

# PR #29514 完整报告

sgl-project/sglang

[diffusion] fix --warmup silently downgrading server-based warmup to request mode

合并时间: 2026-06-28 11:46

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/29514>

## 执行摘要

- 一句话: 修复 --warmup 静默降级为 request 模式
- 推荐动作: 值得合并。这是一个典型的“配置继承边界”bug: default\_args 未记录为 explicit 导致注入的默认值与显式标志交互异常。PR 分析透彻、修复精准、单测完备, 且 H100 实测数据验证有效。值得其他 CLI 参数解析模块参考此模式 (default\_args 与 explicit 标志的一致性)。

## 功能与动机

PR body 明确指出 `sglang serve --warmup` 本应使用 server-based warmup, 但因 `serve` 注入的 `warmup_mode="server"` 通过 `default_args` 传递且未被记录为显式参数, 导致 `_adjust_warmup` 中条件判断 `mode_explicit or not legacy_explicit` 为 `False`, 结果 `server_warmup` 回退到 dataclass 默认值 `False` (即 request 模式)。组合 `--warmup --warmup-resolutions` 时更形成死区: server warmup 不跑, request warmup 因 resolutions 非 `None` 也跳过, 最终无任何 warmup 执行, 首个请求承担 ~2x 冷启动延迟。

## 实现拆解

1. 入口: `python/sglang/multimodal_gen/runtime/server_args.py` 中 `_adjust_warmup` 方法负责解析 warmup 相关参数, 决定 `self.warmup` 和 `self.server_warmup`。
2. 原逻辑分支: `if mode_explicit or not legacy_explicit:` 已处理了两种 case: 显式指定了 `--warmup-mode`, 或 `legacy` 标志未显式。漏掉的是 `legacy` 标志显式 (如 `--warmup`) 但 `warmup_mode` 未显式 (由 `serve` 注入默认值) 的 case, 降级为 request 模式。
3. 修复: 在该分支后新增 `elif self.warmup:` 分支, 当 `legacy` 标志显式且 warmup 被启用时, 设置 `self.server_warmup = self.server_warmup or self.warmup_mode == "server"`, 此处 `self.warmup_mode` 保留的是 "server" 或 `None` 等已解析值, 从而正确启用 server warmup。
4. 测试配套: 在 `test_server_args.py` 的 `TestWarmupModeNormalization` 类中新增两个测试用例: `test_legacy_warmup_on_uses_defaulted_server_mode` 验证 `--warmup` 带默认 server mode 时正确设置 `server_warmup=True`; `test_legacy_warmup_with_resolutions_runs_server_warmup` 验证死区回归: `--warmup --warmup-resolutions` 时执行 server warmup。
5. 配套变更: 无其他配置或部署变更。仅 2 行源码 + 23 行测试, 改动极小。

关键文件：

- python/sglang/multimodal\_gen/test/unit/test\_server\_args.py（模块 测试；类别 test；类型 test-coverage；符号 test\_legacy\_warmup\_on\_uses\_defaulted\_server\_mode, test\_legacy\_warmup\_with\_resolutions\_runs\_server\_warmup）：新增两个单元测试，验证修复的正确性并防止回归：test\_legacy\_warmup\_on\_uses\_defaulted\_server\_mode 和 test\_legacy\_warmup\_with\_resolutions\_runs\_server\_warmup。覆盖了最关键的死区回归场景。
- python/sglang/multimodal\_gen/runtime/server\_args.py（模块 服务器参数；类别 source；类型 core-logic；符号 \_adjust\_warmup）：核心修复文件。在 \_adjust\_warmup 方法中新增 elif 分支，处理 legacy 标志显式而 warmup\_mode 未显式时的 server\_warmup 赋值。仅 +2 行代码。

关键符号：\_adjust\_warmup

## 关键源码片段

### python/sglang/multimodal\_gen/test/unit/test\_server\_args.py

新增两个单元测试，验证修复的正确性并防止回归：test\_legacy\_warmup\_on\_uses\_defaulted\_server\_mode 和 test\_legacy\_warmup\_with\_resolutions\_runs\_server\_warmup。覆盖了最关键的死区回归场景。

```
def test_legacy_warmup_on_uses_defaulted_server_mode(self):
    # `serve --warmup` (legacy ON, mode defaulted to "server" but not
    # explicit) must resolve to server-based warmup, not silently downgrade
    # to request mode.
    sa = self._resolve(
        warmup_mode="server", # 由 serve CLI 注入默认值
        warmup=True,
        explicit=("warmup",), # 仅 --warmup 显式传入
    )
    self.assertEqual(sa.warmup_mode, "server")
    self.assertTrue(sa.warmup)
    self.assertTrue(sa.server_warmup)

def test_legacy_warmup_with_resolutions_runs_server_warmup(self):
    # Dead-zone regression: `serve --warmup --warmup-resolutions X` must run
    # server-based (synthetic) warmup, not end up with no warmup at all
    # (request-based warmup bails out when warmup_resolutions is set).
    sa = self._resolve(
        warmup_mode="server",
        warmup=True,
        warmup_resolutions=["1024x1024"],
        explicit=("warmup",),
    )
    self.assertTrue(sa.warmup)
    self.assertTrue(sa.server_warmup)
    self.assertEqual(sa.warmup_mode, "server")
```

