sglang 2026 年第 33 周（08-10 至 08-16）周报

sgl-project/sglang

周期：2026-08-10 至 2026-08-16

来源 PR：352・重点 PR：24・自动生成

原文链接：http://prhub.com.cn/sgl-project/sglang/reports/2026-08-10-to-2026-08-16

执行摘要

本周（2026-08-10 至 08-16）sglang 共合并 352 个 PR，其中重点 PR 24 个，平均重要性 6.52。变化主线集中在三个方向：多模态 /Diffusion 原生化、缓存与内核基础设施重构以及 配置系统一致性收口。mickqian（41 个）和 BBuf（33 个）贡献了最多代码，diffusion（76）、bugfix（112）、performance（90）是标签热点。同时，NPU、MLX、AMD 等多硬件适配步伐加快，JIT 构建系统重写带来显著的开发者体验提升。

本周重点变化

- Diffusion 原生模型大批落地。Hunyuan3D Paint/Delight（#34980）迁移为原生 SD 2.1 兼容实现，去噪单步约 4 倍提速；Qwen2.5-VL（#34896）和 Qwen3-VL（#34945）视觉编码器原生化，并修复 H100 上的 FP32 RoPE 精度回归；LTX-2.5（#34471）通过架构确认复用了 LTX-2 管线，新增时长预测头与扩散解码器。
- 缓存基础设施重构。HiCache L2 传输改为扁平命令执行（#34793），消除递归分发；CUDA VMM 分配逻辑统一（#34199）收口四类消费者；多模态侧新增内容寻址预缓存基础设施（#34398），为 Kimi-K3 per-image 缓存（#34404）提供底层支持。
- 配置一致性迁移。ch-wan 的 config bags 系列 PR（#34263~#34819）分多步将 runner 侧的 server_args 直读迁移到 bags，覆盖注意力后端决策、动态配置读取等，并以静态 ratchet 测试锁定暴露面。
- JIT 构建系统重写。内容寻址的 ninja 构建缓存（#34274）用自生成 build.ninja 替代正则解析，双 key 追踪依赖，冷启动从 5.26s 降至 0.05s，并修复多个边角问题。
- 多硬件适配。NPU 上 Kimi-K3（#33465）全链路支持，MiniMax M3 w8a8（#32941）打通稀疏注意力和 Eagle3 投机；MLX 后端窗口化 SWA KV（#34166）大幅降低内存占用；AMD 新增 NVFP4 在线转 MXFP4 量化（#29328）。
- 安全性。远程媒体 URL 白名单与下载限制（#34892）落地，配合多模态内容哈希缓存，收紧输入边界。

模块与主题趋势

从标签看，bugfix（112）和 performance（90）占据主要比重，diffusion（76）紧随其后，说明本周是“修问题 + 提性能 + 扩模型”并行的一周。热点文件集中在 server_args.py（12 次改动）和 environ.py（7 次），反映配置与环境变量管理是持续演化的焦点；Kimi-K3 相关文件出现 5 次，说明该模型是重点调试对象。主题趋势上，“原生化”是明确方向——模型、编码器、量化后端、JIT 构建都在从外部依赖走向自包含实现，配合位精确（bit-exact）验证，试图在不损失数值一致性的前提下获得性能收益。另一个值得注意的主题是 配置读取收敛：大量

PR 通过 config bags 和静态检查将散落的 server_args 访问集中管理，降低配置漂移风险。

风险观察

本周风险标签中“核心路径变更”高达42次，其中涉及缓存、调度、注意力后端的重构尤为密集。多个 PR 在合入时缺少直接测试覆盖（27+5+3），特别是 NPU、AMD、MLX 等平台路径多依赖手动验证或夜间测试，如 #33465 的 NPU 后端无自动化测试，#32941 的 CUDA 路径需回归验证。配置 bags 重构涉及默认行为变更（如显式 / 隐式区分），虽然以静态测试锁定，但运行期语义仍需要观察。此外，#34793 存在 CI 失败但作者以维护者身份合并的情况，这种“绕过门禁”的做法可能掩盖回归，值得关注。

重点 PR 速览

- #34274 [kernel] Content-addressed JIT build cache：重写 JIT 构建系统，自生成 ninja + 双 key 缓存，冷启动从 5.26s 降至 0.05s，修复 ROCm 依赖追踪和 TP rank 重复构建问题，是基础设施级改进。
- #34793 refactor(hicache): flatten L2 transfer execution：将 HiCache L2 传输扁平化为命令列表，消除嵌套分发，职责边界清晰，但 CI 失败仍被合并，验证数据需复跑。
- #34471 [diffusion] Support LTX-2.5：通过 meta device 校验确认架构复用 LTX-2.3，新增时长预测头和扩散解码器，支持 TP/SP/CFG 并行，文档多轮优化后合入。
- #34404 [VLM] Cache Kimi-K3 per-image processor artifacts：引入模型无关的 per-media artifact 缓存，Kimi-K3 成为首个 typed adapter，冷热命中位级一致，为调度器 lease 复用铺路。
- #34206 [EPD] Pipeline owner-only multimodal preprocessing：重构 EPD 图像预处理为 owner 流水线，Kimi-K3 吞吐最高 +30%，展示延迟物化抽象。
- #32017 [Model Loading] Overlap checkpoint staging with CUDA graph capture：启动期权重预取与 CUDA graph 捕获重叠，通过哨兵权重 + 存储清单保障正确性，修复了多个 review 发现的正确性问题。
- #34166 [MLX] Window-bounded SWA KV storage and in-graph sampling：MLX 后端窗口化 KV 存储减少约 47-49% 内存，图内采样避免 CPU 桥接，是 MLX 后端重要里程碑。

后续建议

1. 补强测试：针对缺少测试覆盖的新模型和平台路径，尽快补充单元级或注册级测试，尤其是 NPU/MLX/AMD 路径的自动化回归。
2. 验证配置迁移：config bags 大范围改动后，建议在 RC 阶段进行完整场景的端到端验证，特别是 split 加载、投机解码、LoRA 等组合路径。
3. 关注缓存一致性：多个缓存重构（HiCache、preprocess cache、JIT cache）并行进行，需要制定统一的稳定性验证矩阵，防止交叉影响。
4. 规范 CI 纪律：明确“CI 失败需公开原因并重跑”的规则，避免维护者权限被当作绕过门禁的捷径。
5. 生态扩展：serve 后端插件机制（#34753）落地，可进一步打磨文档和示例，吸引更多外部运行时集成。