

部署到 Kubernetes



扫码试看/订阅

《.NET Core 开发实战》视频课程

3.1 部署：演示一个部署流程

Kubernetes 是什么

- Kubernetes 是一个用于自动部署、扩展和管理容器化应用程序的开源系统。

Kubernetes 的优势

- 自动化容器部署
- 资源管理与容器调度
- 服务注册发现与负载均衡
- 内置配置与密钥管理
- 丰富的社区组件
- 极强的可扩展性

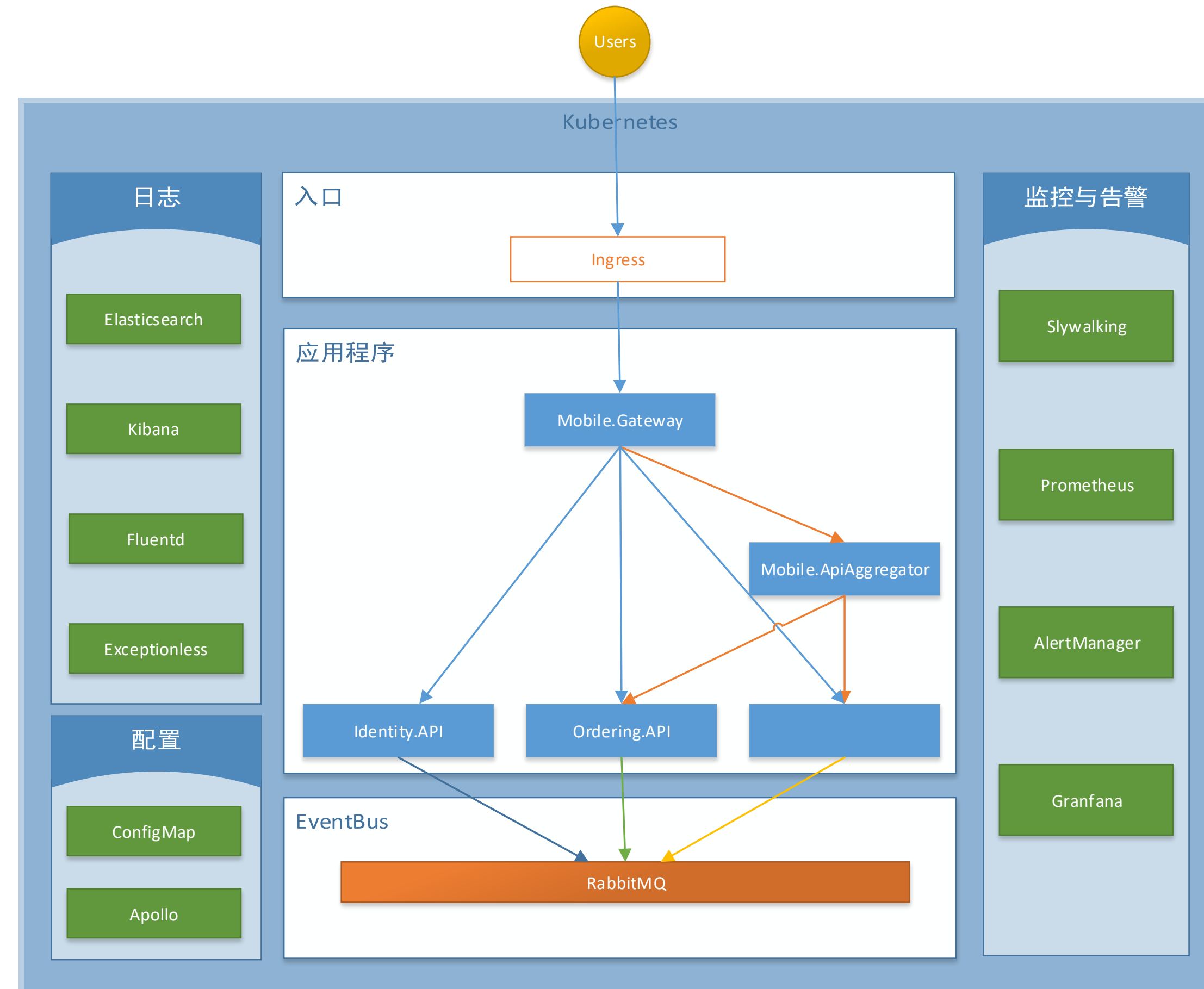
环境搭建

- 安装 Docker Desktop
- 安装 Kubernetes
- 安装 Helm
- 部署基础设施

应用部署

- 准备 Dockerfile
- 构建镜像
- 准备配置
- 部署应用

部署拓扑图



注意点

- 使用 NuGet.config 配置编译时的私有包仓库
- 时区设置

3.2 ConfigMap：实现基本配置方案

创建 ConfigMap 的方式

- 从文件夹创建
- 从文件创建
- 从键值对

使用 ConfigMap 的方式

- 映射为文件
- 映射为环境变量
- 映射为命令行参数

要点

- 版本化管理配置文件，以支持快速回滚
- 共享配置使用环境变量注入

3.3 配置：使用分布式配置中心方案版本化管理配置

什么时候选择配置中心

- 多项目组并行协作
- 运维开发分工职能明确
- 对风险控制有更高诉求
- 对线上配置热更新有诉求

Apollo 配置中心系统的能力

- 权限与审计
- 版本管理
- 热更新
- 原生支持 Java、.NET 客户端
- 目前项目仍很活跃

3.4 健康检查：与 Liveness、Readiness、Startup 探测集成实现高可用

LivenessProbe

- 判断服务是否存活
- 结束“非存活”状态的服务
- 根据重启策略决定是否重启服务

ReadinessProbe

- 判断服务是否 “就绪”
- “就绪” 状态的服务可以接收请求
