

从 Kafka 迁移到 Raft

注意：这篇文章面向的是已熟练掌握通道配置更新交易的读者。由于迁移会涉及到几次通道配置升级交易，因此我们建议您先熟悉 [向通道添加组织](#) 教程，该教程详细讲述了通道更新的流程，熟练掌握后再尝试进行迁移。

对于想把通道从使用基于Kafka的排序服务转换成使用[基于Raft](#)的排序服务的用户，v1.4.2通过在网络各通道上进行一系列配置更新交易，使得这一点得到实现。

该教程将从宏观层面来讲述迁移流程，指出一些必要的细节，不会详细讲述每个命令。

假设与思考

尝试迁移之前，要考虑以下几点：

1. 该流程只针对从Kafka迁移到Raft，暂不支持其他类型orderer共识之间的迁移。
2. 迁移是单方向的。一旦排序服务被迁移成Raft并开始提交交易时，就没有办法再恢复成Kafka。
3. 因为排序节点必须关闭然后重新启动，所以在迁移期间必须允许停机。
4. 若迁移失败，只有当在本文后面规定的迁移点进行了备份，才可能对其恢复，如果未进行备份，则无法恢复至初始状态。
5. 必须在同一维护窗口对所有通道完成迁移。若只迁移部分通道则不可能恢复正常操作。
6. 在迁移过程结束时，每个通道都将拥有相同的Raft节点共识者集，与将出现在排序系统通道上的共识者集一样。这样就可以判断出成功的迁移。
7. 利用已部署的排序节点的现有账本就能完成迁移，orderer的增加或移除应在迁移完成后进行。

整体迁移流程

迁移共分为五个步骤。

1. 将系统置于维护模式，该模式拒绝应用程序交易，只有排序服务管理员能改动通道配置。
2. 关闭系统，当迁移过程中发生错误，进行备份。
3. 启动系统，每个通道都有自己的共识类型，各自的元数据都已修改。
4. 重启系统后系统开始用Raft共识运行；检查各通道以确保达到规定人数。
5. 将系统移出维护模式，重启正常功能。

准备迁移

准备迁移前还需进行以下步骤。

- 设计Raft部署，决定在Raft共识者者保留哪些排序服务节点。至少需要部署三个排序节点，但要注意的是，部署包含五个以上排序节点的共识者集能保证即使其中一个节点关闭，依然能维持高可用度，而对于只包含三个排序节点的共识者集，一旦由于某些原因（例如，正处于维护期间）导致其中一节点关闭，则极大降低了节点的可用度。

- 准备搭建Raft `Metadata` 配置的材料。注意：所有通道都需收到相同的Raft `Metadata` 配置。访问 [Raft配置指南](#) 获取更多相关信息。注意：你或许会发现最简单的方法就是用Raft共识协议来引导新的排序网络，然后从其配置中复制并修改共识元数据部分。无论使用哪种方法，都需要（对每个排序节点）：
 - `hostname`
 - `port`
 - `server certificate`
 - `client certificate`
- 编译系统中所有通道（系统和应用程序）的列表。确保你有对配置更新签名的正确证书。例如，相关的排序服务管理员身份。
- 确保所有的排序服务节点运行的Fabric版本相同，都是v1.4.2或该版之后的版本。
- 确保所有节点运行的Fabric版本不低于v1.4.2。确保所有通道都配置了支持迁移的通道能力。
 - orderer能力 `v1_4_2` （或之后的版本）
 - 通道能力 `v1_4_2` （或之后的版本）

维护模式的入口

建议在维护模式中设置排序服务之前关闭网络的节点和客户端。虽然让节点或客户端继续运行是安全的。但由于排序服务将拒绝所有请求，因此它们的日志将充满良性但会让人误解的故障。

参照 [向通道添加组织](#) 教程中的步骤来pull，翻译并检查每个通道的配置，从系统通道开始。该步中唯一需要你改动的地方是 `/Channel/Orderer/ConsensusType` 中的通道配置。在通道配置的一个JSON代表中需要改动的可能是 `.channel_group.groups.Orderer.values.ConsensusType`。

`ConsensusType` 由三个值代表：`Type`，`Metadata` 和 `State`，其中：

- `Type` 要么是 `kafka` 要么是 `etcdraft`（Raft）。该值只能在维护模式中才能修改。
- 如果 `Type` 是Kafka，那么 `Metadata` 就是空的，但是如果 `ConsensusType` 是 `etcdraft`，那就必须包含有效Raft元数据。下文中还会详细谈到。
- 当通道正在处理交易时 `State` 是 `STATE_NORMAL`，当通道正处于迁移过程中时则为 `STATE_MAINTENANCE`。

在通道配置更新的第一步，只用将 `State` 从 `Normal` 改成 `MAINTENANCE`。暂且先不要改 `Type` 或 `Metadata` 区域。要注意的是，当前 `Type` 应该是 `Kafka`。

