp 服务器查询更多的指标信息，结合告警信息进行分析
- 从github拉取本次发布的tag中所有pr对应的代码变更，结合指标信息和代码变更进行分析
- 最终生成一份ai分析报告，包含ai认为异常的指标查询语句，以及根因分析，解决方案

关键 DB(Application) 定义

```
Application struct {
    Id                string        `bson:"_id"           json:"id,omitempty"  // mongo id
    Name              string        `bson:"name"          json:"name"`         // 应用名称
    Repo              string        `bson:"repo"          json:"repo"`         // 仓库地址
    DeploymentPlatform PlatformType `bson:"deploymentPlatform" json:"deployment_platform" // 部署平台
    DeployPath        string        `bson:"deployPath"    json:"deploy_path"`  // 部署路径
    ConfigPath        string        `bson:"configPath"    json:"config_path"`  // 配置文件路径
    StartCmd          string        `bson:"startCmd"      json:"start_cmd"`    // 启动命令
    StopCmd           string        `bson:"stopCmd"       json:"stop_cmd"`     // 停止命令
    CurrentVersion    string        `bson:"currentVersion" json:"currentVersion" // 当前版本
    PrevVersion       string        `bson:"prevVersion"   json:"prev_version"` // 上一个稳定版本
    MachineCount      int          `bson:"machineCount"  json:"machine_count"` // 机器总数量
    HealthCount       int          `bson:"healthCount"   json:"health_count"`  // 健康机器数量
    ErrorCount        int          `bson:"errorCount"    json:"error_count"`   // 异常机器数量
    AlertCount        int          `bson:"alertCount"    json:"alert_count"`   // 告警机器数量
    Machines          []Machine   `bson:"machines"      json:"machines"`      // 机器列表
    UpStreamAppIds    []string    `bson:"upStreamAppIds" json:"up_stream_ids"` // 上游应用
    DownstreamAppIds  []string    `bson:"downStreamAppIds" json:"down_stream_ids" // 下游服务
    RollbackPolicy    *RollbackPolicy `bson:"rollbackPolicy" json:"rollback_policy" // 回滚策略配置
    Pacer             PacerConfig  `bson:"pacer"         json:"pacer"`         // 发布节奏
    REDMetricsConfig  *REDMetrics  `bson:"redMetricsConfig" json:"red_metrics_config" // RED指标配置,在做基于 AI 的异常分析时可以使用这些指标

    CreatedTime time.Time `bson:"createdTime" json:"createdTime"` // 创建时间
    UpdatedTime time.Time `bson:"updatedAt"   json:"updatedAt"`   // 更新时间
}
```

关键 DB(Deployment) 定义

```
Deployment struct {
    Id            string      `bson:"_id,omitempty"  json:"id,omitempty"`
    AppName       string      `bson:"appName"        json:"app_name"  // 应用名称
    AppId         string      `bson:"appId"          json:"app_id"`
    Status        DeploymentStatus `bson:"status"        json:"status"    // 发布状态
    PackageVersion string    `bson:"packageVersion" json:"package_version" // 包版本
    GrayMachineId string    `bson:"grayMachineId"  json:"gray_machine_id" // 灰度设备ID
    Platform      PlatformType `bson:"platform"       json:"platform"    // 平台类型
    Package        PackageInfo `bson:"package"        json:"package"     // 包信息
    NodeDeployments []NodeDeployment `bson:"nodeDeployments" json:"node_deployments" // 发布机器列表
    CreatedTime    int64        `bson:"createdTime"    json:"createdTime"  // 创建时间戳
    UpdatedTime    int64        `bson:"updatedTime"    json:"updatedTime"  // 更新时间戳
}

PackageInfo struct {
    URL          string      `bson:"url"          json:"url"`
    MD5          string      `bson:"md5"          json:"md5"`
    Size         int64       `bson:"size"         json:"size"`
    CreatedAt    time.Time   `bson:"createdAt"    json:"created_at"`
}
```


后续规划

1. 服务上下游异常追踪
2. 服务拓扑
3. 灰度发布机制
4. 基于 codeagent 自动创建 hotfix

demo time

THANKS !