- 非 “就绪” 状态的服务将会被从流量负载中摘除

StartupProbe

- 启动探针执行成功后，将不再执行，除非应用重启
- 当启动探针检测成功后，其它探针才开始工作
- 适合启动较慢的应用配置
- Kubernetes 1.16 以后的支持

3.5 健康检查：搭建全量健康检查探针和看板

AspNetCore.Diagnostics.HealthCheck 项目

- <https://github.com/Xabaril/AspNetCore.Diagnostics.HealthCheck>

组件包

- 探针检测端: `AspNetCore.HealthChecks.UI`
- 应用端输出: `AspNetCore.HealthChecks.UI.Client`
- 应用端检查项: `AspNetCore.HealthChecks.Xxx`

3.6 ForwardedHeaders: 确保服务在负载均衡下正常工作

场景

- 获取用户 IP 地址
- 获取用户请求的原始 URL

关键的 HTTP 头

- X-Forwarded-For
- X-Forwarded-Proto
- X-Forwarded-Host

要点

- 设置 PathBase
- 设置 ForwardedHeaders 中间件
- 设置受信任的网关或代理的 IP 段

3.7 安全：介绍强制 HTTPS 的两种方式

强制 HTTPS 的两种方式

- Ingress 强制 HTTPS
- 应用强制 HTTPS

Ingress 配置重定向

- `nginx.ingress.kubernetes.io/ssl-redirect: "true"`
- `nginx.ingress.kubernetes.io/force-ssl-redirect: "true"`

应用配置

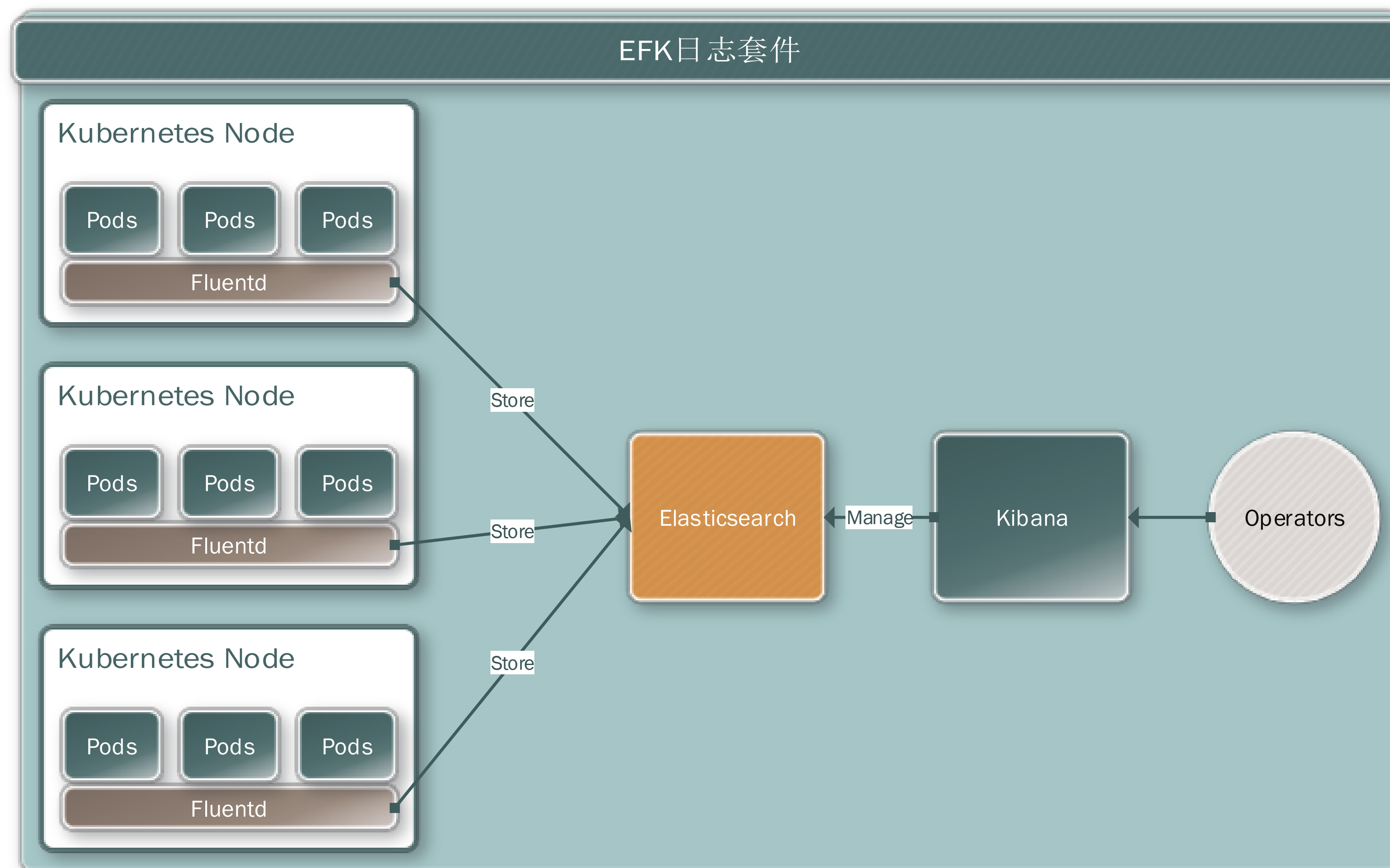
- 配置 https_port
- 配置 HttpsRedirectionOptions
- UseHttpsRedirection

