

PR #2125 完整报告

THUDM/slime

feat(coding_agent_rl): select claude_code/codex harness+adapter pair via SWE_AGENT

合并时间: 2026-06-23 14:57

原文链接: <http://prhub.com.cn/THUDM/slime/pull/2125>

执行摘要

- 一句话: 新增 SWE_AGENT 环境变量支持选择 agent 实现
- 推荐动作: 值得精读。该 PR 展示了一种简单而优雅的 agent 选择模式 (注册表 + 环境变量注入), 符合开闭原则, 便于后续扩展。对于其他需要多后端 / 多实现的场景有借鉴意义。

功能与动机

之前 `generate.py` 在文档字符串和代码中硬编码了 `ClaudeCodeHarness` + `AnthropicAdapter`, 当需要切换为 Codex 的 `CodexHarness` + `OpenAIAdapter` 时必须修改源码。PR body 明确说明目标是 "Make the SWE rollout agent-agnostic", 通过环境变量注入实现 agent 选择, 无需修改生成代码即可支持多 agent 实验。

实现拆解

1. 引入 agent 注册表: 在 `examples/coding_agent_rl/generate.py` 模块顶部新增 `_AGENTS` 字典, 将 agent 名称映射为 (harness类, adapter类) 元组: `claude_code` 对应 (`ClaudeCodeHarness`, `AnthropicAdapter`), `codex` 对应 (`CodexHarness`, `OpenAIAdapter`)。
2. 环境变量解析与校验: 通过 `os.environ.get("SWE_AGENT", "claude_code")` 读取配置, 若值不在 `_AGENTS` 中则抛出 `ValueError`, 给出清晰提示。成功则解包为 `HARNESS_CLS` 和 `ADAPTER_CLS` 两个模块级常量。
3. 替换硬编码引用: 将原来所有直接引用 `ClaudeCodeHarness` 和 `AnthropicAdapter` 的地方 (如 `boot_agent_sandbox` 中的 `await ClaudeCodeHarness().install_cli(cand)`) 替换为 `HARNESS_CLS().install_cli(cand)` 等, 文档字符串也相应更新。
4. Shell 脚本配套: 在 `run_qwen36_35b_a3b_swe_8nodes.sh` 中增加 `export SWE_AGENT="${SWE_AGENT:-claude_code}"` 和将该变量加入 `RUNTIME_ENV_JSON`, 使 Ray workers 能获取相同配置。

关键文件:

- `examples/coding_agent_rl/generate.py` (模块 `coding_agent_rl`; 类别 `source`; 类型 `dependency-wiring`): 核心变更文件, 引入 agent 注册表并替换所有硬编码引用, 是 agent 选择机制的主逻辑所在。
- `examples/coding_agent_rl/run_qwen36_35b_a3b_swe_8nodes.sh` (模块 `coding_agent_rl`; 类别 `other`; 类型 `configuration`): Shell 脚本配套修改: 导出

SWE_AGENT 环境变量并加入 RUNTIME_ENV_JSON，确保 Ray 工作节点能正确继承选择。

关键符号：boot_agent_sandbox

关键源码片段

examples/coding_agent_rl/generate.py

核心变更文件，引入 agent 注册表并替换所有硬编码引用，是 agent 选择机制的主逻辑所在。

```
"""Coding-Agent RL: per-sample generate() function for slime.

--custom-generate-function-path examples.coding_agent_rl.generate.generate

generate() is a four-stage orchestrator: swe.prepare_workspace + harness.run
-> swe.git_diff -> swe.evaluate -> adapter.finish_session. The (harness, adapter)
pair is chosen by the SWE_AGENT env var (claude_code | codex); see _AGENTS below.
...
"""

from slime.agent.adapters import AnthropicAdapter, OpenAIAdapter
from slime.agent.harness import ClaudeCodeHarness, CodexHarness
# ...

# Agent 注册表：将 agent 名称映射为 (harness 类, adapter 类) 的不可变元组
_AGENTS = {
    "claude_code": (ClaudeCodeHarness, AnthropicAdapter), # 默认 agent
    "codex": (CodexHarness, OpenAIAdapter), # Codex agent
}

# 从环境变量读取 SWE_AGENT，默认使用 claude_code
AGENT_NAME = os.environ.get("SWE_AGENT", "claude_code")
if AGENT_NAME not in _AGENTS:
    raise ValueError(f"SWE_AGENT={AGENT_NAME!r} not in {sorted(_AGENTS)}")
HARNESS_CLS, ADAPTER_CLS = _AGENTS[AGENT_NAME]

# 后续所有使用 harness / adapter 的地方都改为引用模块级常量 HARNESS_CLS / ADAPTER_CLS
async def boot_agent_sandbox(image: str, instance_id: str) -> AsyncIterator[E2BSandbox]:
    """Boot 一个全新的 E2B sandbox 并安装所选 harness 工具链。"""
    # ... 省略重试逻辑
    await HARNESS_CLS().install_cli(cand) # 原来是 ClaudeCodeHarness().install_cli(cand)
    # ...
```

examples/coding_agent_rl/run_qwen36_35b_a3b_swe_8nodes.sh

Shell 脚本配套修改：导出 SWE_AGENT 环境变量并加入 RUNTIME_ENV_JSON，确保 Ray 工作节点能正确继承选择。

```
# ===== SWE / claude-code rollout knobs =====
export SWE_AGENT="${SWE_AGENT:-claude_code}" # 新增：默认 claude_code，后续可被覆盖
# ... 其他导出
```

```
# 构建 RUNTIME_ENV_JSON 时包含 SWE_AGENT 变量
RUNTIME_ENV_JSON=$(python3 - <<PY
import json, os
keys = (
    "no_proxy", "NO_PROXY",
    "SWE_AGENT",          # 新增: 确保 worker 也使用相同 agent
    "E2B_API_KEY",
    # ...
)
# 构建包含这些环境变量的 JSON
print(json.dumps({
    k: v for k, v in os.environ.items() if k in keys
}))
PY
)
```

评论区精华

该 PR 无 review 评论，讨论高亮无。

- 暂无高价值评论线程

风险与影响

- 风险：无重大风险。默认值 `claude_code` 保持向后兼容。唯一风险是当 `SWE_AGENT` 被错误设置时，`ValueError` 会在模块导入时立即抛出，而非延迟到运行时，可能中断启动流程；但这是有意的防护性设计。此外，新增的 `_AGENTS` 注册表属于模块级 mutable 对象，在多线程或多进程场景下需要小心修改，但目前是常量，安全。
- 影响：用户 / 开发者：通过设置环境变量 `SWE_AGENT=codex` 即可切换 agent，无需改代码，降低了实验成本。系统：仅影响 `examples/coding_agent_rl` 下的 rollout 流程，不影响其他模块。团队：新增了一种可扩展的 agent 注册模式，后续添加新 agent 只需在 `_AGENTS` 字典中增加条目并确保对应 harness 和 adapter 已实现。
- 风险标记：模块级常量由环境变量决定，启动时即确定

关联脉络

- PR #2005 [coding-agent-rl] Refactor coding-agent RL: turn-node TrajectoryManager + pluggable harness layer: 同一功能线 (`coding_agent_rl`)，该 PR 重构了 harness 和 adapter 基础设施，本 PR 基于此新增 agent 选择机制。