

PIR单测推全

目录

- 背景
- 分工介绍
- 复现方法
- 修复原则
- 历史典型问题介绍
 - 5.1 不能复现问题的单测
 - 5.2 PIR下废弃的单测
 - 5.3 with paddle.pir_utils.OldIrGuard()
 - 5.4 有些API没有支持PIR模式需要新加支持
 - 5.5 静态图逻辑的isinstance只有Variable没有paddle.pir.Value
 - 5.6 新老静态图的Dtype使用的是不用的类型
 - 5.7 将Value转成shape需要用特殊的方法
 - 5.8 PIR Fetch只能Fetch Value，不能是name
 - 5.9 _manual_program_seed相关：
 - 5.10 暴露的PIR一些机制的空缺：
 - 5.11 暴露的一些深层次机制bug：
- 参考资料
 - 6.1 PIR相关的设计文档
 - 6.2 旧版静态图相关的参考资料
 - 6.3 历史修复PR

1. 背景



PIR是Paddle的新版静态图IR表达，替代了旧版静态图的IR。PIR是Paddle的一次重大基础机制升级，目前进入推全、收尾阶段。当前PIR核心机制基本健全，但有些API或者一些特定功能点还没适配PIR机制，这是本次推全工作的一部分。此外，CI单测是保障Paddle质量的重要机制。有些单测基于静态图API开发，在PIR下暴露了PIR适配不健全的问题，我们需要补齐PIR逻辑。有些单测基于旧版静态图特定API开发，在PIR下明确无法支持，我们需要结合单测的测试“初衷”，或者改造单测也支持PIR测试，或者开发等同单测测试PIR，或者隔离单测仅负责旧版动态图的质量保障。

当前，我们正借助CI单测修复来推动以上工作。我们的最终目标是CI所有流水线（PR-CI-Coverage,PR-CE-Framework,PR-CI-Api-Benchmark,PR-CI-Auto-Parallel,PR-CI-CINN,PR-CI-CINN-GPU,PR-CI-CINN-GPU-CUDNN-OFF,PR-CI-CINN-X86,PR-CI-Distribute-stable,PR-CI-Hygon-DCU,PR-CI-Inference,PR-CI-Kunlun-R200,PR-CI-Kunlun-bxcheck,PR-CI-Mac-Python3,PR-CI-Model-benchmark,PR-

CI-NPU-910B-Paddle,PR-CI-OP-benchmark,PR-CI-Py3,PR-CI-Py3-PIR,PR-CI-SOT,PR-CI-Windows,PR-CI-Windows-Inference,PR-CI-Windows-OPENBLAS) 全部支持PIR，之后Paddle将默认工作于PIR模式下。

这些流水线中最基础的流水线是Py3、Coverage、Mac、Windows、Windows-OPENBLAS、PR-CI-NPU-910B-Paddle，这六条流水线也是Paddle三个大方向（框架、分布式、推理）共建的流水线。我们从这五条流水线开始推动修复。其中Py3已修复大半，其余暂未启动。初步、不完全统计五条流水线合并仍有838个以上单测要修复。

2. 分工介绍

- 我们对五条流水线的问题单测做了初步整理合并，标记了归属团队。再将每个团队的单测平均分配给该团队参与的同学。目前的分工如下： [PIR单测统计-7月1日](#)
 - （分布式的同学暂时先写到这里： [分布式单测处理](#)，后边伟宝会处理。）

需要说明的是：

- 由于CI日志复杂，当前的统计可能不全，后期还会增加单测任务给相应同学
- 建议分布式、推理、框架三个方向的Leader结合每位同学负责的领域，调整单测分配，让单测任务与同学负责领域更加匹配