3.8 日志：与 EFK 日志三件套集成

核心组件

- Elasticsearch（存储）
- Fluentd（收集器）
- Kibana（数据看板）

架构图

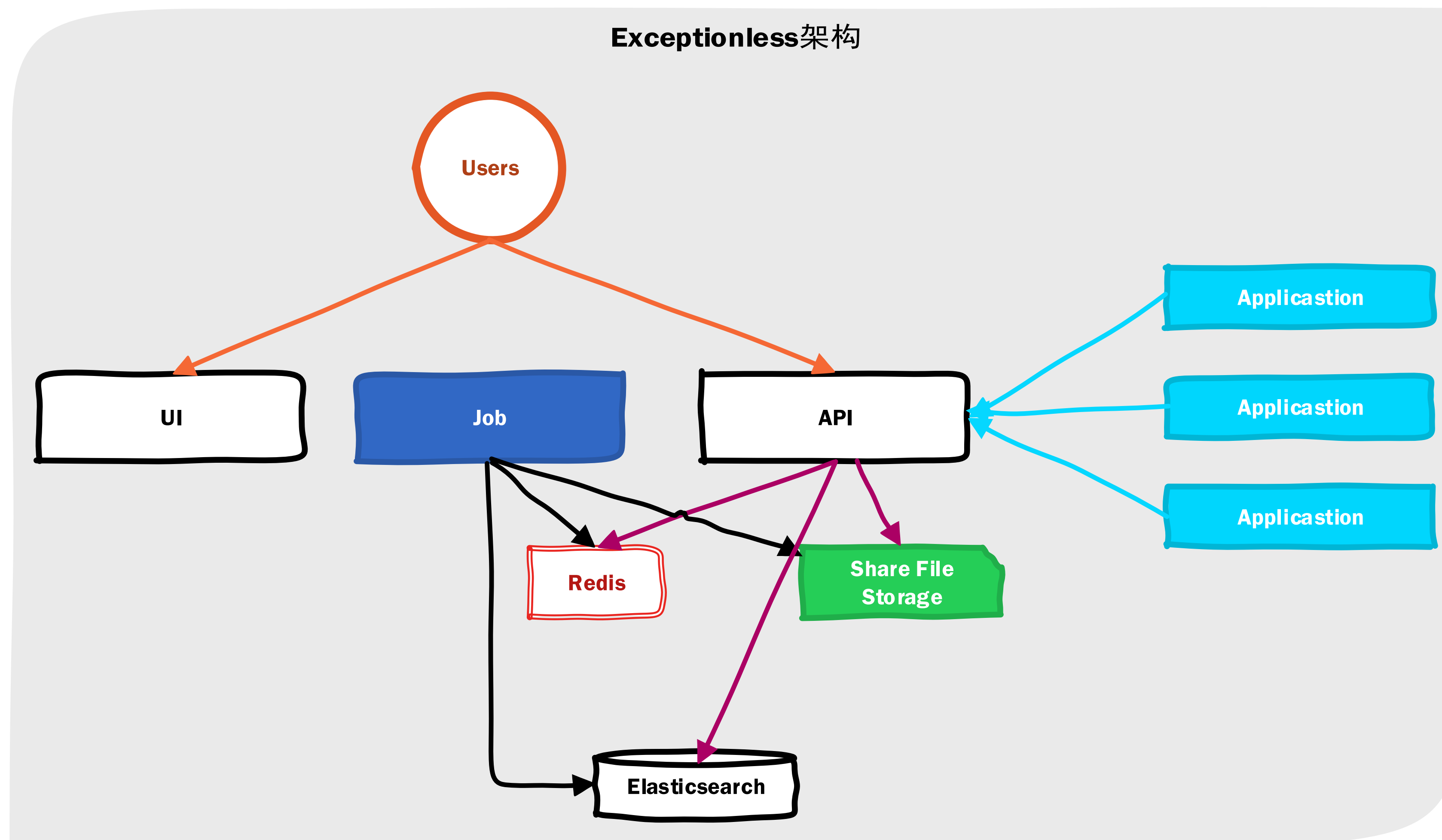


3.9 日志：Exceptionless 日志系统

