

4月22日,FATE开源社区第九期圆桌会圆满落幕。本次圆桌会,三位业内专家分别为我们讲述了,FATE on Spark 传输服务的设计与实现,FATE 基础引擎框架设计与实现,FATE v1.7 版本展望和需求征集。

接下来带大家回顾一下经典**问答环节**,为新老朋友答疑解惑。

## **FATE on Spark 传输服务的设计与实现**

**pulsar 的相关功能之后也会在 kubefate 中安装是吗?**

是的,目前 pulsar 相关功能已经在 kubefate 中有了,但是就是比较简单的 pulsar 的 standalone 部署,后面我们会继续 pulsar 的集群部署。

**刚刚谈到 FATE on eggroll 比 FATE on Spark 就是传输的性能差别,能够展开说说,大概有多大差别?**

其实也不是传输性能差别,而是有些资源利用的不够好。目前在 pulsar 的实现中,数据量太大的时候,中间会有一些卡顿,甚至整个训练传输停滞。通过 debug,数据量大的时候,开的 topic 很多,这时如果没有及时回收,就会导致训练停滞,后面我们会跟 pulsar 一起合作去 fix。其实,也不是传输速率上的差别,而是资源没有及时回收,导致有停滞的情况出现。

**资源没有回收,具体是什么资源?**

具体资源就是 topic,用完没有及时回收。

**pulsar 对应于 FATE 1.6.0 版本中吗?**

是的。

**为什么考虑要开发三种不同的 federation 服务呢?**

eggroll 的传输服务是 rollsite,它没办法单独使用,必须配合 eggroll 里面的 nodemanager 和 clustermanager 服务。所以,在 FATE on Spark 的第一个实现中是以 rabbit MQ 作为传输服务,因为它很简单,实现起来很直观,后面考虑到需要用星型组网和更高吞吐率的需求,所以,我们就引入了 pulsar。

**请问下有测试过 spark executor 比较多时, pulsar 的性能怎么样吗?**

目前还在测试,后续会把测试报告分享给大家。

**pulsar 可以支持 Fate-serving 星型组网吗?**

理论上可以支持,如果是 Fate-serving,可能需要适配,如果有具体的业务需求,可以跟我们提 issue。

## **FATE 基础引擎框架设计与实现**

**为什么在 python 中使用 os.kill 来清理进程,会生成僵尸进程,最终导致 JOB 无法自动标记为 failed?**

在 1.5.1 版本修复了,这个的原因是子进程退出的信号没有捕捉,跟 os.kill 使用没有关系。至于为什么要用 os.kill,它是操作系统提供的基础 API,根据不同的场景指定发送不同的信号,同时 FATE 也有作业异常检测定期回收机制。

**在 docker 容器中的集群部署,FATE 是如何启动 spark 做执行命令的,另外,一般我们的算法是用 MLlib 实现还是主要 RDD 编程实现?**

FATE 目前使用与 fateflow 部署在一起的 spark client 提交任务。

**如果不断将不同数据 upload 到执行态存储，有时会产生一些废弃的数据表，那有没有什么批量删除 upload 过的数据表的办法呀？**

这里有两件事情，第一件事情是 upload 数据表，用户可以自己在 namespace 下制定一些规则，第二件事情就是 upload history，用户可以通过这个接口查到 upload 数据表。但是我们不会直接提供批量删除的接口，因为这样非常危险。所以，我们提供两个接口，一个是 upload history，查出对应的数据表，用户可以通过这个 list，进行自己一些规则过滤，再调用我们 table delete 接口进行删除。

## **FATE-1.7 版本展望与需求征集**

**FATE 支持模型保存的功能吗，类似于 torch.save() 的功能？这样我就可以把模型调出到本地，进行本地测试，比较灵活。**

有保存，保存成 FATE 目录下的名为 model\_local\_cache 的本地文件夹。这里面有每个任务各组件的模型输出 proto buffer。Flow 是调用算法的 save\_model 接口去导出的，可以 grep -r save\_model python/federatedml 去看下具体实现逻辑进行适配。