3. 复现方法

1. 编译Paddle

Py3流水线的编译命令为：

</>Shell

```
1 cmake .. -DCMAKE_BUILD_TYPE=Release -DWITH_GPU=OFF -DWITH_SHARED_PHI=ON -
  DWITH_TENSORRT=OFF -DWITH_ROCM=OFF -DWITH_CINN=OFF -DWITH_DISTRIBUTE=ON -
  DWITH_MKL=OFF -DWITH_AVX=OFF -DCUDA_ARCH_NAME=All -DNEW_RELEASE_PYPI=OFF -
  DNEW_RELEASE_ALL=OFF -DNEW_RELEASE_JIT=OFF -DWITH_PYTHON=ON -DWITH_TESTING=ON -
  DWITH_COVERAGE=OFF -DWITH_INCREMENTAL_COVERAGE=OFF -
  DCMAKE_EXPORT_COMPILE_COMMANDS=ON -DWITH_INFERENCE_API_TEST=ON -DPY_VERSION=3.9 -
  DWITH_PSCORE=ON -DWITH_PSLIB=OFF -DWITH_GLOO=ON -DWITH_XPU=OFF -DWITH_IPU=OFF -
  DXPU_SDK_ROOT= -DWITH_XPU_BKCL=OFF -DWITH_XPU_XRE5=OFF -DWITH_ARM=OFF -
  DWITH_STRIP=ON -DON_INFER=OFF -DWITH_HETERPS=OFF -DWITH_GPU_GRAPH=OFF -
  DCUDA_ARCH_BIN= -DWITH_RECORD_BUILDTIME=OFF -DWITH_UNITY_BUILD=ON -
  DWITH_ONNXRUNTIME=OFF -DWITH_CUDNN_FRONTEND=OFF -DWITH_CPP_TEST=ON
```

Coverage流水线的编译命令为：

</>Shell

```
1 cmake .. -DCMAKE_BUILD_TYPE=Release -DWITH_GPU=ON -DWITH_SHARED_PHI=ON -
  DWITH_TENSORRT=ON -DWITH_ROCM=OFF -DWITH_CINN=OFF -DWITH_DISTRIBUTE=ON -
  DWITH_MKL=ON -DWITH_AVX=ON -DCUDA_ARCH_NAME=Volta -DNEW_RELEASE_PYPI=OFF -
```

```
DNEW_RELEASE_ALL=OFF -DNEW_RELEASE_JIT=OFF -DWITH_PYTHON=ON -DWITH_TESTING=ON -  
DCMAKE_EXPORT_COMPILE_COMMANDS=ON -DWITH_INFERENCE_API_TEST=ON -DPY_VERSION=3.9 -  
DWITH_PSCORE=ON -DWITH_PSLIB=OFF -DWITH_GLOO=ON -DWITH_XPU=OFF -DWITH_IPU=OFF -  
DXPU_SDK_ROOT= -DWITH_XPU_BKCL=OFF -DWITH_XPU_XRE5=OFF -DWITH_ARM=OFF -  
DWITH_STRIP=ON -DON_INFER=ON -DWITH_HETERPS=OFF -DCUDA_ARCH_BIN= -  
DWITH_RECORD_BUILDTIME=OFF -DWITH_UNITY_BUILD=ON -DWITH_ONNXRUNTIME=OFF -  
DWITH_CUDNN_FRONTEND=OFF -DWITH_CPP_TEST=ON
```