Exceptionless 核心特性

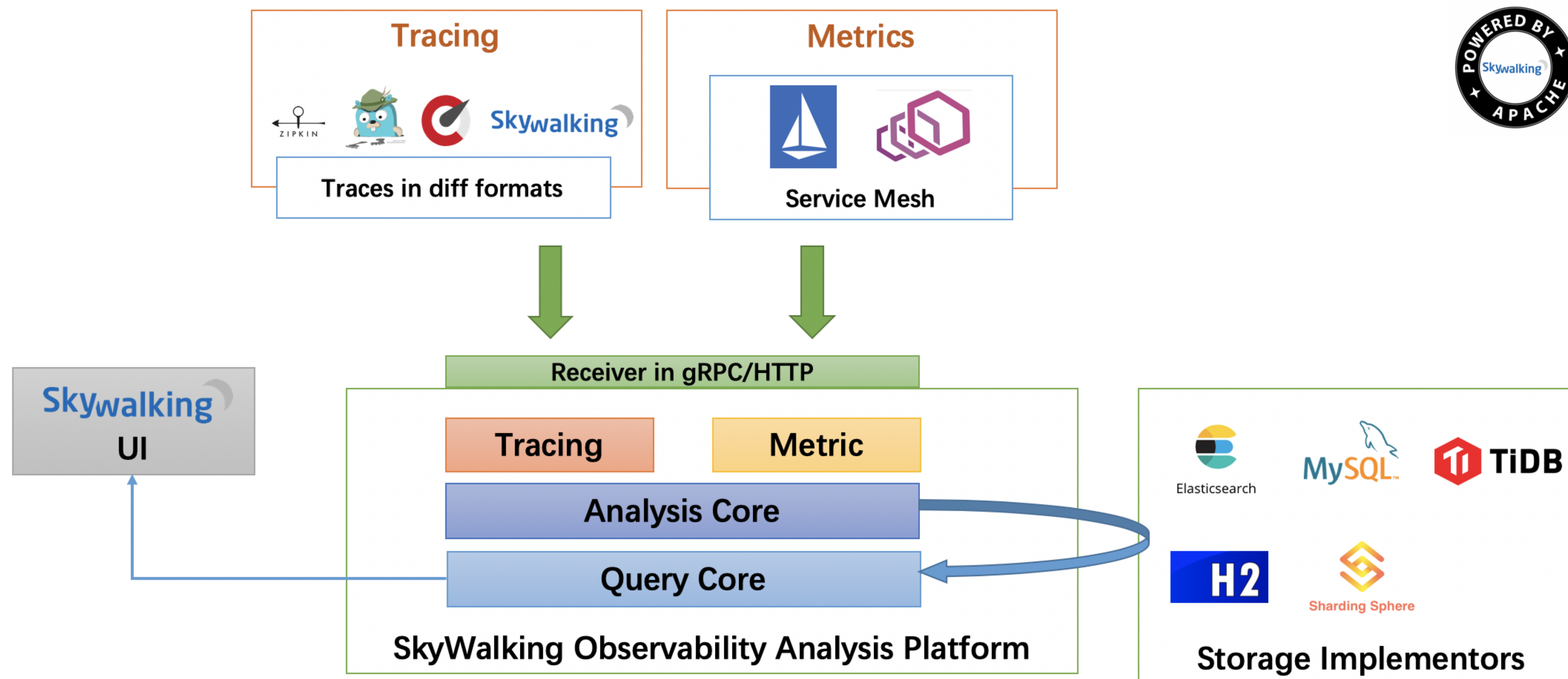
- 友好的界面
- 内置日志分类看板
- 按团队和项目管理应用
- 支持 Web hook 发送异常通知
- 基于 .NET Core 开源

