

PR #2136 完整报告

radixark/miles

openenv/tbench2: drop the daytona bake CLI, which nothing can consume

合并时间: 2026-08-04 23:36

原文链接: <http://prhub.com.cn/radixark/miles/pull/2136>

执行摘要

- 一句话: 删除无人消费的 daytona bake 快照 CLI
- 推荐动作: 建议快速浏览而非精读: 变更单文件且为纯删除, 无新逻辑。值得借鉴的是 PR body 的“可消费性论证”——从 API 参数类型不匹配 (CreateSandboxFromImageParams 与 CreateSandboxFromSnapshotParams)、资源配额约束 (org 500 上限)、全局 grep 检索三层证明死代码无可消费方, 这种删除理由的严谨性可复用于同类清理。合并后应复核 #1913 的 rebase 约定是否落地, 并在 openenv 目录内搜索残留的 bake 或 snapshot_name 引用。

功能与动机

PR body 明确指出: per-task sandbox 重构让 daytona leg 变成声明式——每次 episode 从 Image 定义创建, 定义哈希就是缓存键; 而 bake CLI 注册的是命名快照, create 路径永远无法引用它, 因为 create_task_sandbox() 传递 CreateSandboxFromImageParams(image=...) , 消费快照需要另一参数类型 CreateSandboxFromSnapshotParams(snapshot=...) (daytona SDK 0.198.0) 。并且“避开 org 级快照配额是声明式路径存在的理由, 共享 org 已距离 500 上限不足 2 个快照”, 所以未来也不会改走快照路径。作者通过 grep 验证仓库内无调用者、README 与各远程分支 runbook 无文档化, 确认该 CLI 是 #1710 引入声明式路径时遗留的 pool 时代死代码, 只会烧配额并要求活跃 API key。

实现拆解

1. 删除入口与 CLI 主体: 在 examples/experimental/openenv/tb2_sandbox_daytona.py 中删除 snapshot_name()、bake()、main() 三个函数及文件末尾的 if __name__ == "__main__": sys.exit(main()) 守卫。bake() 原本通过 daytona SDK 的 CreateSnapshotParams 注册命名快照, 并处理 force 重删、日志输出与 1800 秒超时。
2. 清理专用依赖: 删除仅服务于 CLI 的 import argparse、re、sys; 保留 getpass、os、shlex、threading、time 与来自 tb2_sandbox_recipe 的 read_task_config、resolve_docker_image、server_cmd、server_layer_commands、wait_server_ready——它们仍被 create_task_sandbox()、resolve_api_key()、keepalive 线程等声明式路径使用。此处是 dependency-wiring 式清理, 不触碰保留路径的调用关系。
3. 文档同步: 模块 docstring 删除 bake CLI 描述, 新增“*There is deliberately no bake step here*”一节, 明确“image definition 就是缓存键, create 要么命中构建缓存要么预热它, 预注册快照只会花掉声明式路径本想避开的 org 配额”; make_daytona() 的 docstring 也

去掉“the same way this module's own CLI is”措辞。

4. 验证与测试：本地执行 `pytest examples/experimental/openenv/tests/` 得 21 passed、1 skipped（该套件在仓库级 testpaths 之外，CI 不运行）；作者通过 `grep` 验证仓库内唯一 import 该模块的 `openenv_daytona_agent_function.py` 只用 `make_daytona + create_task_sandbox`，README 与各远程分支 runbook 均无 `bake` 引用。
5. 合并顺序约定：与开放 PR #1913 存在 hunk 级冲突——#1913 重写被删除的 `main()` 参数解析为共享 `recipe.add_task_selection_args`；两个 PR 谁先合入，另一个 rebase 时丢弃相应 hunk，且 #1913 的 helper docstring 需改为单数表述（e2b 成为唯一调用者）。

关键文件：

- `examples/experimental/openenv/tb2_sandbox_daytona.py`（模块 沙箱集成；类别 source；类型 dependency-wiring；符号 `snapshot_name`, `bake`, `main`）：唯一变更文件：删除无人消费的 `bake CLI`（`snapshot_name`、`bake`、`main`、`__main__` 守卫）及其专用导入 `argparse`、`re`、`sys`，并重写模块 docstring 说明该 provider 无 `bake` 步骤；保留声明式 `create` 路径、`make_daytona` 等核心符号，`daytona rollout leg` 行为不变。

