PR #36740 完整报告

sgl-project/sglang

cookbook: add a Speculative card to the GLM-5.3-Flash playground

合并时间：2026-08-28 07:04

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/36740

执行摘要

本 PR 为 GLM-5.3-Flash cookbook 的 Playground 交互组件新增 Speculative卡片，让读者无需手改命令即可在 Inherited / Off / EAGLE（Adaptive MTP 5-1-6）/ DFlash2之间切换投机解码算法，并补上 DFlash2 这一此前完全不可达的 block-diffusion 方案。作为支撑，Playground 引擎修复了 --speculative-adaptive 的孤儿 / 重复 flag 问题，并新增选项级 note 机制用于展示命令运行前的前置条件。整体为文档站点改动，无运行时代码路径变化，但 _playground.jsx 的引擎修复对全站 cookbook 的 flag 组合正确性有正面影响。

功能与动机

GLM-5.3-Flash 的 Deploy 面板只能通过 Strategy维度触达投机解码：Low Latency 意味着 adaptive MTP 5/1/6，High Throughput 意味着关闭。读者想换算法、或在 High Throughput 的其他设置上叠加 MTP，只能手改生成的命令。DFlash2 尤其不可达——它的 block-diffusion draft 是独立 checkpoint（incoai/GLM-5.3-Flash-DFlash2），没有任何 cell 引用。本 PR 通过 Playground 的 speculative 轴将算法选择变成一等交互。

实现拆解

1. 新增 speculative 轴配置（docs/src/snippets/configs/zai-org/glm-5.3-flash.jsx）：在 playgroundFeatures 中声明 speculative 卡片，含 4 个选项：
   - Inherited from base（保持 base 不动）
   - Off (greedy)（剥离整个 --speculative-* 家族）
   - EAGLE / Adaptive MTP 5-1-6：flag 与 Low Latency cells 字节一致，保证 base 派生命中该 chip，重复点选为 no-op
   - DFlash2：--speculative-algorithm DFLASH + --speculative-draft-model-path incoai/GLM-5.3-Flash-DFlash2 + --speculative-draft-attention-backend fa4；块大小由 draft checkpoint 推断，不设 --speculative-num-draft-tokens；draft 是稠密模型，需独立 attention backend。
   - 每个选项通过 disable 数组声明 DPAttention 开启或 AMD ROCm（mi300x/mi325x/mi355x）下的不可用原因。
2. 扩展 Playground 引擎（docs/src/snippets/_playground.jsx）：这是本 PR 唯一引擎逻辑改动，两点：
   - 在 speculative 轴的 deriveFromBase 过滤器与 apply 的 stripFlagsByFirstToken 列表补入 --speculative-adaptive，修复 Off 选项后残留孤立 flag、重选 EAGLE 时重复

emit 的问题；

- 新增可选 note 字段：渲染在当前生效选项的 chip 行下方（amber 样式），用于承载 " 命令能组合出但运行前需满足 " 的前置条件。当前无其他配置使用该字段。

3. 同步 cookbook 正文（docs/cookbook/autoregressive/GLM/GLM-5.3-Flash.mdx）：新增 ### Change the speculative algorithm 小节，解释三个算法 chip 的语义，并说明 DFlash2 的 build 前置条件（依赖 #36708）、draft 仓库访问需人工审批（gated: manual）、以及该组合尚未在 cookbook 硬件上实测。
4. 验证方式：node docs/scripts/check_cookbook_configs.mjs 通过；模拟引擎的 overlayCompose -> deriveFromBase -> apply 全链路覆盖 15 个 cell x 4 个 chip（9 种硬件 x 两种策略，AMD 仅 High Throughput），确认任意组合命令合法；修复前同 sweep 报告 12 处孤儿 / 重复 flag，全部指向 --speculative-adaptive。无自动化测试文件随附（文档型改动）。