架构图



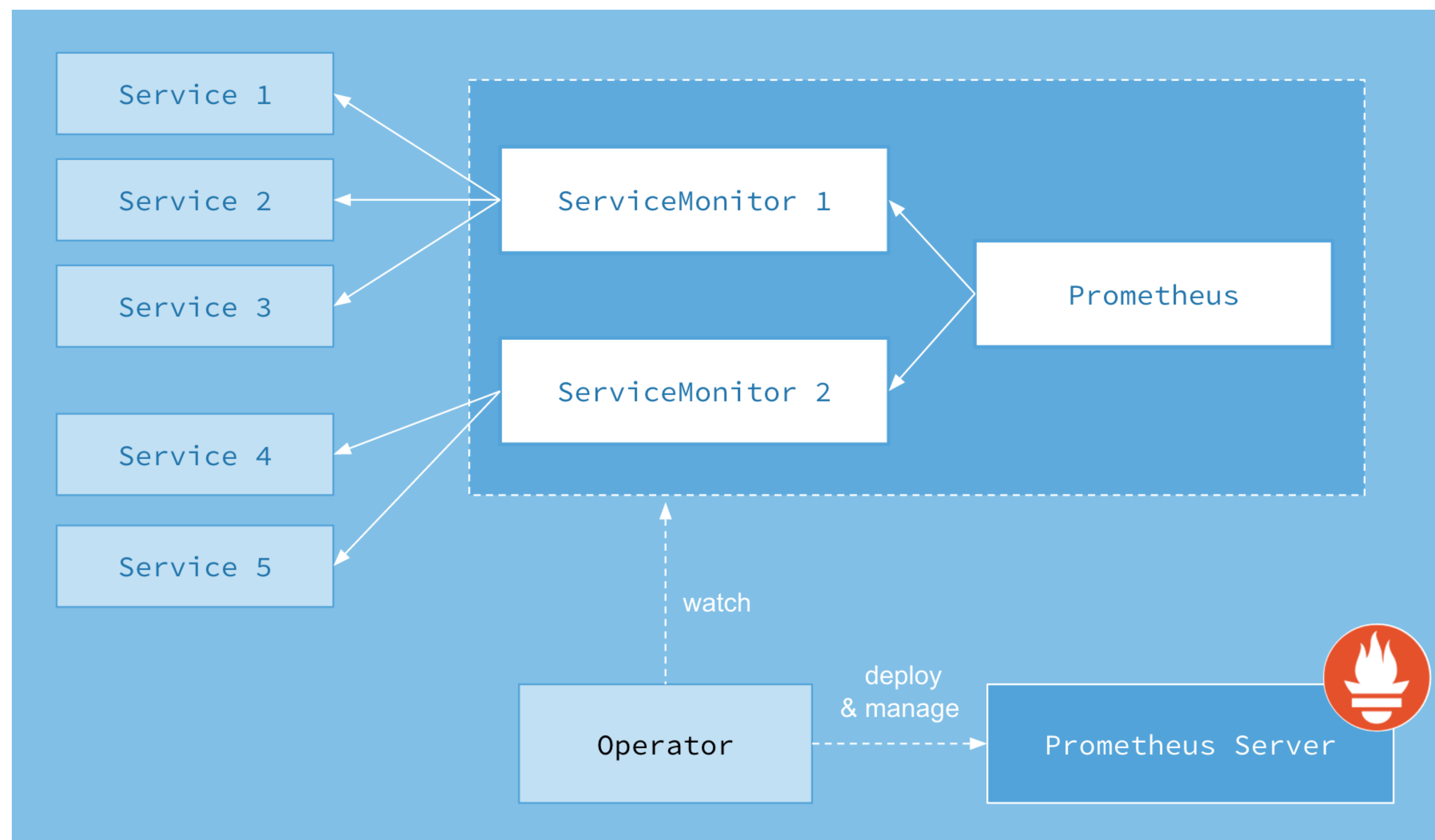
3.10 追踪：集成 SkyWalking.NET 实现追踪

拓扑图



3.11 监控与告警：Prometheus 与 AlertManager

prometheus-operator 架构



优势

- 自动化的安装
- 将配置资源化
- 灵活的扩展能力

3.12 监控与告警：用 Grafana 实现监控看板

Granfana 核心特性

- 支持各种主流数据源
