

PR #24375 完整报告

sgl-project/sglang

[SMG] Expand K8s integration tests: cross-namespace, lifecycle, multi-model

合并时间: 2026-05-14 11:08

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/24375>

执行摘要

- 一句话: 跨命名空间、生命周期、多模型 K8s 集成测试
- 推荐动作: 值得精读, 尤其是测试设计中的以下决策:
 - 使用独占标签 (如 `cross-ns-test=true`、`lifecycle=restart`) 隔离不同测试的 worker 池, 避免跨文件干扰。
 - 在 IP 变化测试中优雅处理 CNI IP 重用场景, 通过 `skip` 而非 `false-pass`。
 - 优雅排空测试验证 `elapsed < grace_secs`, 确保 `deregistration` 在 `deletionTimestamp` 触发而非等到 Pod 完全终止。这些模式可以复用到底层 `sglang` 或其他微服务的集成测试中。

功能与动机

PR Body 指出: 构建在 K8s 集成测试脚手架 (#24278) 之上, 新增 5 个测试覆盖原始套件未触及的服务发现路径, 确保网关在各种场景下的健壮性。

实现拆解

1. 添加 YAML 清单文件: 在 `manifests/` 目录下新增 4 个 YAML 文件 (`gateway-restart.yaml`、`gateway-cluster-scoped.yaml`、`rbac-cluster-scoped.yaml`、`gateway-multimodel.yaml`), 定义不同配置的网关 Deployment 和 Service, 用于测试重启、集群范围发现和多模型隔离。
2. 编写测试辅助函数: 每个测试文件包含私有的辅助函数, 如 `_deploy_worker_pod`、`_safe_force_delete`、`_wait_for_pod_gone`、`_ensure_namespace` 等, 用于部署 / 清理 worker Pod 并获取 Pod 信息。这些函数通过 `subprocess` 调用 `kubectl`, 并注重错误信息的清晰呈现。
3. 实现测试类与 fixture: 使用 `pytest fixture` 管理网关部署 (模块级), 如 `restart_gateway`、`cluster_scoped_gateway`、`multimodel_gateways`, 确保网关就绪后再启动 `port-forward`。测试类 (如 `TestGatewayRestart`、`TestClusterWideDiscovery`、`TestMultiModelSelectorIsolation`) 包含具体的测试方法。
4. 执行断言与轮询: 测试通过网关的 HTTP API 查询注册的 worker, 使用 `_poll_until` 轮询直到条件满足 (如 worker 数量匹配、IP 变更生效), 并通过 URL membership 断言而非单纯计数, 避免跨测试干扰。

关键文件:

- `sgl-model-gateway/e2e_test/k8s_integration/test_lifecycle.py` (模块 测试文件; 类别 `test`; 类型 `test-coverage`; 符号 `_deploy_worker_pod`, `_safe_force_delete`, `_wait_for_pod_gone`, `_gone`) : 核心测试文件, 包含 3 个生命周期场景: 网关重启、Pod IP 变化、优雅排空, 共 554 行。
- `sgl-model-gateway/e2e_test/k8s_integration/test_cross_namespace.py` (模块 测试文件; 类别 `test`; 类型 `test-coverage`; 符号 `_deploy_worker_pod`, `_get_pod_ip`, `_safe_delete_pod`, `_ensure_namespace`) : 验证集群范围服务发现, 使用 `ClusterRole` 且不设 `--service-discovery-namespace`, 确保两个不同命名空间的 `worker` 均被注册。
- `sgl-model-gateway/e2e_test/k8s_integration/test_multi_model.py` (模块 测试文件; 类别 `test`; 类型 `test-coverage`; 符号 `_deploy_model_worker`, `_safe_force_delete`, `multimodel_gateways`, `TestMultiModelSelectorIsolation`) : 验证多模型选择器隔离, 两个网关分别匹配 `model=llama` 和 `model=qwen`, 确保 `worker` 只被对应网关注册。
- `sgl-model-gateway/e2e_test/k8s_integration/manifests/gateway-multimodel.yaml` (模块 清单文件; 类别 `test`; 类型 `test-coverage`) : 定义了两个网关 (`llama` 和 `qwen`) 的 `Deployment` 和 `Service`, 使用不同的 `--selector` 值。
- `sgl-model-gateway/e2e_test/k8s_integration/manifests/gateway-restart.yaml` (模块 清单文件; 类别 `test`; 类型 `test-coverage`) : 定义重启测试专用的网关, 使用不同端口 (`30005`) 和标签选择器 (`lifecycle=restart`), 避免干扰其他测试。
- `sgl-model-gateway/e2e_test/k8s_integration/manifests/gateway-cluster-scoped.yaml` (模块 清单文件; 类别 `test`; 类型 `test-coverage`) : 定义集群范围发现网关, 不设 `--service-discovery-namespace`, 并使用 `ClusterRole`。
- `sgl-model-gateway/e2e_test/k8s_integration/manifests/rbac-cluster-scoped.yaml` (模块 清单文件; 类别 `test`; 类型 `test-coverage`) : 定义集群范围的 RBAC 规则, 供 `cluster-scoped gateway` 使用。

关键符号: `test_workers_re_discovered_without_duplicates_after_restart`, `test_pod_ip_change`, `test_graceful_drain`, `test_workers_in_two_namespaces_are_both_discovered`, `test_each_gateway_sees_only_its_model_pool`, `_wait_for_pod_gone`, `_gone`, `restart_gateway`, `cluster_scoped_gateway`, `multimodel_gateways`

