# PR #25724 完整报告

sgl-project/sglang

Return a mamba tracking entry from the cache lookup instead of mutating caller lists

合并时间：2026-05-19 09:21

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/25724

## 执行摘要

- 一句话：重构 Mamba 缓存查询返回 NamedTuple
- 推荐动作：该 PR 是典型的"提取返回值"重构，代码整洁度提升明显，值得精读以学习如何消除跨方法副作用。

## 功能与动机

PR 标题和提交信息明确指出，旧版本中 _mamba_radix_cache_v2_req_prepare_for_extend 接收三个调用方拥有的列表并在内部进行追加操作，这些副作用跨越了方法体，不够透明。通过返回一个 NamedTuple 并在调用方显式追加，可以将副作用集中管理，使代码意图更清晰。

## 实现拆解

1. 定义 NamedTuple：在 schedule_batch.py 中，于 ScheduleBatch 类之前新增 _MambaRadixCacheV2TrackEntry NamedTuple，包含 track_mask（bool）、track_index（int）、track_seqlen（int）三个字段。
2. 修改导入：在文件顶部的 typing 导入中增加 NamedTuple。
3. 重构方法签名：将 _mamba_radix_cache_v2_req_prepare_for_extend 的参数从三个列表（mamba_track_mask_cpu、mamba_track_indices_cpu、mamba_track_seqlens_cpu）简化为仅接收 req: Req，返回类型改为 _MambaRadixCacheV2TrackEntry。
4. 方法内部调整：移除方法内部的三个 .append() 调用，改为在方法末位返回一个 _MambaRadixCacheV2TrackEntry(mask, index, seqlen) 实例。其中 track_index 的获取方式由 mamba_track_indices_cpu.append(...) 改为本地变量 track_index = ...。
5. 调用方适配：在 ScheduleBatch.prepare_for_extend 中，将原来的调用方式替换为 track_entry = self._mamba_radix_cache_v2_req_prepare_for_extend(req)，然后显式将 track_entry 的三个字段分别追加到对应的列表中。

该重构不涉及测试、配置或部署配套改动，属于纯源代碼重构。

关键文件：

- python/sglang/srt/managers/schedule_batch.py（模块 调度；类别 source；类型 core-logic；符号 _MambaRadixCacheV2TrackEntry）：唯一变更文件，包含了核心逻辑的修改和新增序号定义。

关键符号：_mamba_radix_cache_v2_req_prepare_for_extend, prepare_for_extend, repr

## 关键源码片段

### python/sglang/srt/managers/schedule_batch.py

唯一变更文件，包含了核心逻辑的修改和新增序号定义。

```
# 文件：python/sglang/srt/managers/schedule_batch.py
# 以下为 PR 新增的 NamedTuple 定义及调用处关键变更

class _MambaRadixCacheV2TrackEntry(NamedTuple):
    track_mask: bool
    track_index: int
    track_seqlen: int


@dataclasses.dataclass
class ScheduleBatch(ScheduleBatchDisaggregationDecodeMixin):
    ...

    def prepare_for_extend(self):
        ...
        if get_global_server_args().enable_mamba_extra_buffer():
            # 以前直接传入列表让方法追加，现在返回条目后再显式追加
            track_entry = self._mamba_radix_cache_v2_req_prepare_for_extend(req)
            mamba_track_mask_cpu.append(track_entry.track_mask)
            mamba_track_indices_cpu.append(track_entry.track_index)
            mamba_track_seqlens_cpu.append(track_entry.track_seqlen)
        ...

    def _mamba_radix_cache_v2_req_prepare_for_extend(
        self, req: Req,
    ) -> "_MambaRadixCacheV2TrackEntry":
        # 方法内部不再直接操作外部列表，而是构造并返回 NamedTuple
        mask = req.extend_input_len >= mamba_cache_chunk_size
        track_index = req.mamba_ping_pong_track_buffer[
            req.mamba_next_track_idx
        ].item()
        ... # 计算 mamba_track_seqlen ...
        return _MambaRadixCacheV2TrackEntry(
            track_mask=mask,
            track_index=track_index,
            track_seqlen=mamba_track_seqlen,
        )
```

## 评论区精华

无 review 评论或讨论。

- 暂无高价值评论线程

## 风险与影响

- 风险：重构范围限定在单个私有方法及其调用处，变更逻辑等价，且 PR 作者即为合并者，风险低。但缺少对应的单元测试覆盖该私有方法的返回行为，若未来有他人修改该方法逻辑，可能因未理解返回值结构而引入 bug。
- 影响：影响范围仅限于 python/glang/srt/managers/schedule_batch.py 中 ScheduleBatch 类的两个方法（prepare_for_extend 和 _mamba_radix_cache_v2_req_prepare_for_extend）。对外部模块无影响，属于局部重构。对用户透明，无功能变化。
- 风险标记：缺少测试覆盖

## 关联脉络

- 暂无明显关联 PR