- 丰富的图形组件
- 灵活的看板组织方式
- 灵活的权限支持
- 支持 OAuth 账户支持

3.13 prometheus-net: 自定义监控指标利器

项目地址

- <https://github.com/prometheus-net/prometheus-net>

组件包

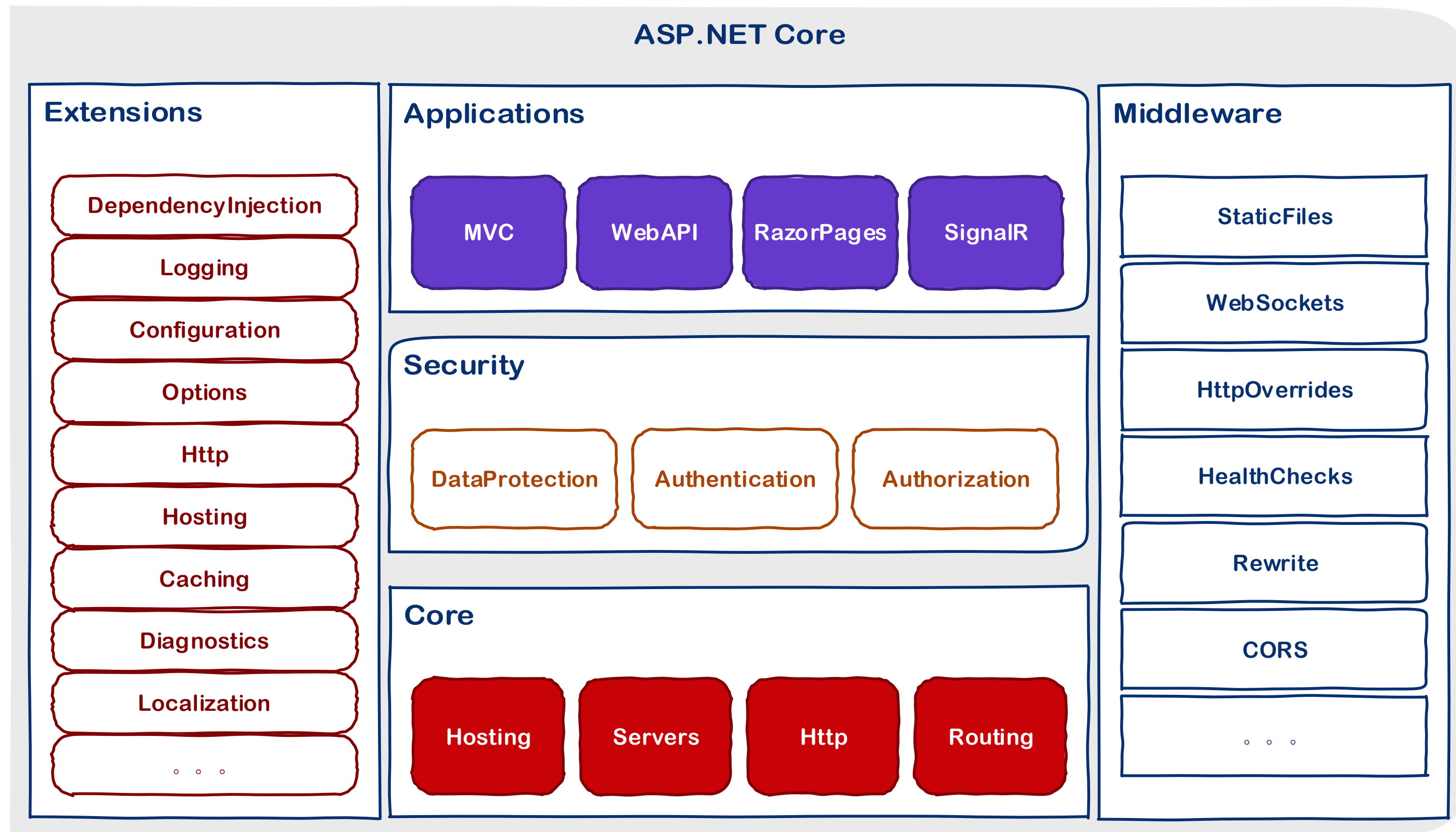
- prometheus-net
- prometheus-net.AspNetCore

