

# 多IDC数据分布

--MySQL多机房部署

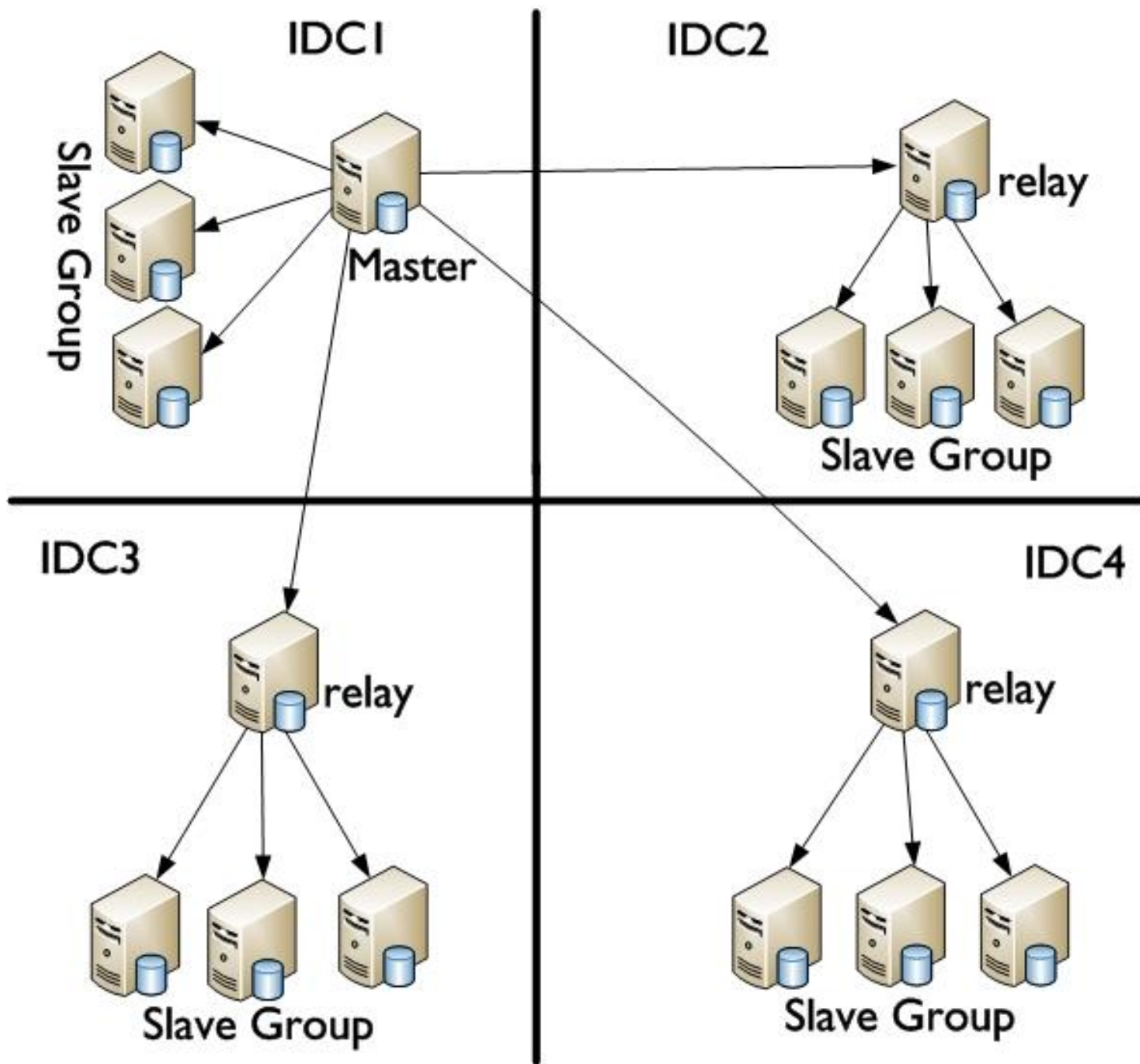
杨海朝  
@jackbillow  
jackbillow@gmail.com

