PR #23309 完整报告

sgl-project/sglang

[HiCache] feat: default storage prefetch timeout

合并时间：2026-05-12 09:49

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/23309

执行摘要

- 一句话：更改 HiCache 预取超时默认值并添加上限
- 推荐动作：本 PR 是生产环境优化的重要一步，值得阅读其设计决策：引入硬上限防止长 prompt 无限等待，以及对默认超时参数的理论推导。如果有自定义预取策略的用户需要注意默认行为的变更。

功能与动机

Best-effort prefetch often misses HiCache reuse because requests proceed before storage prefetch completes. Make timeout the default to improve practical cache-hit behavior while keeping waiting bounded and providing more production-friendly defaults. The previous defaults — prefetch_timeout_base=1s, prefetch_timeout_per_ki_token=0.25s — produced very long timeouts on large prompts (e.g. ~257s at 1M tokens) and no upper bound at all, which is unfriendly for serving SLAs. New defaults bring both per-request overhead and tail latency down, and add a hard cap.

实现拆解

1. 在 hicache_storage.py 中新增 PrefetchTimeoutConfig 冻结数据类，封装 base、per_ki_token、max 三个超时参数，默认值分别为 2.0s、0.1s、30.0s。
2. 在 hi_mamba_radix_cache.py 和 hiradix_cache.py 中修改 _parse_storage_backend_extra_config 返回值为一个 PrefetchTimeoutConfig 对象，替换原来的两个独立浮点参数。同时更新 _apply_storage_runtime_config 和 attach_storage_backend 方法，统一使用 prefetch_timeout_config。
3. 在 server_args.py 中将 hicache_storage_prefetch_policy 的默认值从 "best_effort" 改为 "timeout"，使新超时策略成为默认行为。
4. 在 hi_mamba_radix_cache.py 和 hiradix_cache.py 的 _apply_storage_runtime_config 中移除 prefetch_timeout_per_page 的独立计算，直接使用 PrefetchTimeoutConfig.per_ki_token 和 max，在 _prefetch_timeout_check_linear_func 中实现 min(max, base + per_ki_token * tokens/1024)。

5. 更新四份文档（hicache_design.mdx、hicache_storage_runtime_attach_detach.mdx、server_arguments.mdx、ascend_npu_support_features.mdx），反映新参数和默认值。

关键文件：

- python/sglang/srt/mem_cache/hicache_storage.py（模块 缓存层；类别 source；类型 core-logic；符号 PrefetchTimeoutConfig）：新增 PrefetchTimeoutConfig 数据类，定义超时参数的默认值和封装。
- python/sglang/srt/server_args.py（模块 参数配置；类别 source；类型 core-logic）：变更默认预取策略从 best_effort 到 timeout，影响所有 HiCache 用户。
- python/sglang/srt/mem_cache/hiradix_cache.py（模块 缓存层；类别 source；类型 core-logic；符号 _parse_storage_backend_extra_config, _apply_storage_runtime_config, attach_storage_backend）：核心缓存类，修改参数解析和应用逻辑以支持统一的 PrefetchTimeoutConfig。
- python/sglang/srt/mem_cache/hi_mamba_radix_cache.py（模块 缓存层；类别 source；类型 dependency-wiring；符号 _parse_storage_backend_extra_config, _apply_storage_runtime_config, attach_storage_backend）：HiMambaRadixCache 类同样需要参数解析适配，变化与 hiradix_cache.py 类似。
- docs_new/docs/advanced_features/hicache_design.mdx（模块 文档；类别 other；类型 core-logic）：更新了超时计算公式和默认值文档。
- docs_new/docs/advanced_features/hicache_storage_runtime_attach_detach.mdx（模块 文档；类别 other；类型 core-logic）：更新 extra_config 可包含 prefetch_timeout_max。
- docs_new/docs/advanced_features/server_arguments.mdx（模块 文档；类别 other；类型 core-logic）：更新默认策略值。
- docs_new/docs/hardware-platforms/ascend-npus/ascend_npu_support_features.mdx（模块 文档；类别 other；类型 core-logic）：更新默认策略值。