场景

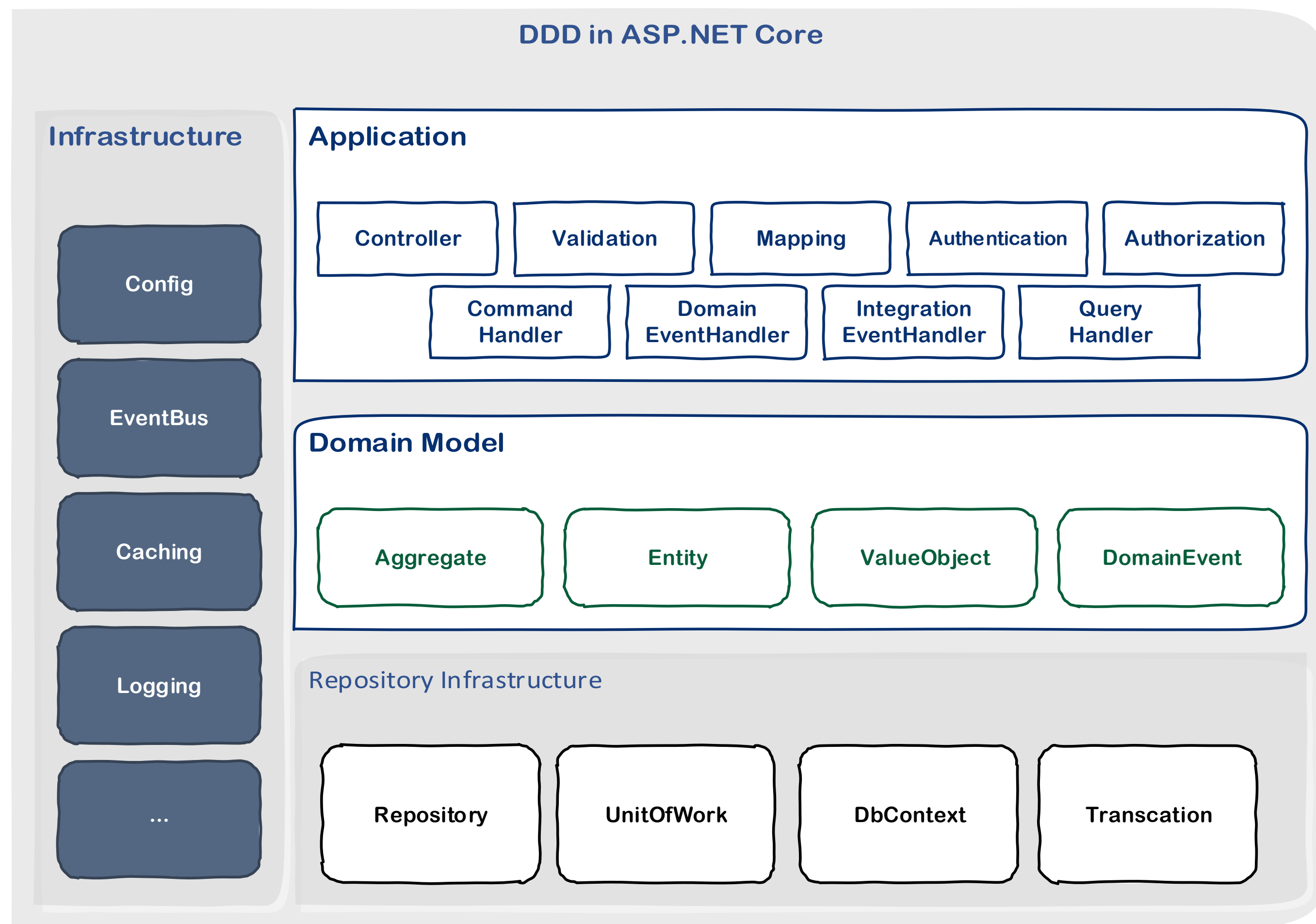
- 应用运行状况指标
- 自定义指标

3.14 结束语

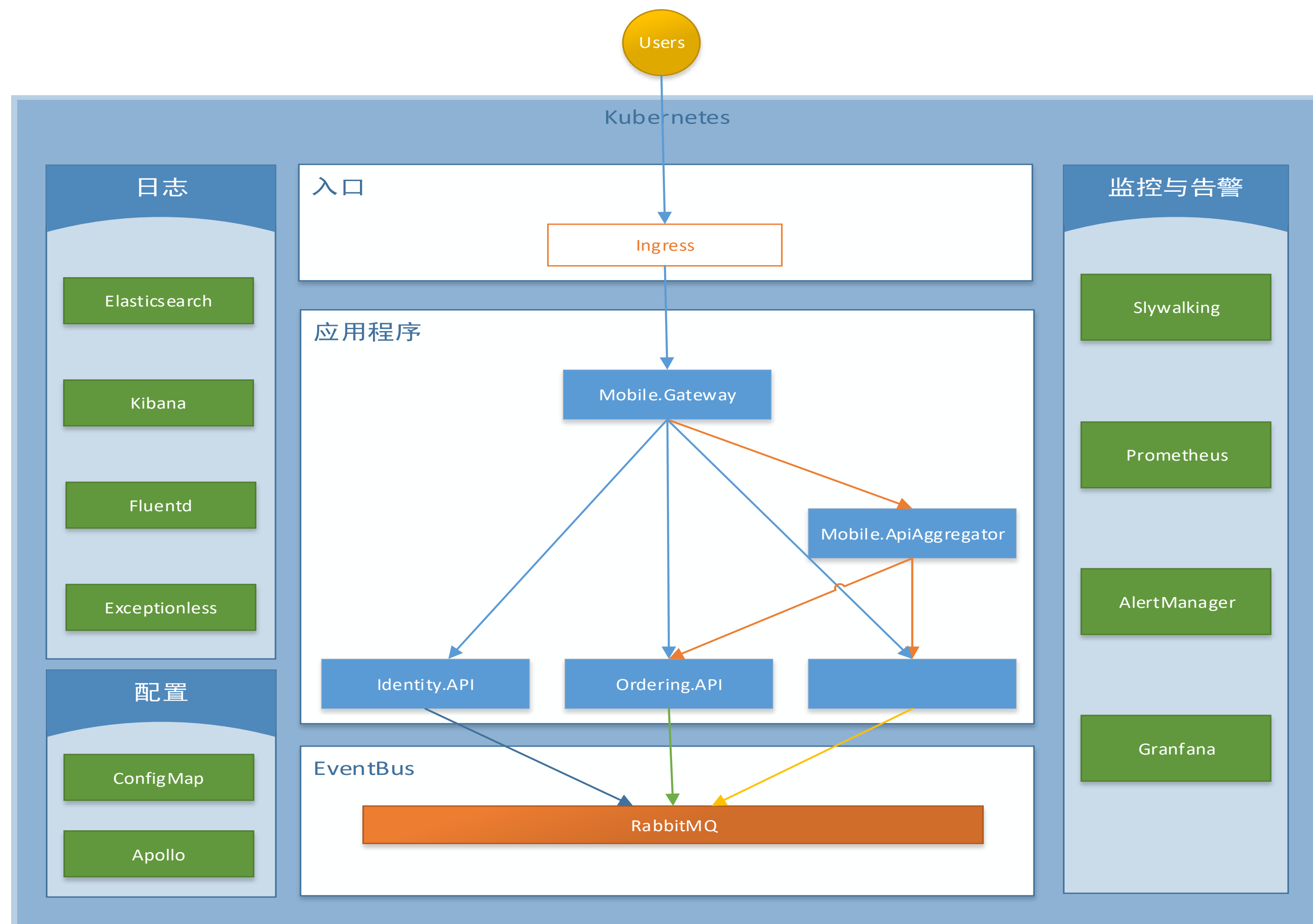