docs/src/snippets/_playground.jsx

Playground 引擎核心文件：新增 note 字段的解析 / 渲染，speculative 轴 deriveFromBase 过滤与 apply strip 列表补入 --speculative-adaptive，修复 Off 选项的孤儿 / 重复 flag，是本 PR 中唯一有通用引擎逻辑改动的文件。

```
// speculative 轴：单选。current = 保持 base 不动，off = 剥离（greedy），
// 其他选项 = 剥离 base 的 --speculative-* 后拼接 option.flags。
// 该轴还会渲染选项自带的可选 `note`（前置条件提示），见下方 render 部分。
speculative: {
  initState: () => "current",

  // 把 base 单元格里出现的 --speculative-* flag 集合与每个选项预设比对：
  // 完全没有 → "off"；有一些但无预设匹配 → "current"。
  deriveFromBase: (cell, fc) => {
    const flags = (cell && cell.flags) || [];
    const baseSpec = flags.filter((f) => {
      const head = f.split(/[\s=]/)[0];
      return head === "--speculative-algorithm"
          || head === "--speculative-num-steps"
          || head === "--speculative-eagle-topk"
          || head === "--speculative-num-draft-tokens"
          // 自适应草稿深度是 EAGLE 预设的一部分，不是独立旋钮：
          // base 携带它时，切换算法必须一起剥离，否则 flag 残留且
          // 只有 EAGLE/EAGLE3 会识别它，服务端会告警。
          || head === "--speculative-adaptive"
          || head === "--speculative-dspark-block-size"
          || head === "--enable-linear-replayssm-spec"
          || head === "--linear-replayssm-cache-len"
          || head === "--speculative-ngram-max-bfs-breadth";
    });
    if (baseSpec.length === 0) return "off";
    for (const opt of (fc.options || [])) {
      if (!opt.flags || opt.flags.length !== baseSpec.length) continue;
      const ok = opt.flags.every((pf) => baseSpec.includes(pf));
```

```
      if (ok) return opt.id;
    }
    return "current";
  },

  apply: ({ flags, value, fc, sel, h, derived }) => {
    if (value === "current") return { flags };
    // 与 base 已一致时保持原位，避免 flag 顺序抖动产生假 diff。
    if (derived && value === derived) return { flags };
    const picked = (fc.options || []).find((p) => p.id === value);
    if (picked && h.evaluateChip(picked, {
      ...sel,
      dpAttnOn: h.hasFlag(flags, "--enable-dp-attention"),
    }).disabled) {
      // 被禁用的 chip 不允许泄漏进命令，直接保持 base 不动。
      return { flags };
    }
    // 剥离全部旧 speculative flag，再拼接所选预设，避免孤儿 /
    // 重复 flag（修复前 --speculative-adaptive 会残留或重复 emit）。
    flags = h.stripFlagsByFirstToken(flags, [
      "--speculative-algorithm", "--speculative-num-steps",
      "--speculative-eagle-topk", "--speculative-num-draft-tokens",
      "--speculative-adaptive",
      "--speculative-dspark-block-size",
      "--enable-linear-replayssm-spec",
      "--linear-replayssm-cache-len",
      "--speculative-ngram-max-bfs-breadth",
    ]);
    // ... 随后把 picked.flags 拼入并返回
  },
}

// 选项上可带可选 note：命令能组合出来、但运行前读者还需满足的前置条件
// （例如依赖尚未合入 pin 定镜像的代码）。只在该选项为当前生效选项时渲染，
// 卡片保持 chip 行形态，直到这个选择真正需要读者行动。
const note = (visible.find((c) => c.value === display) || {}).note;
return (
  <div key={axisId} style={s.card}>
    <div style={s.compactRow}>title + chips</div>
    {note && <div style={s.axisNote}>{note}</div>}
  </div>
);
```