关键符号：PrefetchTimeoutConfig, _parse_storage_backend_extra_config, _apply_storage_runtime_config, attach_storage_backend, _prefetch_timeout_check_linear_func

关键源码片段

python/sglang/srt/mem_cache/hicache_storage.py

新增 PrefetchTimeoutConfig 数据类，定义超时参数的默认值和封装。

```
@dataclass(frozen=True)
class PrefetchTimeoutConfig:
    """Knobs for the linear prefetch-timeout policy used by HiCache."""

    base: float = 2.0 # 固定开销，与 token 数无关，单位秒
    per_ki_token: float = 0.1 # 每 1024 个 token 的增量超时，单位秒
    max: float = 30.0 # 线性超时的硬上限，单位秒
```

python/sglang/srt/server_args.py

变更默认预取策略从 best_effort 到 timeout，影响所有 HiCache 用户。

```
# Hierarchical cache
enable_hierarchical_cache: bool = False
hicache_ratio: float = 2.0
hicache_size: int = 0
hicache_write_policy: str = "write_through"
hicache_io_backend: str = "kernel"
hicache_mem_layout: str = "layer_first"
hicache_storage_backend: Optional[str] = None
hicache_storage_prefetch_policy: str = "timeout" # 默认从 best_effort 改为 timeout
hicache_storage_backend_extra_config: Optional[str] = None
```

python/sglang/srt/mem_cache/hiradix_cache.py

核心缓存类，修改参数解析和应用逻辑以支持统一的 PrefetchTimeoutConfig。

```
from sglang.srt.mem_cache.hicache_storage import (
    PoolHitPolicy,
    PoolName,
    PoolTransfer,
    PrefetchTimeoutConfig, # 新增导入
)

# 在 __init__ 中解析 extra_config
(
    extra_config,
    prefetch_threshold,
    prefetch_timeout_config, # 改为单个 PrefetchTimeoutConfig 对象
    hicache_storage_pass_prefix_keys,
) = self._parse_storage_backend_extra_config(
    server_args.hicache_storage_backend_extra_config
)

# 传递给运行时配置
self._apply_storage_runtime_config(
    storage_backend=server_args.hicache_storage_backend,
    prefetch_threshold=prefetch_threshold,
    prefetch_timeout_config=prefetch_timeout_config, # 传递对象
    hicache_storage_pass_prefix_keys=hicache_storage_pass_prefix_keys,
    enable_storage=self.enable_storage,
    enable_storage_metrics=self.enable_storage_metrics,
    extra_metric_labels=self.extra_metric_labels,
)
```

评论区精华

合并者 xiezhq-hermann 在 Issue 评论中询问默认值是否满意（'btw are we happy about the default value for timeout?'），未收到明确反对，最终 PR 被批准。另外，chatgpt-codex-connector 评论提醒更新相关文档以匹配新的默认策略，该建议已被采纳。

- 默认超时值的满意度讨论 (question): 未收到反对意见，PR 被批准，默认值被采纳。
- 文档更新提醒 (documentation): PR 的后续提交（或最终版本）中已更新相关文档，建议被采纳。

风险与影响

- 风险：默认策略从 best_effort 改为 timeout 可能导致依赖旧默认行为的用户遇到不同的预取等待行为，尤其是长 prompt 场景下超时上限 30s 可能截断原本可完成的预取，影响缓存命中率。此外，配置项 prefetch_timeout_max 新增，需要确保向后兼容：如果用户配置了旧格式的 extra_config 字符串，解析应能正确处理。文档更新可能仍有遗漏。
- 影响：影响范围限于启用 HiCache 存储后端的用户，默认情况下（未设置 hicache_storage_prefetch_policy）从 best_effort 切换到 timeout，带来更可控的等待时间。短 prompt 超时略有增加（1K tokens 从 1.25s 到 2.1s），但长 prompt 显著降低。系统层面无性能风险。团队需要确保文档与行为一致。
- 风险标记：默认策略变更，文档同步，长 prompt 超时限制

关联脉络

- 暂无明显关联 PR