

PR #23312 完整报告

sgl-project/sglang

Docs/url redirect

合并时间: 2026-04-21 12:26

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/23312>

执行摘要

- 一句话: 新增文档重定向配置和生成脚本, 确保旧版 Sphinx URL 能正确跳转到新版 Mintlify 站点。
- 推荐动作: 对于文档维护者, 建议精读 `gen_redirects.py` 脚本以理解映射规则和自动化逻辑; 对于其他工程师, 此 PR 可作为文档迁移基础设施的参考案例。

功能与动机

根据 PR body 描述, 切换文档部署到 Mintlify 后, 旧版 URL (如 `/<section>/<page>.html`) 会 404, 而外部链接、博客和搜索结果均依赖这些旧链接。因此需要添加同域重定向来避免大规模链接断裂。

实现拆解

1. 新增生成脚本: 在 `docs_new/scripts/gen_redirects.py` 中, 定义了映射规则 (如目录重命名 `SECTION_RENAMERS` 和显式覆盖 `EXPLICIT`), 并实现核心函数 `old_url_from_path`、`new_url_for` 和 `list_new_urls`, 自动遍历旧文档路径并生成重定向列表。
2. 更新配置文件: 修改 `docs_new/docs.json`, 在 `redirects` 数组中添加 147 个重定向条目, 将旧 URL (带 `.html` 后缀) 映射到新 URL (无后缀, 带 `/docs/` 前缀)。
3. 脚本集成: 通过运行 `main()` 函数, 脚本验证目标 URL 是否存在并更新 `docs.json`, 确保重定向的准确性和可维护性。
4. 格式化和调整: 提交历史包括对脚本的格式化和可执行位调整, 以符合代码风格。

关键文件:

- `docs_new/scripts/gen_redirects.py` (模块 文档脚本; 类别 `source`; 类型 `core-logic`; 符号 `old_url_from_path`, `new_url_for`, `list_new_urls`, `main`): 核心脚本, 负责生成旧 Sphinx 路径到新 Mintlify 路径的映射, 确保重定向准确性和可维护性。
- `docs_new/docs.json` (模块 文档配置; 类别 `config`; 类型 `configuration`): Mintlify 文档站点的配置文件, 扩展了重定向列表, 直接决定了实际跳转行为。

关键符号: `old_url_from_path`, `new_url_for`, `list_new_urls`, `main`

关键源码片段

`docs_new/scripts/gen_redirects.py`

核心脚本，负责生成旧 Sphinx 路径到新 Mintlify 路径的映射，确保重定向准确性和可维护性。

```
#!/usr/bin/env python3
"""Generate Mintlify docs.json redirects from old Sphinx paths to new Mintlify paths."""

from __future__ import annotations

import json
import os
from pathlib import Path

REPO = Path(__file__).resolve().parent.parent.parent
OLD_DOCS = REPO / "docs"
NEW_DOCS = REPO / "docs_new" / "docs"

# Directory-level renames (old → new, under /docs/ prefix)
SECTION_RENAMES = {
    "get_started": "get-started",
    "platforms": "hardware-platforms",
    "supported_models": "supported-models",
    "diffusion": "sglang-diffusion",
}

# Explicit file-level mappings. Keys are old URL paths (no .html, with leading /).
# Values are new URL paths (with /docs/ prefix, no extension).
EXPLICIT = {
    "/get_started/install": "/docs/get-started/installation",
    # ... 更多显式映射，覆盖特定文件重命名
}

def old_url_from_path(old_path: Path) -> str:
    """Convert a file path under docs/ to the old Sphinx URL."""
    relative = old_path.relative_to(OLD_DOCS)
    url = "/" + str(relative.with_suffix("")).replace("\\", "/")
    if url.endswith("/index"):
        url = url[:-6] # 移除 /index 后缀
    return url + ".html" # 添加 .html 后缀以匹配旧 URL 格式

def new_url_for(old_url: str) -> str:
    """Map an old URL to a new Mintlify URL."""
    # 优先检查显式映射
    if old_url in EXPLICIT:
        return EXPLICIT[old_url]
    # 应用目录重命名规则
    parts = old_url.strip("/").split("/")
    if parts[0] in SECTION_RENAMES:
        parts[0] = SECTION_RENAMES[parts[0]]
    new_url = "/docs/" + "/".join(parts)
    return new_url.rstrip(".html") # 移除 .html 后缀以匹配新格式
```

```

def list_new_urls() -> set[str]:
    """Collect all existing file paths under docs_new/docs."""
    urls = set()
    for ext in (".mdx", ".ipynb", ".md"):
        for path in NEW_DOCS.rglob(f"*{ext}"):
            relative = path.relative_to(NEW_DOCS)
            url = "/docs/" + str(relative.with_suffix("")).replace("\\", "/")
            urls.add(url)
    return urls

def main():
    """Generate redirects and update docs.json."""
    redirects = []
    for old_path in OLD_DOCS.rglob("*.html"):
        old_url = old_url_from_path(old_path)
        new_url = new_url_for(old_url)
        redirects.append({"source": old_url, "destination": new_url})

    # 验证目标 URL 是否存在
    existing_urls = list_new_urls()
    for r in redirects:
        if r["destination"] not in existing_urls:
            print(f"Warning: destination {r['destination']} does not exist")

    # 更新 docs.json
    docs_json_path = REPO / "docs_new" / "docs.json"
    with open(docs_json_path, "r") as f:
        config = json.load(f)
    config["redirects"] = redirects
    with open(docs_json_path, "w") as f:
        json.dump(config, f, indent=2)
    print(f"Updated {len(redirects)} redirects in docs.json")

if __name__ == "__main__":
    main()

```

评论区精华

由于 review 评论为空，本 PR 在合并前未经过详细技术讨论，没有争议点或设计权衡记录。

- 暂无高价值评论线程

风险与影响

- 风险：主要风险包括：
 - 配置错误：重定向条目可能遗漏某些旧路径或指向不存在的目标，导致用户访问 404。
 - 脚本依赖：生成脚本依赖于文件系统遍历，若文档结构未来变化，脚本可能失效或需要更新。

- 重定向循环：配置不当可能引发无限重定向，影响用户体验和 SEO。
- 影响：对用户：确保旧书签、外部链接和搜索结果能无缝跳转，提升文档可访问性和用户体验。对系统：仅增加文档站点的重定向配置，无核心功能、性能或安全影响。对团队：提供了自动化工具，便于未来文档结构变更时的重定向维护。
- 风险标记：重定向配置错误，链接覆盖不全

关联脉络

- PR #23001 Add new Mintlify documentation site (docs_new/): 同属文档迁移工作，本 PR 的重定向配置是为了配合该 PR 新增的 Mintlify 文档站点，确保旧链接的兼容性。