2011-12-24

# 尝试 I

Master → Relay → Slave

# 尝试 I



# 特点

1. Slave和前端在一个IDC
2. DB集中写, cache本地写
3. 某一机房是核心

# 挑战

1.不适合写入量大的业务

2.Cache清理机制复杂

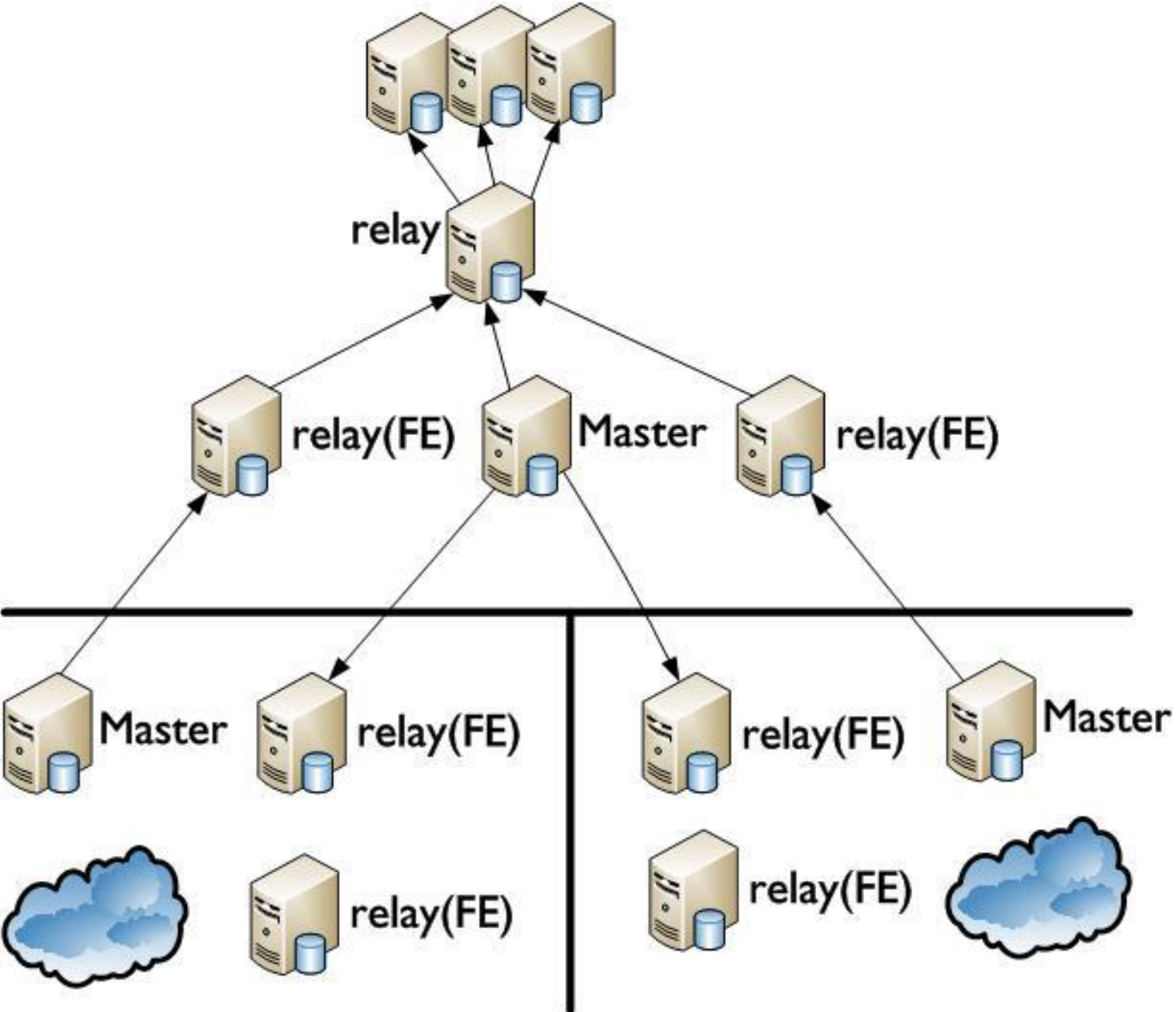
3.注意同步延时问题

4.Relay容灾

# 尝试2

MySQL federated engine

# 尝试2



# 特点

- 1.利用FE实现多master到单slave的同步
- 2.FE不提供在线服务
- 3.DB和Cache本地读本地写
- 4.N个IDC部署中每个IDC需要部署N-1个relay