关键符号：`snapshot_name`, `bake`, `main`, `make_daytona`

关键源码片段

`examples/experimental/openenv/tb2_sandbox_daytona.py`

唯一变更文件：删除无人消费的 `bake CLI`（`snapshot_name`、`bake`、`main`、`__main__` 守卫）及其专用导入 `argparse`、`re`、`sys`，并重写模块 docstring 说明该 provider 无 `bake` 步骤；保留声明式 `create` 路径、`make_daytona` 等核心符号，`daytona rollout leg` 行为不变。

```
"""Daytona materialization of the per-task Terminal-Bench-2 sandbox recipe.
```

```
The recipe itself — the shell layers that turn a task's official image into a combined task+env-server image — lives in ``tb2_sandbox_recipe`` (sibling module) and is provider-agnostic. This module is everything Daytona-specific about turning that recipe into a running cloud sandbox:
```

```
``create_task_sandbox(...)`` per-episode declarative create straight from the ``Image`` definition. Named snapshots count against an org-level quota, so registering one per task may not scale to a full task suite; the declarative path avoids the quota entirely, and repeat creates hit Daytona's build cache (~1min after the first build).
```

```
There is deliberately no bake step here: on this provider the image definition IS the cache key, so a create either hits the build cache or warms it, and nothing a create passes can name a pre-registered snapshot — registering one per task would only spend the org quota the declarative path exists to avoid.
```

```
"""
```

```
# 变更要点：删除 bake CLI（snapshot_name / bake / main / __main__ 守卫）后，  
# docstring 明确写出 " 没有 bake 步骤 "：因为 create 只传 CreateSandboxFromImageParams，  
# 快照消费参数（CreateSandboxFromSnapshotParams）在声明式路径上永远不可达，预注册  
# 只会消耗 org 级配额（共享 org 距 500 上限不足 2 个快照）。
```

```

import getpass
import os
import shlex
import threading
import time
from pathlib import Path

# 原仅服务于 bake CLI 的 argparse、re、sys 三个导入已同步移除；
# 以下导入仍被 create_task_sandbox() 与 keepalive 线程等保留路径使用。
from tb2_sandbox_recipe import (
    read_task_config,
    resolve_docker_image,
    server_cmd,
    server_layer_commands,
    wait_server_ready,
)

def make_daytona():
    """Daytona client: key from resolve_api_key(), endpoint from optional
    DAYTONA_API_URL. Public: callers driving create_task_sandbox() need a
    client configured this way.
    """
    # 注意：原 docstring 末尾的 "the same way this module's own CLI is"
    # 已随 bake CLI 删除而移除，因为模块不再自备命令行入口。
    from daytona import Daytona, DaytonaConfig

    return Daytona(
        DaytonaConfig(
            api_key=resolve_api_key(),
            api_url=os.getenv("DAYTONA_API_URL", "https://app.daytona.io/api"),
        )
    )

```

评论区精华

该 PR 没有实质 review 评论：review_comments_count = 0，唯一审核者 Shi-Dong 直接 APPROVED (LGTM!)；issue 评论只有 gemini-code-assist 的自动停用通知，无信息量。值得注意的“讨论”其实在 PR body 中：作者用三段论证（API 参数类型不匹配、org 快照配额约束、grep 全局检索）证明 bake CLI 无消费方，并主动预告了与开放 PR #1913 的 rebase 冲突及处理约定——这种把删除理由写进 body、供后续合并人参考的做法值得借鉴。

- 与 #1913 的 rebase 冲突预期 (other): 尚无结论，需按合并顺序协调；若 #1913 先合入，本 PR 的删除不受影响，仅需丢弃其对已删 main() 的改写。

风险与影响

- 风险：风险整体较低：删除的是无调用方的实验模块入口，主要风险点是外部手工执行 `python tb2_sandbox_daytona.py` 的脚本会失效（作者已 `grep` 确认仓库内不存在此类用法，但仓库外使用无法验证）。合并流程风险在于与 #1913 的 hunk 级冲突，需按约定顺序 `rebase`，否则可能出现误删或残留 `main()` 引用的半合并状态。此外该模块测试套件在仓库级 `testpaths` 之外、CI 不运行，删除缺乏自动化回归保障——不过删除的函数本身无测试覆盖，风险可接受。收益面：消除了误用 `bake` 导致 `org` 快照配额被消耗的隐患（共享 `org` 距 500 上限不足 2 个快照），并去掉模块中误导性的 `warm cache` 文档。
- 影响：影响范围限定在 `examples/experimental/openenv` 示例链路：
`tb2_sandbox_daytona.py` 净减 61 行，模块职责收窄为单一声明式创建路径；
`openenv_daytona_agent_function.py` 仅使用 `make_daytona` 与 `create_task_sandbox`，行为不受影响。对 `miles` 核心库、`rollout` 主链路、CI 无影响。对团队的实质影响是消除配额浪费隐患，以及需要与 #1913 协调合并顺序；对使用者而言，模块不再提供可执行的快照预热入口，但该入口原本就无法被任何 `create` 消费，属于清理而非功能回退。
- 风险标记：__main__入口删除，与 #1913 合并冲突，测试未纳入 CI

关联脉络

- PR #1790 `openenv/tbench2`: score the shared-server leg natively; retire the adapter compensation: 同一 `openenv/tbench2` 示例链路；该 PR 退役了共享服务器腿的适配补偿，本 PR 继续收敛遗留的 `daytona` 快照 `bake` CLI，同属“旧机制切换为声明式 / 原生路径”的演进方向。