## python/sglang/multimodal\_gen/runtime/server\_args.py

核心修复文件。在 `_adjust_warmup` 方法中新增 `elif` 分支，处理 `legacy` 标志显式而 `warmup_mode` 未显式时的 `server_warmup` 赋值。仅 +2 行代码。

```
def _adjust_warmup(self):
    # --warmup-mode > --warmup/--server-warmup
    mode_explicit = self.is_arg_explicitly_set("warmup_mode")
    legacy_explicit = self.is_arg_explicitly_set(
        "warmup"
    ) or self.is_arg_explicitly_set("server_warmup")
    if self.warmup_mode is not None:
        if self.warmup_mode not in WARMUP_MODES:
            raise ValueError(
                f"Invalid --warmup-mode {self.warmup_mode!r}; "
                f"expected one of {WARMUP_MODES}."
            )
        if mode_explicit and legacy_explicit:
            logger.warning(
                "Both --warmup-mode and the deprecated --warmup/--server-warmup "
                "were set; --warmup-mode=%s takes precedence.",
                self.warmup_mode,
            )
        if mode_explicit or not legacy_explicit:
            # mode_explicit: 用户显式指定 --warmup-mode, 直接使用
            # not legacy_explicit: 无 legacy 标志, 直接用 warmup_mode (如 bare `serve`
            # 注入的默认值)
            self.warmup = self.warmup_mode != "off"
            self.server_warmup = self.warmup_mode == "server"
        elif self.warmup:
            # [!] 新增分支: legacy 标志显式 (如 --warmup) 但 warmup_mode 未显式
            # 此时应沿用已解析的 warmup_mode (如 serve 注入的 "server")
            self.server_warmup = self.server_warmup or self.warmup_mode == "server"

    # Explicit resolutions imply warmup is on (request-based).
    if self.warmup_resolutions is not None:
        self.warmup = True

    if self.disagg_role != RoleType.MONOLITHIC:
        self.server_warmup = False

    if not self.warmup:
        self.server_warmup = False

    self.warmup_mode = (
        "off" if not self.warmup
        else "server" if self.server_warmup
        else "request"
    )
```

## 评论区精华

仅有一条自动 Code Review 评论来自 [gemini-code-assist\[bot\]](#)，确认变更正确且无 review 意见。无人工 review 讨论。PR 作者 @mickqian 在 body 中详尽分析了根因和修复逻辑，并提供了 H100 实测数据：修复后 Z-Image-Turbo 从 13.46s（无 warmup）降至 0.74s（vs baseline 0.65s），其余 diffusion 模型也恢复至 baseline 水平。

- 自动 Code Review 确认无额外意见 (other): 无额外修改需求。

## 风险与影响

- 风险：风险极低：
  - 仅新增一个 `elif self.warmup:` 分支，不会影响已有分支逻辑（warmup 为 False 时跳过，与之前行为一致）。
  - 单测覆盖了 legacy 显式 + 默认 server mode、legacy + resolutions 两种关键路径。
  - 改动涉及 server args 解析，属冷启动路径，不影响推理运行时。
  - 但 `_adjust_warmup` 中逻辑分支较多，需确保无其他未覆盖的组合（如 `--server-warmup` 不带 `--warmup-mode`）。现有单测已涵盖 `--warmup false` 和 `bare serve` 等场景，回归风险小。
  - 影响：用户影响：修复后 `sglang serve --warmup` 和 `--warmup --warmup-resolutions` 正确执行 server-based warmup，diffusion 模型首个请求延迟降低约 50%-90%（如 Z-Image-Turbo 从 13.46s 降至 0.74s）。所有依赖 CLI warmup 的 diffusion 用户受益。系统影响：无性能或内存影响。团队影响：降低因 warmup 配置陷阱导致的 debug 成本。
- 风险标记：分支逻辑嵌套需谨慎，单测覆盖充分

## 关联脉络

- 暂无明显关联 PR