

# PR #49587 完整报告

vllm-project/vllm

[Docs] Use `gen-files` for generated docs content

合并时间: 2026-07-25 22:43

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/49587>

## 执行摘要

- 一句话: 统一文档生成方式为 gen-files
- 推荐动作: 值得精读, 尤其是 generated\_content.py 的设计和 fill\_markers 的严格校验理念; 涉及文档自动化的团队可学习 gen-files 插件的使用模式。

## 功能与动机

PR body 指出之前文档生成方法混杂 (mkdocs hooks + pre-commit hooks), 维护困难。通过统一到 gen-files, 简化构建流程, 减少 git 跟踪的生成内容, 并使 CLI 帮助文本与文档同步。

## 实现拆解

实现拆解分以下步骤:

1. 新增文档生成核心库 generated\_content.py, 提供 fill\_markers 函数, 支持在文档源页中通过 `--8<-- "gen:<key>"` 标记插入生成内容, 避免 pymdownx.snippets 的隐式行为。
2. 将原本在 docs/mkdocs/hooks/ 下的 generate\_argparse.py、generate\_metrics.py、generate\_examples.py 重命名搬迁到 docs/mkdocs/gen\_files/, 并改造成 gen-files 脚本: 去掉 on\_startup 函数封装, 直接在模块作用域执行生成逻辑; 将内容写入替换为 mkdocs\_gen\_files.open; 从写独立文件改为调用 fill\_markers 填充标记。
3. 将原本作为 pre-commit 钩子的 tools/pre\_commit/generate\_attention\_backend\_docs.py 搬迁到 docs/mkdocs/gen\_files/generate\_attention\_backends.py, 移除 is\_relevant\_file 和 RELEVANT\_PATTERNS 等预提交逻辑, 改用 fill\_markers 将生成的表格嵌入 docs/design/attention\_backends.md。
4. 修改 docs/mkdocs/hooks/url\_schemes.py, 增加 replace\_docs\_link 方法, 将 docs.vllm.ai 绝对链接重写为相对链接, 并处理 vllm.config.<Class> API 引用转换为 mkdocstrings 交叉引用。
5. 调整 mkdocs.yaml 配置, 注册 gen-files 插件并移除不再需要的生成文件路径; 同步删除旧的 .inc.md 文件和 pre-commit 钩子配置。
6. 更新 vllm/entrypoints/cli/ 下的入口文件 (mm\_processor.py、launch.py), 添加 DESCRIPTION 和子命令注册, 使 CLI 帮助文本完整, 以便 gen-files 能生成准确的 CLI 参考。

关键文件:

- docs/mkdocs/gen\_files/generate\_argparse.py (模块 文档生成; 类别 source; 类型 rename-or-move; 符号 on\_startup, format\_help, linkify\_docs\_urls, child\_link) : 核心生成脚本, 重命名自 docs/mkdocs/hooks/generate\_argparse.py, 改造成 gen-files 脚本, 负责生成 CLI 参考页面。
- docs/mkdocs/gen\_files/generated\_content.py (模块 文档生成; 类别 source; 类型 dependency-wiring; 符号 fill\_markers) : 新增核心模块, 提供 fill\_markers 函数, 是统一生成机制的基石。
- docs/mkdocs/hooks/url\_schemes.py (模块 文档生成; 类别 source; 类型 core-logic; 符号 replace\_docs\_link) : 扩展 URL 方案预处理, 新增 docs.vllm.ai 绝对链接重写和 API 引用转换, 确保生成页面中链接正确。
- docs/mkdocs/gen\_files/generate\_attention\_backends.py (模块 文档生成; 类别 source; 类型 rename-or-move; 符号 generate\_markdown\_table, generate\_usage\_section, generate\_priority\_section, \_priority\_block) : 重命名自 tools/pre\_commit/generate\_attention\_backend\_docs.py, 从 pre-commit 钩子改为 gen-files 脚本, 使用 fill\_markers 生成 attention backends 特性表格。
- docs/mkdocs/gen\_files/generate\_metrics.py (模块 文档生成; 类别 source; 类型 rename-or-move; 符号 on\_startup) : 重命名自 docs/mkdocs/hooks/generate\_metrics.py, 改为 gen-files 脚本, 使用 fill\_markers 将指标表格嵌入使用页面。

关键符号: fill\_markers, replace\_docs\_link, format\_help, linkify\_docs\_urls, child\_link, generate\_markdown\_table, generate\_usage\_section, generate\_priority\_section, generate\_legend, \_feature\_table, generate\_mla\_section, on\_startup

## 关键源码片段

### docs/mkdocs/gen\_files/generated\_content.py

新增核心模块, 提供 fill\_markers 函数, 是统一生成机制的基石。

```
# SPDX-License-Identifier: Apache-2.0
# SPDX-FileCopyrightText: Copyright contributors to the vLLM project
"""
```

Inline build-time generated content into existing docs pages.