ASP.NET Core 框架



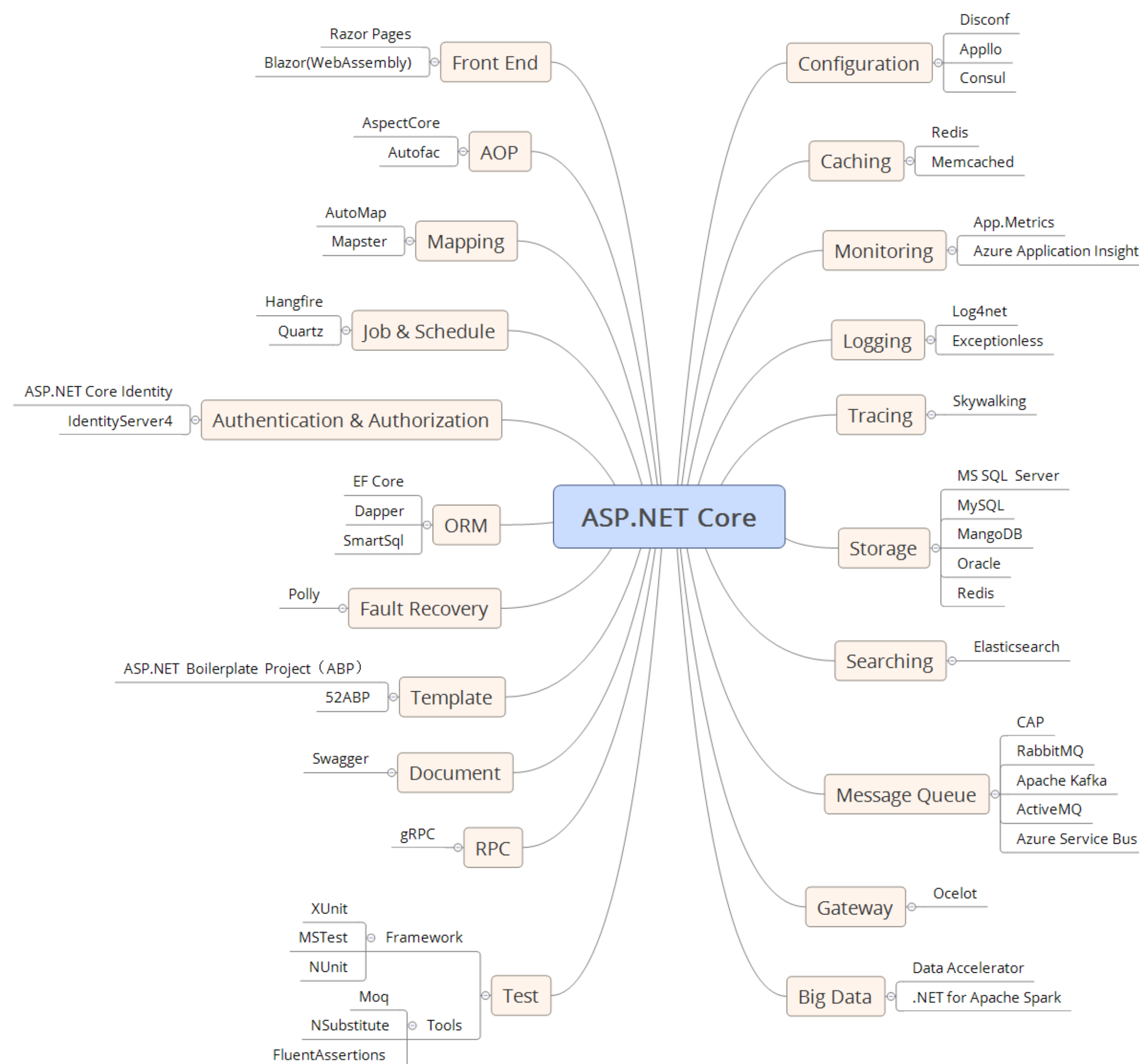
领域驱动设计



微服务架构



ASP.NET Core 生态





扫码试看/订阅

《.NET Core 开发实战》视频课程