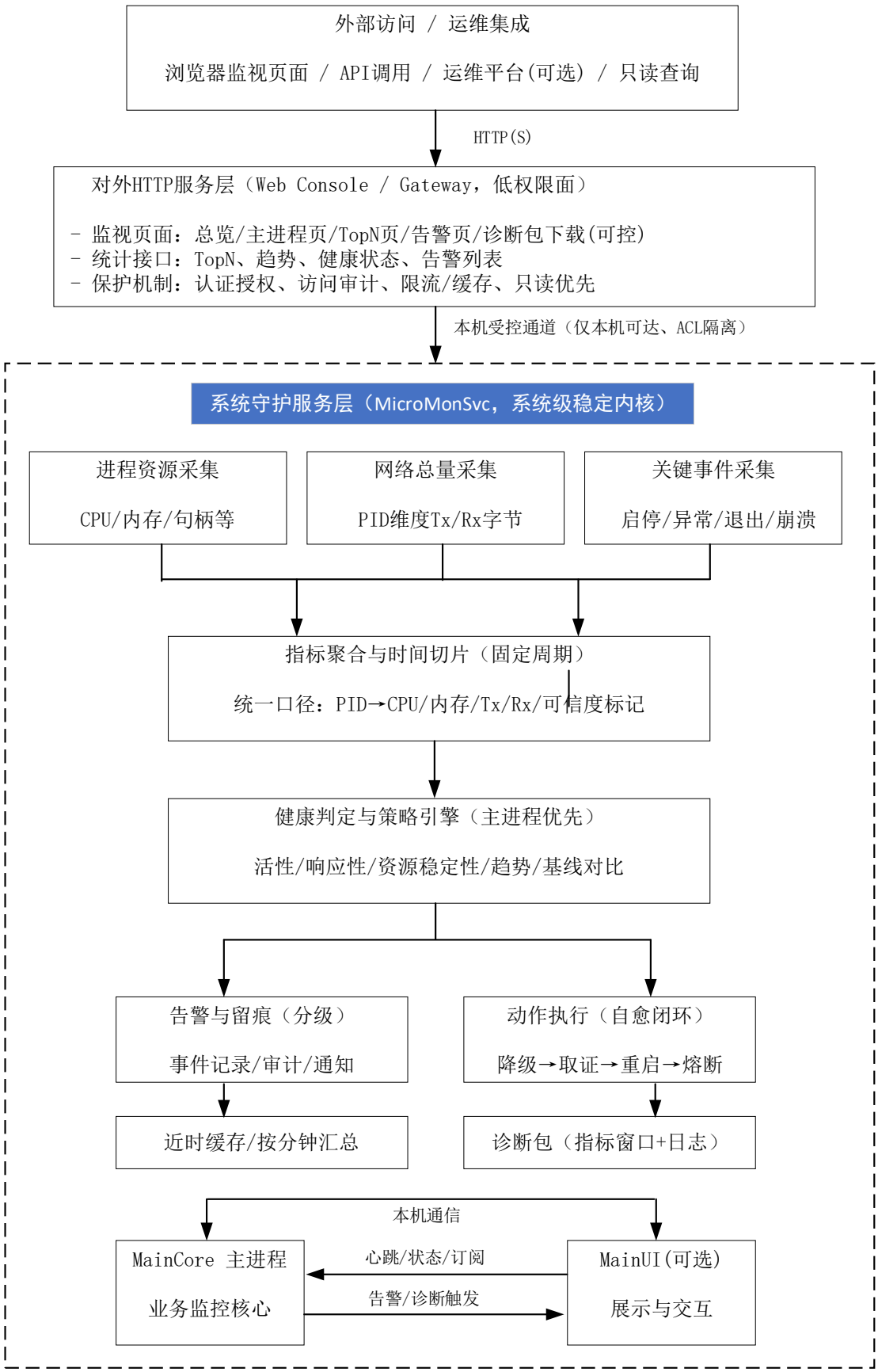

图 1 电厂核心软件监视保障微系统架构设计

该系统采用先进的微系统设计方式进行架构，采用“对外服务层 + 系统守护服务层 + 主进程”的三段式结构：最外层为对外 HTTP 服务，承担监视页面与统计查询；中间层为系统级守护服务，作为稳定内核持续采集全进程 CPU、内存与网络总量并完成健康判定与自愈；最内层为监控软件主进程，专注业务功能与必要的自报状态。三段式结构的关键在于边界清晰：对外 HTTP 面不直接触达系统级能力，只读取守护服务的聚合结果；守护服务不依赖主进程稳定运行即可完成发现、留痕与处置；主进程即便卡死或异常退出，守护服务仍能维持观测链路并触发恢复动作。该结构使“观测能力”不再与“业务能力”同生共死，也使外部访问量、页面渲染与 API 查询不会反向拖垮采集与判定，从工程上显著降低了单点失效与耦合放大风险。

从先进性角度看，该架构体现为四点：其一，外部支点原则——把稳定性保障从主进程内部移出，形成独立的守护内核，从而具备“主进程失效时仍可观测、可处置”的能力；其二，统一口径原则——将不同来源的资源信号在固定周期内完成对齐与时间切片，后续判定只面对结构化序列，规则清晰且可解释；其三，闭环可控原则——告警分级与动作编排强调“先可逆后强制”，并用冷却、次数上限与熔断机制抑制重启风暴，使自动化处置可预期、可审计；其四，访问面隔离原则——对外 HTTP 服务以低权限运行，通过认证、审计、限流与缓存把外部访问变成“可控负载”，避免把系统级能力暴露在网络面上，同时确保监视页面与统计接口在高并发场景仍稳定输出。

守护服务层的工作机制可概括为“采集—聚合—判定—处置—留痕”五步流水线。采集阶段以低侵入方式获取全进程资源画像：CPU 与内存以全量快照为主，避免对单进程频繁重查询引入抖动；网络仅统计进程维度的发送/接收总字节增量，不引入连接细粒度数据结构，从而在“足够用于稳定性判断”的前提下显著降低数据量与处理成本。