Mac流水线的编译命令为：

</>

Shell

```
1 cmake .. -DCMAKE_BUILD_TYPE=Release -DWITH_GPU=OFF -DWITH_TENSORRT=OFF -  
  DWITH_ROCM=OFF -DWITH_CINN=OFF -DWITH_DISTRIBUTE=OFF -DWITH_MKL=ON -DWITH_AVX=ON -  
  DCUDA_ARCH_NAME=All -DNEW_RELEASE_PYPI=OFF -DNEW_RELEASE_ALL=OFF -  
  DNEW_RELEASE_JIT=OFF -DWITH_PYTHON=ON -DWITH_TESTING=ON -DWITH_COVERAGE=OFF -  
  DWITH_INCREMENTAL_COVERAGE=OFF -DCMAKE_EXPORT_COMPILE_COMMANDS=ON -  
  DWITH_INFERENCE_API_TEST=OFF -DPY_VERSION=3.9 -DWITH_PSCORE=OFF -DWITH_PSLIB=OFF -  
  DWITH_GLOO=OFF -DWITH_XPU=OFF -DWITH_IPU=OFF -DXPU_SDK_ROOT= -DWITH_XPU_BKCL=OFF -  
  DWITH_ARM=OFF -DWITH_STRIP=ON -DON_INFER=OFF -DWITH_HETERPS=OFF -DCUDA_ARCH_BIN= -  
  DWITH_RECORD_BUILDTIME=OFF -DWITH_UNITY_BUILD=OFF -DWITH_ONNXRUNTIME=OFF -  
  DWITH_CUDNN_FRONTEND=OFF -DWITH_SHARED_PHI=ON
```

2. 执行单测

</>

Shell

```
1 # 先安装whl包，进入build目录  
2 ctest -R 单测名称 -V      # 原版测试  
3 FLAGS_enable_pir_api=1 ctest -R 单测名称 -V      # PIR模式测试
```

Windows流水线复现参考：[📖 windows复现定位手册（1.0版）](#)

NPU流水线复现参考：[📖 PR-CI-NPU 调试开发手册](#)，NPU调试环境可以联系 田戈骁。

更详细的Paddle编译、开发、git提交流程等可参考：[📖 新人入职培训手册](#) [📖 飞桨新人培训课件](#)

4. 修复原则

1. Py3流水线：优先修复Py3问题单测，因为Py3是最容易复现、调试友好的流水线。

- Py3流水线的问题单测均在test/deprecated/xxx/目录下，修复后将单测迁移到test/xxx/目录下
- Py3流水线有一天附属流水线叫Py3-PIR。Py3流水线用于测试FLAGS_enable_pir_api=False的场景，Py3-PIR用于测试FLAGS_enable_pir_api=True的场景。
- Py3-PIR流水线不测试deprecated下的单测，因此，只要迁移到test/xxx/后，Py3-PIR会自动测试FLAGS_enable_pir_api=True的场景

2. 如果不是Windows或者Mac独有的错误单测，只需要在Py3和Coverage下调试修复就可以提交PR了。因为大概率Py3和Coverage修复后，其它3条流水线也就恢复正常了。单测修复后，除Py3流水线外，需要

新加一个单测配置，意思是流水线要专门测试一下FLAGS_enable_pir_api的情况：

</>

Shell

```
1 if(WITH_COVERAGE OR APPLE OR WIN32 OR WITH_ASCEND_CL)
2   py_test_modules(test_xxx MODULES test_xxx ENVS FLAGS_enable_pir_api=1)
```

3. 如果单测在本地无法复现，可以直接添加`ENVS FLAGS_enable_pir_api=1`的单测配置，如果CI能够通过，说明这个单测问题已被别人修复，直接标记单测状态为“已修复”即可。
4. 有些单测，专门用于老静态图的测试，在与Mentor确认后，可以不做PIR下的测试。直接将单测迁移到test/deprecated/xxx/目录下，并将单测文件名修改为test_xxx_deprecated.py。
 - a. 注意，无论是向test/deprecated/xxx/目录移入单测，还是移出单测。要确认一下，其目录的CMakeList.txt中是否有关于该单测的特殊配置，配置也需同步迁移。比如该单测在CMakeList.txt的某个List列表中，或者为该单测设置了特定的超时时间。

5. 历史典型问题介绍

5.1 不能复现问题的单测

Py3下有些单测不能复现问题，需要从deprecated目录移出，示例

PR: <https://github.com/PaddlePaddle/Paddle/pull/64914>

其它流水线如果不能复现，需要使用py_test_modules(test_xxx MODULES test_xxx ENVS FLAGS_enable_pir_api=1)做CI验证，并将py_test_modules这个逻辑合入develop。

