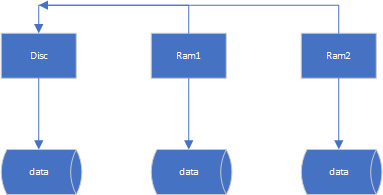
# Rabbitmq主备机房切割步骤

## 部署拓扑图

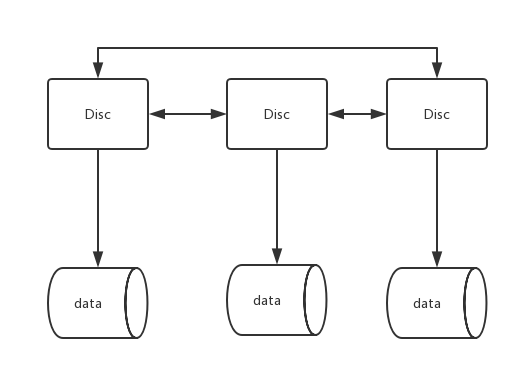
## rabbitmq部署拓扑图

现有stagging环境部署图如下：



一个磁盘节点，两个内存节点，内存节点的数据都镜像到磁盘节点，同时所有节点都做数据持久化

prod环境下，部署调整为：



所有节点都备份，全量镜像

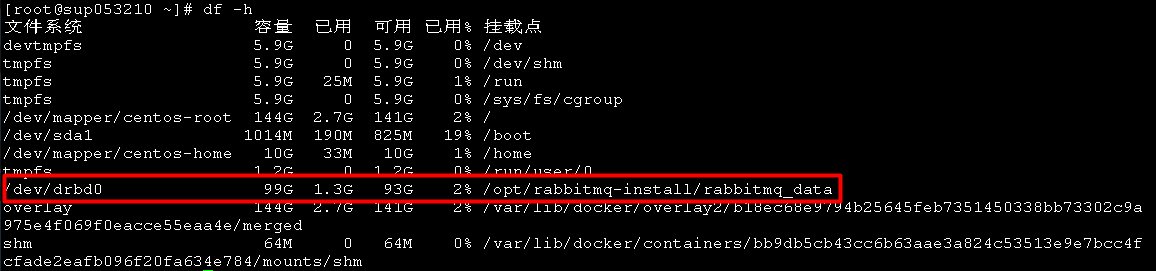
## 主备机房切割对应关系

|  |  |  |  |
| --- | --- | --- | --- |
| stagging | 节点 | 节点 | prod |
| 192.168.211.137 | disc | disc(主) | 192.168.210.53 |
| 192.168.211.171 | ram | disc | 192.168.210.61 |
| 192.168.211.188 | ram | disc | 192.168.210.147 |