docs/src/snippets/configs/zai-org/glm-5.3-flash.jsx

GLM-5.3-Flash 专属配置：新增 speculative 轴 4 个选项，声明 EAGLE 与 DFlash2 的 flags、禁用条件（DP-Attention / AMD ROCm）及 DFlash2 的 note 前置条件说明，是本 PR 的功能落地文件。

```
// ----- Card: "Speculative" -----
```

```
// Deploy 面板只能通过 Strategy 维度选择投机解码（Low Latency = adaptive MTP,
// High Throughput = off），本卡片提供更细粒度控制，并补上没有任何 cell 携带的
// DFlash2（其 draft 是独立 checkpoint）。
// EAGLE 预设与 Low Latency cells 字节一致，故 Low Latency base 会派生到该 chip
// （而非显示 Inherited from base），重复点选为无操作，不产生假 diff。
speculative: {
  options: [
    { id: "current", label: "Inherited from base" },
    { id: "off", label: "Off (greedy)" },
    {
      id: "eagle",
      label: "EAGLE / Adaptive MTP 5-1-6",
      flags: [
        "--speculative-algorithm EAGLE",
        "--speculative-num-steps 5",
        "--speculative-eagle-topk 1",
        "--speculative-num-draft-tokens 6",
        "--speculative-adaptive",
      ],
      disable: [
        {
          when: { dpAttnOn: [true] },
          reason: "Adaptive MTP does not support DP-Attention — the server falls back to a static
          draft depth and warns. Turn DP-Attention off in the Attention card above.",
        },
        {
          when: { hw: ["mi300x", "mi325x", "mi355x"] },
          reason: "MTP speculative decoding has not been validated for GLM-5.3-Flash on AMD
          ROCm; the Strategy row disables Low Latency there for the same reason.",
        },
      ],
    },
    {
      id: "dflash",
      label: "DFlash2",
      // Block-wise draft：块大小从 draft checkpoint 推断，所以不设
      // --speculative-num-draft-tokens。draft 是稠密模型，不能跑在
      // 目标模型的 DSA 后端上，因此必须显式指定 draft attention backend。
      flags: [
        "--speculative-algorithm DFLASH",
        "--speculative-draft-model-path incoai/GLM-5.3-Flash-DFlash2",
        "--speculative-draft-attention-backend fa4",
      ],
      // DFLASH 需要该模型的 hidden-state capture（#36708），目前合入的是
      // #36507 的 xinyuan/glm-5.3-flash-support 分支而非 main，所以 Install
      // 手风琴 pin 定的镜像不含该能力。待 #36507 合入并发布新镜像后删除本条。
      note: "⚠🤔 Needs the GLM-5.3-Flash hidden-state capture from PR #36708. It is merged
      into the PR #36507 support branch (xinyuan/glm-5.3-flash-support), not into main, so pull
      that branch at its current head — or add #36708's commit on top of an older checkout —
```

```
        before serving. The lmsysorg/sglang:glm-5.3-flash image alone is not enough.",
        disable: [
          {
            when: { dpAttnOn: [true] },
            reason: "DFLASH speculative decoding does not support DP-Attention — the server
            rejects the combination at startup. Turn DP-Attention off in the Attention card above.",
          },
          {
            when: { hw: ["mi300x", "mi325x", "mi355x"] },
            reason: "DFLASH speculative decoding only supports CUDA and NPU devices; the server
            rejects it on ROCm at startup.",
          },
        ],
      },
    ],
  },
```

评论区精华

Review 正式评论为 0（仅 mintlify bot 的 preview 部署通知），wisclmy0611 直接 APPROVED。技术讨论集中在 PR body 的自述中：

> "The speculative axis's flag family did not include --speculative-adaptive... Picking Off (greedy) left it behind as an orphan (--speculative-adaptive with no algorithm), and picking a preset that re-emits it produced the flag twice."
>
> "The speculative card had nowhere to say that: renderChip only surfaces a tooltip while a chip is disabled, and DFlash2 is selectable. So a speculative option may now carry an optional note..."
>
> "Left out here because completing the family changes which chip renders as active on ~8 other model pages, which deserves its own review."

风险与影响

- DFlash2 依赖未合入 main 的能力：note 明示读者需使用 xinyuan/glm-5.3-flash-support 分支或手动叠加 #36708 的 commit，否则命令无法启动。该 note 是临时方案，需在 #36507 合入并发布镜像后及时清理，否则文档会长期保留过时指引。
- 引擎改动影响面有限但有遗留：--speculative-adaptive 修复经验证对全站 15 cell x 4 chip 组合有效，但 speculative 家族仍缺 8 个 flag，其他页面（如 rednote/dots3-note.jsx、Qwen/qwen3.8-27b.jsx）的孤儿 / 重复 flag 问题依旧存在，属于已知未解决项。
- 缺少自动化测试：配置组合正确性依赖手工 sweep 验证，后续如需长期演进，建议为 _playground.jsx 的 deriveFromBase/apply 链路补充单元测试。

影响范围主要在文档站点：GLM-5.3-Flash 页面获得细粒度投机解码控制能力，同时 _playground.jsx 的 note 机制成为全站 cookbook 可复用的新渲染能力。团队侧需跟踪 #36507 合入进度并在随后清理 note。

关联脉络

本 PR 是 GLM-5.3-Flash cookbook 演进的一部分，与近期 #36544（GLM-5.3-Flash cookbook 的 HiCache / EAGLE / DCP4 配置更新）属于同一文档产品线。它同时暴露出 Playground 引擎 speculative flag 家族完成度不足的系统性问题——这正是近期 #36620、#36621、#36725 等 config 系列重构（统一配置解析、让 handler 显式声明决策）所关注的 " 配置派生一致性 " 主题在文档站的延伸。后续值得跟踪两个方向：#36507 分支合入主线的进度，以及 speculative flag 家族补全的独立 PR。