Source pages mark where generated content goes with a snippet-style marker, `--8<-- \ngen:<key>\``, so the insertion point is explicit and readable. The substitution happens here (at gen-files time, before mkdocs-gen-files shadows the page), not via pymdownx.snippets, so the content can be generated at build time without living in a real file on disk.

The `gen:` prefix keeps these markers distinct from real pymdownx.snippets includes, and `fill\_markers` fails loudly if a marker is missing or left behind (pymdownx.snippets would otherwise silently drop an unsubstituted marker).

```
"""
```

```
from pathlib import Path
```

```

import mkdocs_gen_files
import regex as re

DOCS_DIR = Path(__file__).parent.parent.parent
_MARKER = '--8<-- "gen:{key}"'
_ANY_MARKER = re.compile(r'--8<-- "gen:[^"]*"')

def fill_markers(doc_path: str, blocks: dict[str, str]) -> None:
    """Replace `--8<-- "gen:<key">` markers in a docs page with generated content.

    Args:
        doc_path: Docs-relative path of the source page to fill.
        blocks: Mapping of marker key to the markdown to insert in its place.

    Raises:
        FileNotFoundError: If the source page does not exist.
        ValueError: If an expected marker is missing, or any `gen:` marker
            is left unsubstituted after filling.
    """
    source = DOCS_DIR / doc_path
    if not source.exists():
        raise FileNotFoundError(
            f"Cannot fill markers in missing page: {doc_path}"
        )

    text = source.read_text()
    # 逐个替换标记为对应的生成内容
    for key, content in blocks.items():
        marker = _MARKER.format(key=key)
        if marker not in text:
            raise ValueError(
                f"{doc_path}: missing marker {marker}"
            )
        text = text.replace(marker, content)

    # 严格检查是否还有未替换的 gen: 标记
    if leftover := _ANY_MARKER.search(text):
        raise ValueError(
            f"{doc_path}: unsubstituted marker {leftover.group()}"
        )

    # 通过 mkdocs_gen_files 写入虚拟文件
    with mkdocs_gen_files.open(doc_path, "w") as f:
        f.write(text)
    # 让编辑按钮指向真实源页（而不是虚拟路径）
    mkdocs_gen_files.set_edit_path(doc_path, doc_path)

```

[docs/mkdocs/hooks/url\\_schemes.py](#)

扩展 URL 方案预处理，新增 docs.vllm.ai 绝对链接重写和 API 引用转换，确保生成页面中链接正确。

```
class UrlSchemesPreprocessor(Preprocessor):
    def run(self, lines):
        page = self.ext.page
        files = self.ext.files
        if page is None:
            return lines

        # 定义替换函数
        def replace_docs_link(match: re.Match) -> str:
            """Rewrite absolute docs.vllm.ai links as doc-relative links."""
            title = match.group("title")
            path = match.group("path").rstrip("/")
            fragment = match.group("fragment") or ""

            # vllm.config.<Class> API reference -> mkdocstrings cross-reference
            if path == "api/vllm/config" and re.fullmatch(
                r"#vllm\.\config\.\w+", fragment
            ):
                ident = fragment[1:]
                return f"~{ident}~[{ident}]"

            # 其他文档页 -> 相对于当前页面的链接
            src = f"{path.removesuffix('.html')}.md"
            if files.get_file_from_path(src) is None:
                return match.group(0) # 未知页面，保留原链接

            rel = posixpath.relpath(src, posixpath.dirname(page.file.src_uri))
            if title.startswith("http"):
                title = path.rstrip("/").split("/")[-1].replace("-", " ").title()
                title += fragment.replace("#", " § ").replace("-", " ").title()

            return f"[{title}]({rel}{fragment})"

        # 对每行应用替换
        lines = [docs_link.sub(replace_docs_link, line) for line in lines]
        # ... 其他替换 ...
        return lines
```

## 评论区精华

PR 得到 1 个 approve 评论（来自 DarkLight1337），无其他讨论或争议。

- 暂无高价值评论线程

## 风险与影响

- 风险：风险较低。主要影响文档构建流程，gen-files 插件配置错误可能导致文档缺失或构建失败。但作者已验证预览页面（metrics、engine\_args、attention\_backends、CLI 均正常）。不涉及运行时逻辑，对用户无影响。fill\_markers 对标记严格校验，若源文档中标记丢失或残留会报错，但这也避免了静默缺失。
- 影响：对用户：文档浏览体验不变，但生成内容（如表、参数列表）保持与代码同步更可靠。对系统：不再需要在 git 中跟踪 docs/generated/ 下的文件，构建时不需要 pre-commit 钩子。对团队：文档生成逻辑统一在 docs/mkdocs/gen\_files/ 下，维护更简单；添加新生成页面时可复用 generated\_content.py 的 fill\_markers。
- 风险标记：文档构建变更，无运行时影响，构建失败风险低

## 关联脉络

- 暂无明显关联 PR