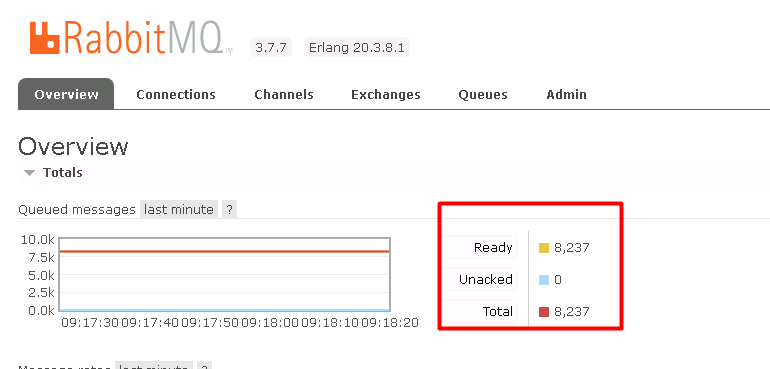
切割后，主机房rabbitmq主节点为：192.168.210.53

## 数据迁移方案

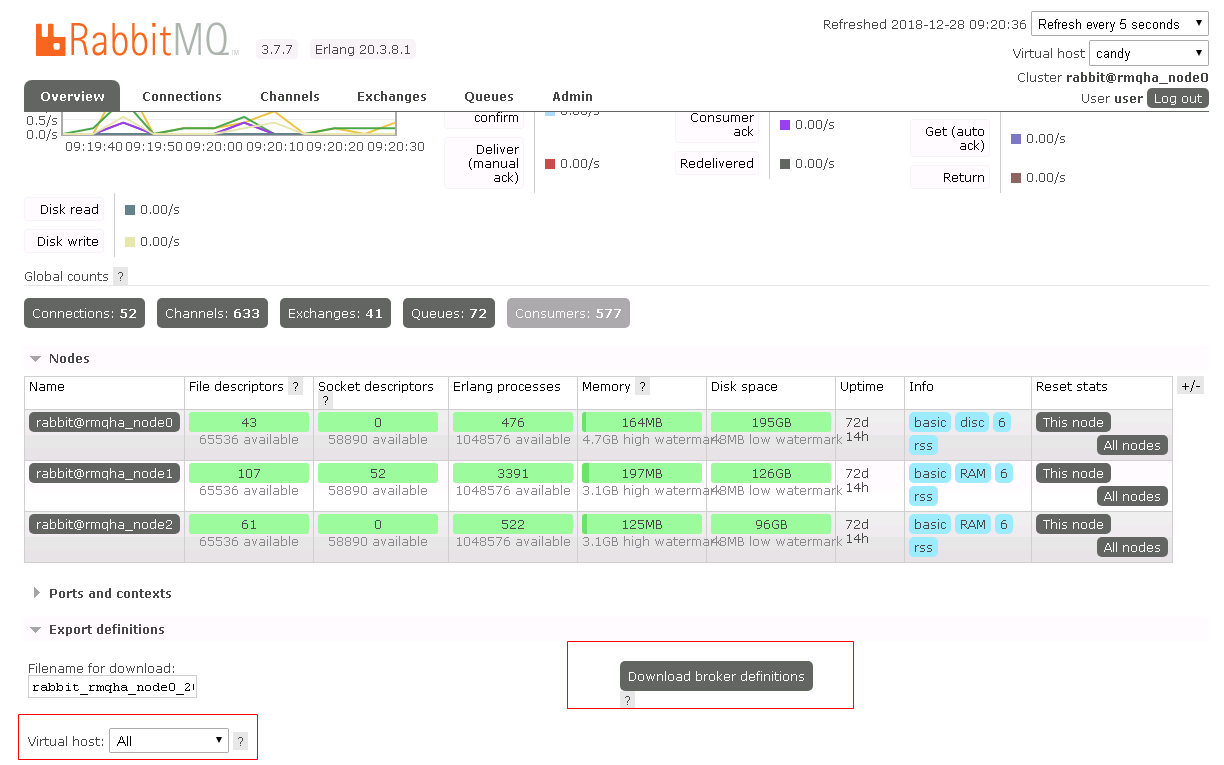
* **Setp1:** prod环境，停所有服务，**备份/**opt/rabbitmq-install/rabbitmq\_data下所有内容后删除。确认三个节点DRDB环境完好，如下：



* **Step 2:** stagging环境，登陆web界面，查看当前消息总量：



* **Step3:** stagging环境，登陆web界面，导出所有元数据信息：



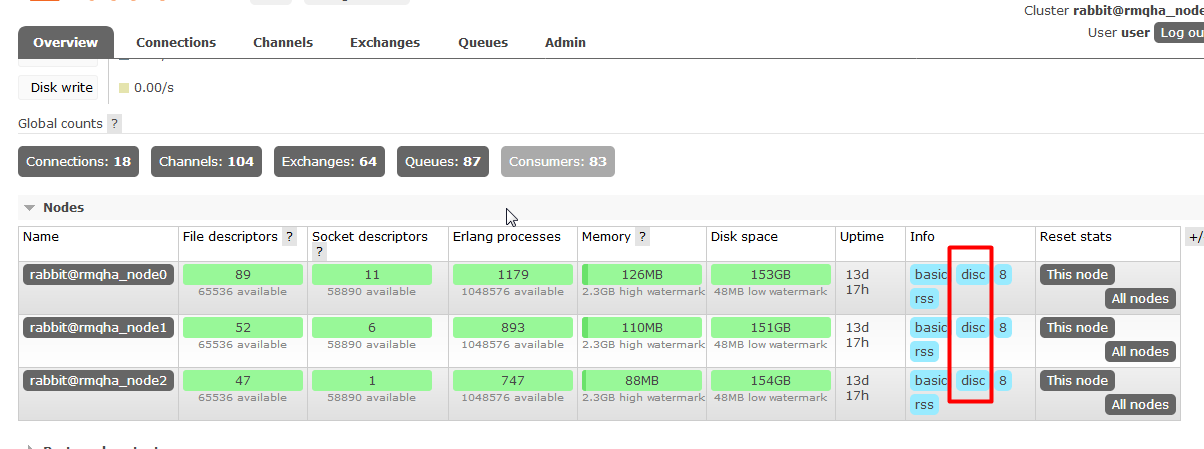
* **Step4:** staging环境，停止所有节点rabbitmq服务，停服前确认无生产消费，**备份三个节点**/opt/rabbitmq-install/rabbitmq\_data下所有内容，将备份数据打成tar包
* **Step5:** prod环境，将stagging备份数据导入到/opt/rabbitmq-install/rabbitmq\_data下，解tar包
* **Step6:** prod环境，依次启动服务，210.53 → 210.61 → 210.147
* **Step7:** prod环境，启动服务后登陆web界面，确认消息总量和stagging前一致，并导出元数据信息
* **Step8:** prod环境，比对主备机房导出元数据信息文件，确认是否完全一致
* **Step9:** prod环境，在210.61上执行：

