

PR #27591 完整报告

sgl-project/sglang

[router] Add request/TTFT/worker metrics + Grafana dashboard to experimental sgl-router

合并时间: 2026-06-09 22:25

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/27591>

执行摘要

本次 PR 为实验性 sgl-router 增加了请求端到端延迟、首 Token 时间、响应状态码以及 Worker 级别健康 / 状态 / 并发等 7 个 Prometheus 指标，并附带一份可直接导入的 Grafana 仪表盘，使该路由器的可观测性达到与模型网关相当的水平。核心实现通过泛化 Histogram、添加断路器单锁快照和 SSE 首字节钩子来实现精准的延迟分桶和 TTFT 仅对成功流记录。全部新增功能均经单元和集成测试覆盖。

功能与动机

实验性 sgl-router 此前仅暴露 6 种指标，运维人员无法直接监控请求延迟、首 Token 时间和 Worker 状态。本次变更的目标是：“Bring the experimental sgl-router's Prometheus surface toward parity with the SMG model gateway's applicable metrics, and add an operator-focused Grafana dashboard so the new router is observable out of the box.” (PR Body)。新增指标使操作员可以开箱即用地观测路由器健康状态和性能。

实现拆解

1. 指标注册表扩展(experimental/sgl-router/src/server/metrics.rs) – 新增 request_duration、ttft（使用通用 Histogram，桶边界由常量 REQUEST_DURATION_BUCKETS 定义）、responses_total 计数器，以及 workers、worker_health、worker_cb_state、worker_inflight_requests 四个 Gauge。Histogram 支持自定义桶边界，并带有 debug_assert 递增检查和非有限值静默丢弃保护。
2. 断路器快照(experimental/sgl-router/src/health/circuit_breaker.rs) – 新增 CircuitBreaker::snapshot() 方法，在单次锁获取内返回 (admit, state_code) 对，避免指标抓取时因两次锁读取到不一致的状态。定义 CircuitSnapshot 结构体。新加全套单元测试覆盖各种状态。
3. SSE 泵首字节钩子(experimental/sgl-router/src/proxy/sse.rs) – bytes_stream_to_body 新增 on_first_byte 参数，在循环中首次读到 Ok 字节块时触发一次（通过 take() 保证仅一次），错误优先流不会触发，确保 TTFT 指标只记录真实生成的 token。
4. 聊天处理器集成(experimental/sgl-router/src/server/routes/chat.rs) – 引入 RecordDurationOnDrop RAII 守卫，实现 Drop 时调用 observe_request_duration。对流式请求，该守卫被移入 SSE 泵的 stream_guards，在流结束时记录端到端延迟；非流式请求在分派处直接记录。TTFT 钩子仅安装在流式分支，且转发函数 forward_streaming_to 对非 2xx 响应丢弃钩子，避免错误体污染 TTFT。

5. Worker 指标采样(experimental/sgl-router/src/server/routes/metrics.rs) – /metrics 端点从 WorkerRegistry 实时采样所有 Worker，构造 WorkerSnapshot 并传给 render_with_workers，使 worker gauge 在抓取时反映最新状态，已移除的 Worker 立即停止输出序列。断路器状态通过单次 snapshot 获取，保证 health 和 state_code 一致。
6. Grafana 仪表盘(experimental/sgl-router/monitoring/) – 新增可导入的 grafana-dashboard.json，包含所有 13 个指标族的 Panels，支持 datasource、model_id、worker_url 模板变量。配套 README.md 说明 Prometheus 抓取配置和导入步骤。
7. 测试覆盖– 包括 Histogram 桶 / 和 / 计数验证、+Inf 溢出、非有限值丢弃、断路器 snapshot 各状态正确性、SSE on_first_byte 仅触发一次且错误流不触发，以及端到端聊天路由测试验证 2xx 流式同时记录 TTFT 和 duration、5xx 流式仅记录 duration 和状态但不记录 TTFT (experimental/sgl-router/tests/proxy/chat_routing.rs)。

关键源码片段

experimental/sgl-router/src/server/metrics.rs

核心指标注册表，新增 7 个指标族，泛化 Histogram，添加新渲染入口。

// experimental/sgl-router/src/server/metrics.rs — 泛化 Histogram 及新指标字段

/// 通用直方图，桶边界由调用方传入，支持 +Inf 桶和非有限值丢弃。

```
pub struct Histogram {
    buckets: Vec<u64>, // 非累积计数，长度 = bounds.len() + 1
    sum: f64,
    count: u64,
}

impl Histogram {
    pub fn new(bounds: &'static [f64]) -> Self {
        // 确保桶边界严格递增
        debug_assert!(bounds.windows(2).all(|w| w[0] < w[1]),
            "bucket bounds must be strictly ascending");
        Self { buckets: vec![0; bounds.len() + 1], sum: 0.0, count: 0 }
    }

    pub fn observe(&mut self, value: f64) {
        if !value.is_finite() { return; } // NaN / Inf 不记录
        self.sum += value;
        self.count += 1;
        for (i, &bound) in self.buckets.iter().enumerate() {
            if value <= bound { self.buckets[i] += 1; return; }
        }
        self.buckets[self.buckets.len() - 1] += 1; // +Inf 桶
    }
}
```