关键源码片段

`sgl-model-gateway/e2e_test/k8s_integration/test_lifecycle.py`

核心测试文件, 包含 3 个生命周期场景: 网关重启、Pod IP 变化、优雅排空, 共 554 行。

```
"""Worker lifecycle integration tests.
```

```
Covers three scenarios that the existing reconciliation/PD tests don't:
```

1. Gateway pod restart with persistent workers — verifies the K8s watcher re-discovers existing pods after the gateway restarts, with no duplicate registrations.
2. Pod IP change (same pod name, new IP) — verifies the gateway's worker registry tracks the new IP after a pod is force-deleted and recreated, not the stale one.
3. Graceful drain — verifies the gateway deregisters a worker as soon as

K8s sets ``metadata.deletionTimestamp`` (handled by `handle_pod_deletion` in `sgl-model-gateway/src/service_discovery.rs:533`), instead of waiting for the pod to fully terminate. This is what keeps the registry fresh during long `terminationGracePeriodSeconds` windows / `preStop` hooks.

```
"""
```

```
# ... imports and constants ...
```

```
def _wait_for_pod_gone(name: str, timeout: int = 60):
```

```
    """Wait until a pod no longer exists.
```

```
    Uses `kubectl get -o name --ignore-not-found`: empty stdout means the pod
    is gone (no need to substring-match "NotFound" against stderr, which is
    locale- and version-fragile). Any non-zero rc is a real cluster error
    (apise...
```

```
"""
```

```
def check_gone():
```

```
    result = _kubectl(
        "get", "pod", name, "-n", NAMESPACE, "-o", "name",
        "--ignore-not-found", check=False,
    )
```

```
    if result.returncode != 0:
```

```
        raise RuntimeError(f"kubectl get failed: {result.stderr}")
```

```
    # stdout empty means pod is gone
```

```
    return not result.stdout.strip()
```

```
    # Use the existing _poll_until helper instead of a custom loop
```

```
    _poll_until(check_gone, timeout=timeout, pause=2)
```

上述代码片段展示了 `_wait_for_pod_gone` 函数，它利用 `--ignore-not-found` 标志和 `_poll_until` 轮询（在第二提交中根据 review 重构）。其他关键函数如 `_deploy_worker_pod`、`restart_gateway fixture` 和三个测试方法（`test_workers_re_discovered_without_duplicates`、`_after_restart`、`test_pod_ip_change`、`test_graceful_drain`）协同工作，覆盖声明中的场景。

评论区精华

Review 由 `gemini-code-assist[bot]` 提出三点改进建议，均已在第二提交中解决：

- 复用 `_poll_until` helper: `_wait_for_pod_gone` 自定义轮询逻辑重复了 `confstest.py` 中的 `_poll_until`，建议改为调用已有函数以提高维护性。
- 安全访问 `items[0]`: 直接访问 `res["items"][0]` 可能因空列表引发 `IndexError`，应在访问前用 `assert` 检查列表非空。
- 避免脆弱的字符串匹配: 在 `_gone` 函数中通过检查 `stderr` 是否包含 "NotFound" 来判定 Pod 是否存在，这受语言环境影响，建议改用 `ignore-not-found` 标志并检查输出是否为空。以上建议均被作者采纳并在第二提交中修复。
- 复用 `_poll_until` 替代自定义轮询 (design): 作者在第二提交中重构了 `_wait_for_pod_gone`，使用 `_poll_until` 进行轮询，并保留了自定义检查函数。

- 安全访问 items[0] 防空列表 (correctness): 作者在第二提交中添加了 `assert res.get("items")`, "No pods found for smg-gateway-restart" 保护, 使失败信息更可读。
- 避免脆弱的 `stderr` 字符串匹配 (testing): 作者在第二提交中修改了 `_gone` 函数, 使用 `--ignore-not-found` 标志并检查 `stdout` 是否为空, 消除了依赖 `stderr` 文本的脆弱性。

风险与影响

- 风险: 本次变更仅涉及测试文件和部署清单, 未修改任何运行时源码, 因此源代码回归风险非常低。主要风险来自测试本身对 K8s 集群环境的依赖: `kubectl` 命令可能因上下文配置错误而失败; Pod 状态轮询超时可能导致测试不稳定; IP 重用场景被跳过但未标记强 `skip`, 可能遗漏问题。但这些风险已被 PR 中的设计 (独占标签、模块级 fixture、轮询重试) 和 `review` 修复降低了。此外, 新增的 YAML 清单引入了额外的网关 Deployment, 需要确保 CI 集群资源充足。
- 影响: 影响范围: 仅限 `sgl-model-gateway` 的 K8s 集成测试套件。影响程度: 低。对用户无感知, 对系统无性能影响, 但显著提高了服务发现功能 (跨命名空间、生命周期、多模型) 的测试信心。测试套件从 8 个增加到 13 个 (加上原 PR 的 8 个), 覆盖了更多关键路径。团队可通过这些测试更早地发现回归。
- 风险标记: 测试依赖 K8s 集群环境, `kubectl` 配置依赖, IP 重用场景被跳过未全覆盖, 新增额外网关部署需资源

关联脉络

- PR #24278 K8s integration test scaffolding (original): 本 PR 构建在 #24278 的测试框架之上, 使用了其提供的 `conftest` 函数 (如 `_poll_until`、`_port_forward_start` 等) 和基础清单。