docker exec –it rmqha\_node1 rabbitmqctl change\_cluster\_node\_type disc

* **Step10:** prod环境，在210.147上执行：

docker exec –it rmqha\_node2 rabbitmqctl change\_cluster\_node\_type disc

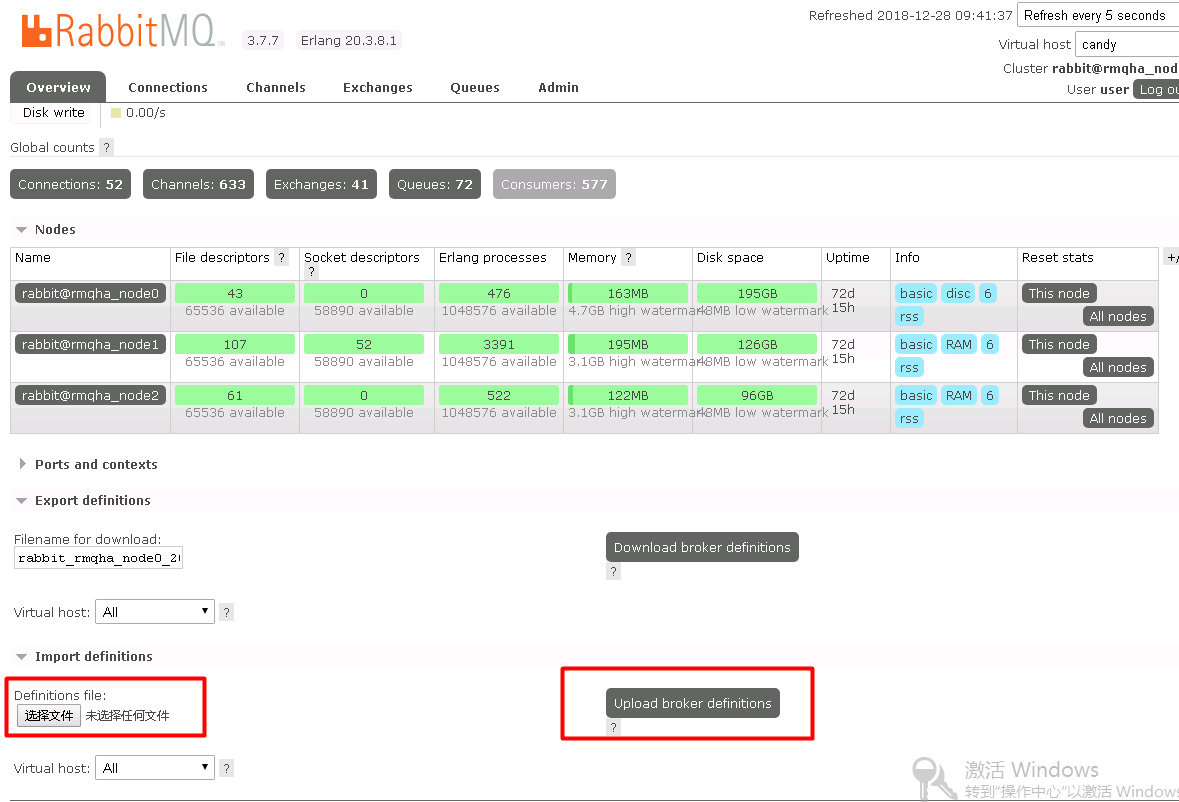
* **Step11:** prod环境，登陆web界面确认节点是否都已切换为disc



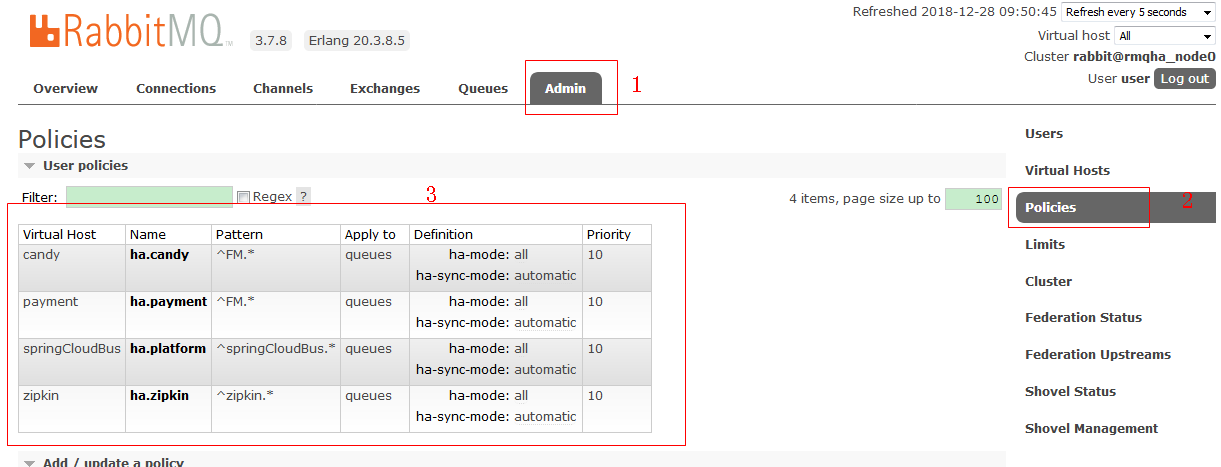
* **step12:** prod环境，导入高可用策略，文件：



登陆web界面，导入该策略



导入后，依次点击： Admin → Policies，查看策略是否导入完成



确认策略是否生效：

