PR #38848 完整报告

vllm-project/vllm

[Bugfix] Fix Qwen3 tool parser for Responses API tools

合并时间：2026-04-08 10:08

原文链接：http://prhub.com.cn/vllm-project/vllm/pull/38848

执行摘要

本次 PR 修复了 Qwen3 工具解析器在 Responses API 中无法正确处理 FunctionTool 扁平结构的问题，通过统一工具属性查找逻辑，确保参数类型正确解析，提升了工具调用的正确性和数据格式一致性。

功能与动机

Qwen3 工具解析器（包括 Qwen3CoderToolParser 和 Qwen3XMLToolParser）原先假设所有工具都有 .function.name 和 .function.parameters 结构（对应 ChatCompletionToolsParam 类型），但 Responses API 的 FunctionTool 对象使用扁平结构（.name 和 .parameters 直接定义），导致参数类型查找失败，所有值被错误地返回为字符串。如 PR body 所述，修复前 dimensions 参数返回字符串化的 JSON，修复后正确返回对象。

实现拆解

实现方案围绕统一工具属性查找逻辑展开：

1. 共享工具函数：在 vllm/tool_parsers/utils.py 新增 find_tool_properties 函数，委托给现有的 _extract_tool_info 处理两种工具类型，返回属性配置。
2. 解析器重构：在 vllm/tool_parsers/qwen3coder_tool_parser.py 和 qwen3xml_tool_parser.py 中移除冗余的 _get_arguments_config 和 _get_param_type 方法，改为调用新函数。
3. 测试更新：在 tests/tool_parsers/test_qwen3coder_tool_parser.py 中更新测试 fixture，支持 ChatCompletionToolsParam 和 FunctionTool 两种工具类型的参数化测试。

关键代码示例（来自 utils.py）：

```
def find_tool_properties(
    tools: list[Tool] | None,
    tool_name: str,
) -> dict[str, Any]:
    """Find a tool by name and return its properties dict, or {}."""
    if not tools:
        return {}
    for tool in tools:
        name, params = _extract_tool_info(tool)
        if name == tool_name:
```

```
            return (params or {}).get("properties", {})
    return {}
```

评论区精华

Review 讨论聚焦于两个核心点：

1. 正确性权衡：gemini-code-assist[bot] 指出 find_tool_properties 缺少回退逻辑，可能破坏非标准工具定义的兼容性；sfeng33 反驳称 > "This is not a concern because OpenAPI/JSON Schema spec requires tool parameters to be an object schema with a 'properties' key." 基于规范假设，此疑虑未完全解决但 PR 被批准。
2. 测试覆盖：gemini-code-assist[bot] 指出测试 fixture 只包含 'xml' 参数，导致 Qwen3CoderToolParser 未被测试；sfeng33 回应 > "This is technically correct, but out of scope for this PR." 测试覆盖问题被视为未解决。

风险与影响

- 技术风险：find_tool_properties 依赖 OpenAPI/JSON Schema 规范，若工具定义缺少 'properties' 键，参数类型转换可能失败；测试覆盖率不足增加回归风险；核心解析逻辑修改可能影响流式和并行工具调用场景。
- 影响分析：直接影响使用 Responses API 与 Qwen3 模型进行工具调用的用户，修复参数类型错误提升正确性；系统层面统一工具查找逻辑，减少代码冗余，增强可维护性。

关联脉络

- 与 PR #38860（修复 tool-calling 和 responses-api 参数传递）紧密相关，同属 tool-calling 模块的 bugfix 系列。
- 与 PR #38755（迁移 Responses API 流式解析到统一解析器）一脉相承，显示仓库正推进统一解析器架构，以减少重复代码和提升兼容性。 结合历史 PR，可见仓库在 tool-calling 和 responses-api 领域持续优化，注重规范遵循和模块化设计。