当处于维护模式时，正常交易，与迁移无关的配置更新，peer用来索取新区块而发出的请求，这些都会被拒绝。这样做是为了避免在迁移过程中需要对节点进行备份和（如果需要的话）恢复，因为节点只有在成功完成迁移时才会收到更新。换句话说，我们想把排序服务备份点（下一步的内容）放置在节点的账本之前，以期能够在需要的时候撤销操作。不过，排序节点管理员能发出 `Deliver` 请求（管理员需要该功能以继续迁移流程）。

验证每个排序服务节点在各自通道是否已进入维护模式。要想完成验证，可通过获取最后一个配置区块，确保每个通道上的 `Type`，`Metadata`，`State` 分别是 `kafka`，空（上文中我们刚谈到Kafka没有元数据），`STATE_MAINTENANCE`。

当通道成功更新后，就可以备份排序服务了。

备份文件和关闭服务器

关闭所有的排序节点，Kafka服务器和Zookeeper服务器。**先关闭排序服务节点至关重要。**随后，在允许Kafka服务将其日志刷新到磁盘（一般耗时30秒，但可能会更久，具体时间取决于你的系统情况）后，应关闭Kafka服务器。在关闭orderer的同时关闭Kafka服务器会导致orderer的文件系统状态比Kafka服务器的新，这可能会阻止你的系统启动。

为这些服务器的文件系统创建一个备份。随后重启Kafka服务，紧接着再重启排序服务节点。

在维护模式中切换成Raft

迁移流程的下一步是为各通道进行通道配置更新。在配置更新中，将 `Type` 切换成 `etcdraft` (Raft) 的同时保持 `State` 为 `STATE_MAINTENANCE`，填写 `Metadata` 配置。我们强烈建议让所有通道上的 `Metadata` 配置保持一致。如果你想用不同的节点组建不同的共识者集，你就能在系统被重启为 `etcdraft` 模式后重新配置 `Metadata` 的配置。因此，提供一个完全相同的元数据对象并因此提供相同的共识者集就意味着，当重启节点时，如果系统通道组成仲裁并且能够退出维护模式时，那么其他通道也可能执行相同的操作。为各通道提供不同的共识者集会​​导致有一个通道成功形成集合而另一个通道失败了。

随后，验证每个排序服务是否都通过拉取和检测每个通道的配置来提交了 `ConsensusType` 改变配置更新。

注意：对于每一个通道，改变 `ConsensusType` 的交易必须是重启节点（下一步会谈到）之前的最后一次配置交易。如果这一步之后发生其他配置交易，那么节点最有可能在重启时崩溃，或者导致未定义的行为。

重启和验证主节点

注意：**重启后必须退出维护模式。**

在每个通道都完成 `ConsensusType` 更新后，关闭所有排序服务节点，关闭所有Kafka服务器和Zookeeper，然后仅重启排序服务节点。重启和Raft节点一样，在每个通道上组成一个集合，选出每个通道上的主节点。

注意:由于基于raft的排序服务使用客户端和服务端TLS证书 在排序节点之间进行身份验证，在您再次启动它们之前需要 **额外的配置**，请参阅[本地配置](#) 一节了解更多细节。

重启完成后，确保通过检测节点日志来**验证**各通道上都已选出各自的主节点（下文中指出了你需检测的内容）。这将证实迁移流程顺利完成。

当一个leader节点被选举出来，将显示对应各个通道的日志:

```
"Raft leader changed: 0 -> node-number channel=channel-name
node=node-number "
```

例如：

```
2019-05-26 10:07:44.075 UTC [orderer.consensus.etcdraft] serveRequest ->
INFO 047 Raft leader changed: 0 -> 1 channel=testchannel1 node=2
```

在这个例子中 `node 2` 指明了主节点由通道 `testchannel1` 的集合选举出来（主节点是 `node 1`）。

切出维护模式

在各通道上执行另一项通道配置更新（向截止目前你已发送过配置更新的排序节点发送配置更新），从而将 `State` 从 `STATE_MAINTENANCE` 切换成 `STATE_NORMAL`。如正常一样，从系统通道先开始。如果在排序系统通道上成功了，有可能所有通道上的迁移都成功了。要进行验证，请从排序节点中获取系统通道的最后一个配置区块，验证当前 `State` 为 `STATE_NORMAL`。为了完整性，请在各排序节点上进行验证。

当流程完成时，所有通道上的排序服务就能接受所有交易了。如果你像我们建议的那样关闭了节点和应用程序，现在你可以重启它们了。

中止和恢复

如果在迁移过程中还未退出维护模式时出现一个问题，只需执行以下恢复步骤：

1. 关闭排序节点和Kafka服务（服务器和Zookeeper的整体）。
2. 在改变 `ConsensusType` 之前将这些服务器的文件系统恢复成维护模式下进行的备份。
3. 重启这些服务器，排序节点将在维护模式下引导至Kafka。
4. 发送退出维护模式的配置更新，以继续使用Kafka来作为你的共识机制，或者在备份后恢复这些指令，修补阻碍形成Raft仲裁的错误，用修正过的Raft配置 `Metadata` 重新进行迁移。

若出现以下状态，则表明迁移可能未成功：

1. 某些节点崩溃或关闭。
2. 日志中没有关于各通道上成功选举出一个主节点的记录。
3. 尝试在系统通道上切换成 `STATE_NORMAL` 模式，但是失败了。