**横向 serving 支持没太懂，可以展开讲下么？比如 TF 是可以从 Fate-serving 迁移到 TF-serving 么？**

现在横向模型里面，训练最终结果是单边拥有完整模型，跟纵向不一样，纵向是依赖于多方的，至少两方。一个典型例子是将横向逻辑回归模型转成 sklearn 的模型，复用 sklearn 的 serving 能力，不能简单理解为从 Fate-serving 迁移到 KF-serving，Fate-serving 是 FATE 提供的纵向联邦多方在线推理能力。横向模型可以直接适配业界 Serving 框架。

**Fate flow 现在的 pip 包和这里的 Fate flow 能区别开吗？**

Fate flow 现在还没有 pip 包，大家提到的 Fate flow 不是 FATE 官方的包，现在 FATE 官方的包只有 Fate-client 和 Fate-test。

**FATE 的模型支持用 onnx 转吗，比如说 tf 训练完转 FATE 上。**

不支持 onnx 转，下个版本支持的是 FATE 训练完之后转到其他框架，暂时不支持其它平台的异构模型转成 FATE 去使用。

**FATE 有机会支持 OT 吗？**

FATE 是支持 OT 协议的。secure\_information\_retrieval 组件使用了 OT 协议，当前 PSI 目前暂时没有用到 OT 的方案。

**1.7 版本可以有加载预训练模型的功能吗？**

1.7 预计会支持 check point 的功能，支持用户每迭代若干轮保存一次模型。warm start 的话需要结合用法来考虑，要看排期。

**checkpoint 的话，安全该怎么定，也许还是用同样的数据，但是标签方变了。**

checkpoint 模型是保存在本方的平台的，使用同样的数据但标签方变了说明模型是标签方给了其他方，做了模型迁移，这是需要协商决定的，非系统层面问题。

**如果从其他平台把这个预训练权重下载下来，放到服务方上，FATE 只需要 load 这个预训练模型就可以，这样可以吗？**

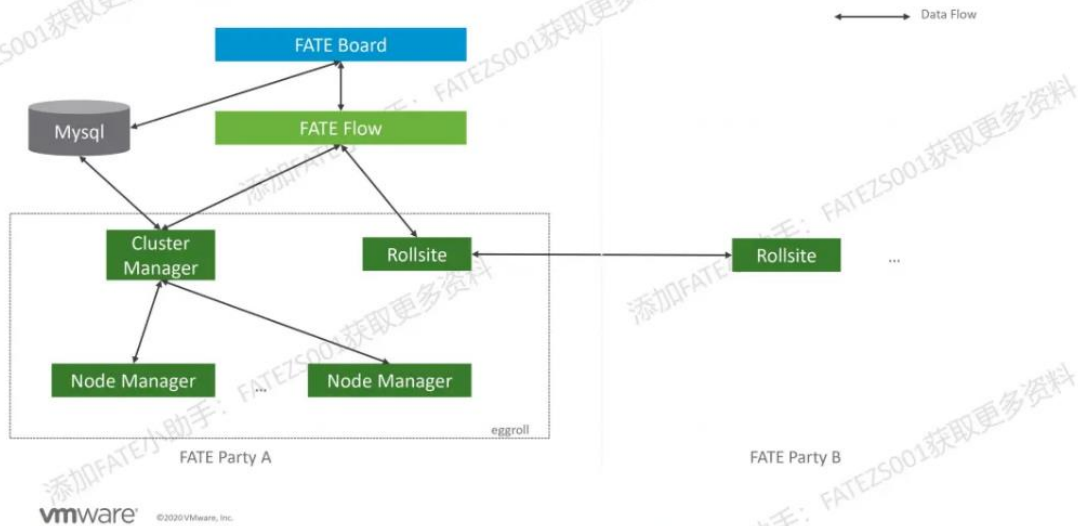
1.7 的规划中暂时不支持其它平台的模型转到 FATE 去热启，只支持 FATE 转到其他平台支持的模型去复用平台 Serving 能力。