5.2 PIR下废弃的单测

明确不需要支持PIR的可以直接放到deprecated目录相应的子文件夹下，并以_deprecated后缀命名，示例

PR: <https://github.com/PaddlePaddle/Paddle/pull/64078/files>

5.3 with paddle.pir_utils.OldIrGuard()

有些逻辑，必须运行在旧静态图模型下，需要使用with paddle.pir_utils.OldIrGuard()标记，示例

PR: [https://github.com/PaddlePaddle/Paddle/pull/64055/files#diff-](https://github.com/PaddlePaddle/Paddle/pull/64055/files#diff-0b7387eda330700c5a5691f3215c085b3f56547a3549d3a3657c7c221571bac7)

[0b7387eda330700c5a5691f3215c085b3f56547a3549d3a3657c7c221571bac7](https://github.com/PaddlePaddle/Paddle/pull/64055/files#diff-0b7387eda330700c5a5691f3215c085b3f56547a3549d3a3657c7c221571bac7), [https://github.com/PaddlePaddle/Paddle/pull/65440/files#diff-](https://github.com/PaddlePaddle/Paddle/pull/65440/files#diff-128600c228e85fea0d4c06ef543697590fde5a8b3b855b2324048e9f384657c9)
[128600c228e85fea0d4c06ef543697590fde5a8b3b855b2324048e9f384657c9](https://github.com/PaddlePaddle/Paddle/pull/65440/files#diff-128600c228e85fea0d4c06ef543697590fde5a8b3b855b2324048e9f384657c9)

5.4 有些API没有支持PIR模式需要新加支持

旧静态图使用append_op支持，新静态图使用in_pir_mode: _C_ops.xxx来支持。有些API没有写pir mode的分支。示例PR: <https://github.com/PaddlePaddle/Paddle/pull/64645/files#diff-92bb1c906155a8410263bbff53f613df4285352f2c746b78afdbfb44e96ef3ec>

5.5 静态图逻辑的isinstance只有Variable没有paddle.pir.Value

示例PR: <https://github.com/PaddlePaddle/Paddle/pull/64908/files>

5.6 新老静态图的Dtype使用的是不用的类型

新静态图使用的是paddle.base.core.DataType，老静态图用的是VarType，因此在某些逻辑上，容易出现交叉Dtype的使用导致报错，示例PR:

<https://github.com/PaddlePaddle/Paddle/pull/64463/files#diff-0b7387eda330700c5a5691f3215c085b3f56547a3549d3a3657c7c221571bac7>

5.7 将Value转成shape需要用特殊的方法

示例PR: <https://github.com/PaddlePaddle/Paddle/pull/64307/files#diff-92bb1c906155a8410263bbff53f613df4285352f2c746b78afdbfb44e96ef3ec>

5.8 PIR Fetch只能Fetch Value，不能是name

示例PR: <https://github.com/PaddlePaddle/Paddle/pull/64088/files#diff-1a32d9fa6eb612eb36b89465fd002b4bda45e74d51f7bfb93d5ae3eda1d4887e>

5.9 _manual_program_seed相关:

_manual_program_seed用于配置随机种子，如果在PIR FLAGS默认打开的情况下，动态图需要用这个种子。需要加with paddle.pir_utils.OldIrGuard()。示例

PR: <https://github.com/PaddlePaddle/Paddle/pull/65473/files>

5.10 暴露的PIR一些机制的空缺:

示例

PR: <https://github.com/PaddlePaddle/Paddle/pull/64442>, <https://github.com/PaddlePaddle/Paddle/pull/65239>

5.11 暴露的一些深层次机制bug:

示例

PR: <https://github.com/PaddlePaddle/Paddle/pull/64312>, <https://github.com/PaddlePaddle/Paddle/pull/65307>