/// 指标注册表（节选新增字段）

```
pub struct MetricsRegistry {
```

```

// ... 原有字段 ...
request_duration: Mutex<HashMap<String, Histogram>>,
ttft_seconds: Mutex<HashMap<String, Histogram>>,
responses_total: Mutex<HashMap<u16, Arc<AtomicU64>>>,
}

impl MetricsRegistry {
    pub fn observe_request_duration(&self, model: &str, value: f64) {
        let mut map = self.request_duration.lock().unwrap();
        map.entry(model.to_string())
            .or_insert_with(|| Histogram::new(REQUEST_DURATION_BUCKETS))
            .observe(value);
    }

    pub fn observe_ttft(&self, model: &str, value: f64) {
        let mut map = self.ttft_seconds.lock().unwrap();
        map.entry(model.to_string())
            .or_insert_with(|| Histogram::new(REQUEST_DURATION_BUCKETS))
            .observe(value);
    }

    pub fn record_response(&self, status: u16) {
        let mut map = self.responses_total.lock().unwrap();
        let counter = map.entry(status)
            .or_insert_with(|| Arc::new(AtomicU64::new(0)));
        counter.fetch_add(1, Ordering::Relaxed);
    }
}

```

experimental/sgl-router/src/health/circuit_breaker.rs

新增 snapshot 方法，提供单锁一致的状态快照，支撑 worker_gauge 渲染。

// experimental/sgl-router/src/health/circuit_breaker.rs — 一致性快照

/// 快照结果：admit 为是否允许请求，state_code 编码当前状态。

#[derive(Debug, Clone, Copy)]

```

pub struct CircuitSnapshot {
    pub admit: bool,
    pub state_code: u8, // 0=closed, 1=open, 2=half_open
}

```

```

impl CircuitBreaker {
    /// 在单锁获取内返回 (admit, state_code) 一致对,
    /// 用于指标抓取，避免两次锁间状态变化导致矛盾。
    pub fn snapshot(&self) -> CircuitSnapshot {
        let g = self.inner.lock().unwrap();
        let (admit, state_code) = match g.state {
            State::Closed => (true, 0),
            State::Open { opened_at } =>

```

```

        (opened_at.elapsed() >= self.config.cool_down, 1),
        State::HalfOpen { probe_in_flight } => (!probe_in_flight, 2),
    };
    CircuitSnapshot { admit, state_code }
}
}

```

experimental/sgl-router/src/proxy/sse.rs

添加 on_first_byte 钩子，支撑 TTFT 指标采集。

```

// experimental/sgl-router/src/proxy/sse.rs — 首字节钩子集成到 SSE 泵

// bytes_stream_to_body 新增 on_first_byte 参数
pub fn bytes_stream_to_body<S, E>(
    stream: S,
    stream_guards: Option<Box<dyn Send + 'static>>,
    on_complete: Option<Box<dyn FnOnce(bool) + Send + 'static>>,
    on_first_byte: Option<Box<dyn FnOnce() + Send + 'static>>, // <-- 新增
) -> Body { ... }

// 在泵循环内，仅对第一个 Ok 块触发一次
let mut on_first_byte = on_first_byte;
while let Some(chunk) = s.next().await {
    // ... 错误处理 ...
    let is_err_chunk = item.is_err();
    if !is_err_chunk {
        if let Some(hook) = on_first_byte.take() { // take 保证只触发一次
            hook(); // 记录 TTFT
        }
    }
    // ...
}

```

评论区精华

无。

风险与影响

- 风险：指标注册表使用 Mutex 保护，在极高并发抓取时可能成为瓶颈，但当前 15s 抓取间隔下风险低。SSE 泵的 on_first_byte 逻辑若误触发可能污染 TTFT，但 take() 语义和测试保证了正确性。断路器快照返回的 admit 与 state_code 在 cooldown 边界时可能出现逻辑“不一致”（Open + admit=true），但这是设计意图，运维人员需理解。
- 影响：操作员可立即使用 Grafana 监控路由器；路由器新增约 10 个指标，内存开销可忽略；代码库新增 3000+ 行但主要为仪表盘配置，核心逻辑 500+ 行；所有变更都有测试覆盖，回归风险较低。

关联脉络

本次 PR 是实验性 `sgl-router` 可观测性建设的第二步（此前已有 6 个基础指标）。后续可能继续添加 PD 池级别的指标或预填 / 解码分阶段的延迟指标。与模型网关的指标对齐也为未来统一监控面板铺平了道路。历史 PR #27615（为推测解码添加 NVTX）可视为同一可观测性主题在另一模块的体现。