**1.7 版本有没有考虑加入一些激励机制的方案？**

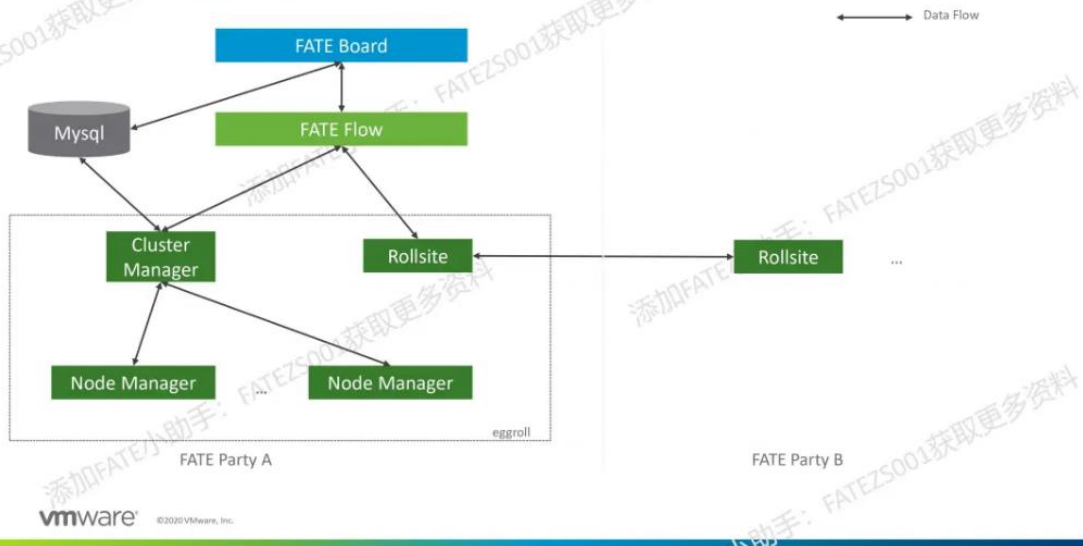
目前还没有考虑，未来可能会有。

以下为本次圆桌会的部分内容介绍，添加小助手（FATEZS001）可获取详细资料：

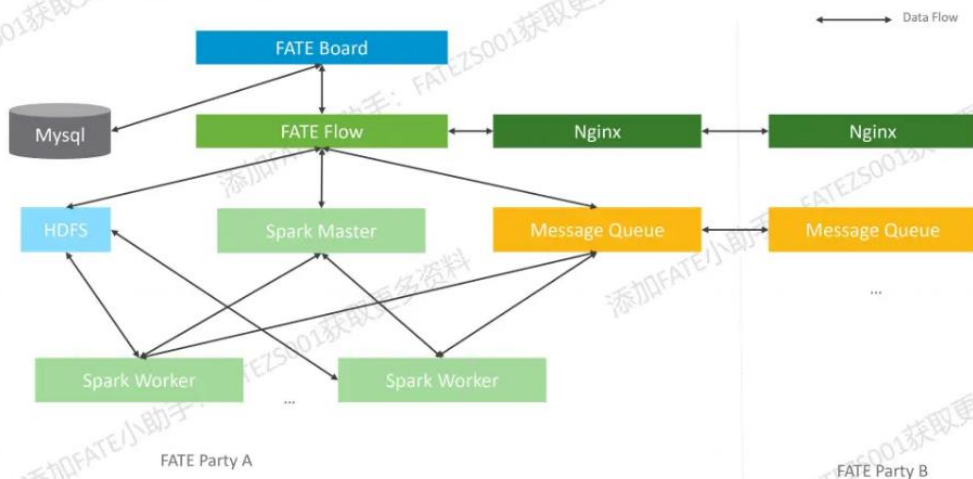
## FATE on Eggroll架构



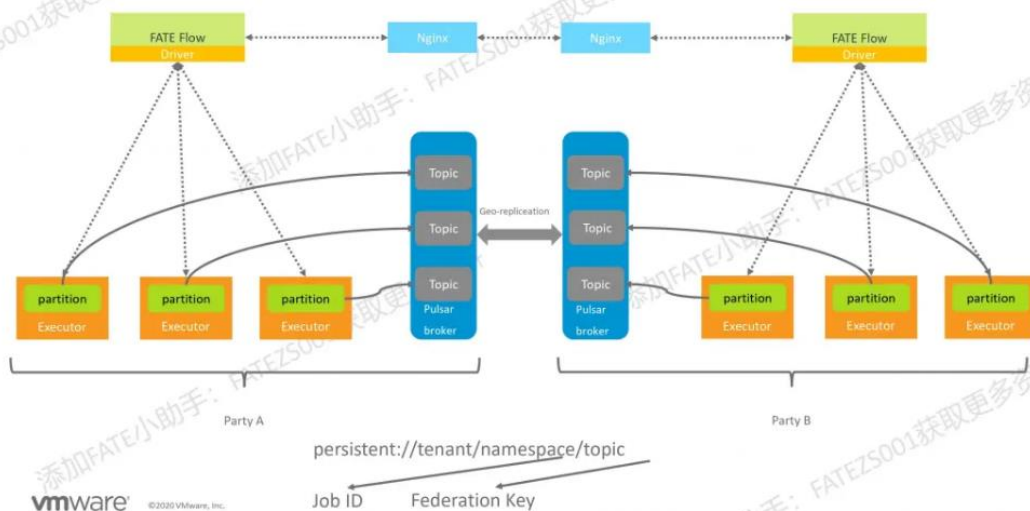