此外，还有很多不典型问题，只能靠大家自己慢慢调试分析。

6. 参考资料

6.1 PIR相关的设计文档

《IR Dialect源码学习》

- IR升级整体规划以及第一阶段(类型系统)设计文档
- IR升级第二阶段(模型结构)设计评审
- 新IR下自动微分模块设计
- IR控制流设计评审方案
- PIR适配OneDNN的方案
- PIR 适配 AMP 的方案
- PIR下 Save/load 体系设计
- 新 IR 下基于 DRR 的 Pass 简化技术方案
- 静态图半自动并行架构基于 PIR 重构升级

6.2 旧版静态图相关的参考资料

《静态图执行过程》



Paddle Fluid框架执行逻辑 v1.2 - 陈威行.pptx
2.2MB



PaddlePaddle学习笔记——王欢.docx
2.1MB



Paddle学习笔记——王欢.pptx

5.5MB

6.3 历史修复PR

<https://github.com/PaddlePaddle/Paddle/pull/64055>

<https://github.com/PaddlePaddle/Paddle/pull/64064>

<https://github.com/PaddlePaddle/Paddle/pull/64078>

<https://github.com/PaddlePaddle/Paddle/pull/64088>

<https://github.com/PaddlePaddle/Paddle/pull/64096>

<https://github.com/PaddlePaddle/Paddle/pull/64124>

<https://github.com/PaddlePaddle/Paddle/pull/64166>

<https://github.com/PaddlePaddle/Paddle/pull/64276>

<https://github.com/PaddlePaddle/Paddle/pull/64277>

<https://github.com/PaddlePaddle/Paddle/pull/64307>

<https://github.com/PaddlePaddle/Paddle/pull/64308>

<https://github.com/PaddlePaddle/Paddle/pull/64312>

<https://github.com/PaddlePaddle/Paddle/pull/64314>

<https://github.com/PaddlePaddle/Paddle/pull/64319>

<https://github.com/PaddlePaddle/Paddle/pull/64350>

<https://github.com/PaddlePaddle/Paddle/pull/64442>

<https://github.com/PaddlePaddle/Paddle/pull/64454>

<https://github.com/PaddlePaddle/Paddle/pull/64455>

<https://github.com/PaddlePaddle/Paddle/pull/64457>

<https://github.com/PaddlePaddle/Paddle/pull/64463>

<https://github.com/PaddlePaddle/Paddle/pull/64486>

<https://github.com/PaddlePaddle/Paddle/pull/64487>

<https://github.com/PaddlePaddle/Paddle/pull/64645>

<https://github.com/PaddlePaddle/Paddle/pull/64754>

<https://github.com/PaddlePaddle/Paddle/pull/64845>

<https://github.com/PaddlePaddle/Paddle/pull/64891>

<https://github.com/PaddlePaddle/Paddle/pull/64904>

<https://github.com/PaddlePaddle/Paddle/pull/64908>

<https://github.com/PaddlePaddle/Paddle/pull/64914>
<https://github.com/PaddlePaddle/Paddle/pull/64917>
<https://github.com/PaddlePaddle/Paddle/pull/64923>
<https://github.com/PaddlePaddle/Paddle/pull/64960>
<https://github.com/PaddlePaddle/Paddle/pull/64966>
<https://github.com/PaddlePaddle/Paddle/pull/65034>
<https://github.com/PaddlePaddle/Paddle/pull/65038>
<https://github.com/PaddlePaddle/Paddle/pull/65239>
<https://github.com/PaddlePaddle/Paddle/pull/65307>
<https://github.com/PaddlePaddle/Paddle/pull/65346>
<https://github.com/PaddlePaddle/Paddle/pull/65352>
<https://github.com/PaddlePaddle/Paddle/pull/65440>
<https://github.com/PaddlePaddle/Paddle/pull/65473>