聚合阶段以固定周期把快照数据与网络增量合并，形成统一的 PID 指标条目，并附带必要的可信度标记（例如事件拥塞或采样滞后导致的可信度下降），确保后续规则不会在“数据不完整”时做出强结论。判定阶段以主进程优先，分别从活性、响应性与资源稳定性三类信号刻画健康状态：活性回答“是否还在”，响应性回答“是否还在工作”，资源稳定性回答“是否正在走向失控”。判定逻辑强调持续时间与趋势，而不是一次性阈值：短时尖峰只记录，持续异常进入告警；内存更看净增与回落情况，网络更看相对基线的持续偏离，以减少误报并更贴近稳定性故障形态。

对外 HTTP 服务层被定位为“可观测出口”，其职责是把守护服务的聚合结果以页面与接口形式对外呈现，而不是在此层进行复杂计算。页面通常分为四类即可满足运维：总体态势（全局负载与告警概览）、主进程专页（心跳、响应性、资源曲线与处置记录）、全进程 TopN（CPU/内存/网络占用排行与趋势）、告警与审计（告警原因码、触发规则摘要、证据窗口指纹）。接口侧同样以只读为主，提供健康状态、TopN、时间窗趋势、告警列表等查询能力。为了保证“外部访问不影响内核稳定”，HTTP 层应具备三类保护：其一，认证授权与最小权限（默认只读，敏感操作需强授权或默认关闭）；其二，访问审计（对外查询与下载行为留痕，便于追踪与合规）；其三，限流与缓存（高频查询命中缓存、趋势分页、TopN 预计算），把外部访问转化为可控负载，避免对内核形成反压。

自愈闭环的设计强调“动作可控、证据先行”。系统将告警分为提示、警告与严重三个级别：提示仅留痕；警告在留痕基础上收集证据窗口（例如近 10–30 分钟指标与关键日志）；严重才进入动作执行。动作顺序遵循先可逆后强制：先降级非关键功能以释放资源，再生成诊断包确保问题可复盘，最后才执行重启以恢复可控状态。为避免重启风暴，动作模块必须实现冷却时间、次数上限与熔断：若短时间内多次处置无效，系统停止自动重启并持续告警，保留现场证据等待人工介入。整个闭环的目标并非“自动化越多越好”，而是保证处置行为可解释、可追溯、可审计，且不会因为处置本身制造更大的不确定性。