## FATE on Eggroll架构



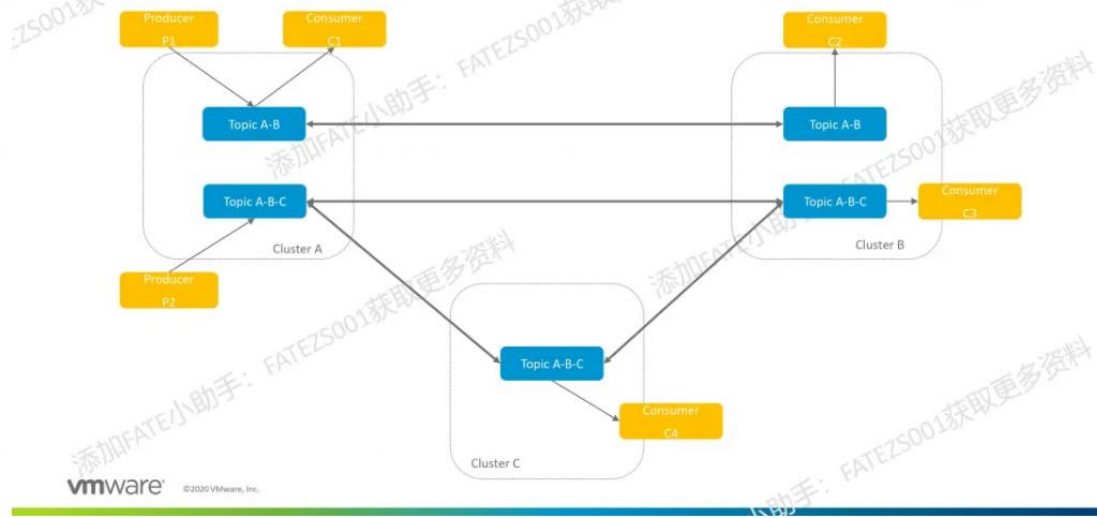
## FATE on Spark架构



## FATE on Spark with Pulsar实现细节



## Pulsar的跨地域复制(Geo-replication)



获取会议 PPT，或对圆桌会还有别的疑问？欢迎联系 FATE 开源社区助手获得帮助。

原文链接：<https://mp.weixin.qq.com/s/41TVzaG6oBLhNYbGy6qztzw>