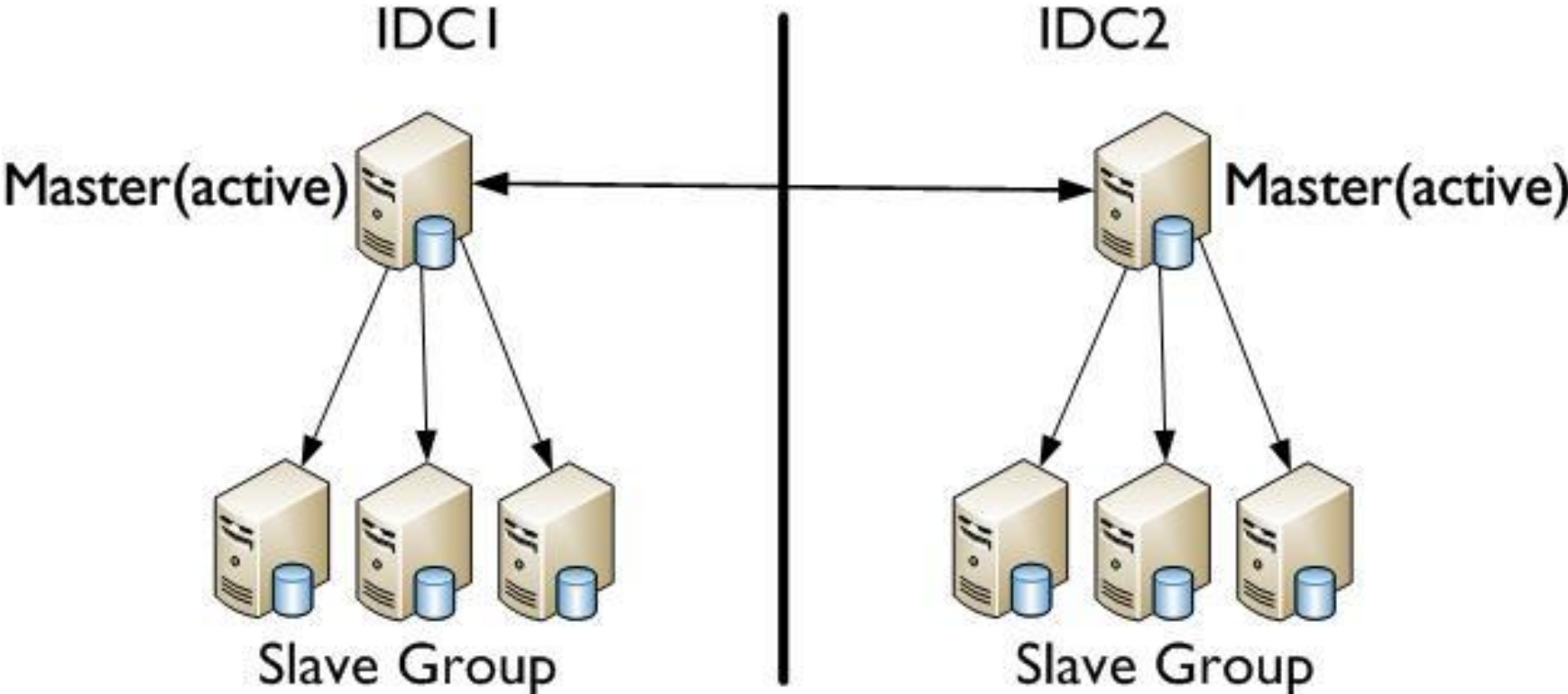
# 挑战

1. MySQL协议太重
2. 存在更新丢失问题
3. 维护难度大

# 尝试3

MySQL Master/Master

# 尝试3



# 特点

1.双写

2.容灾优势明显

# 挑战

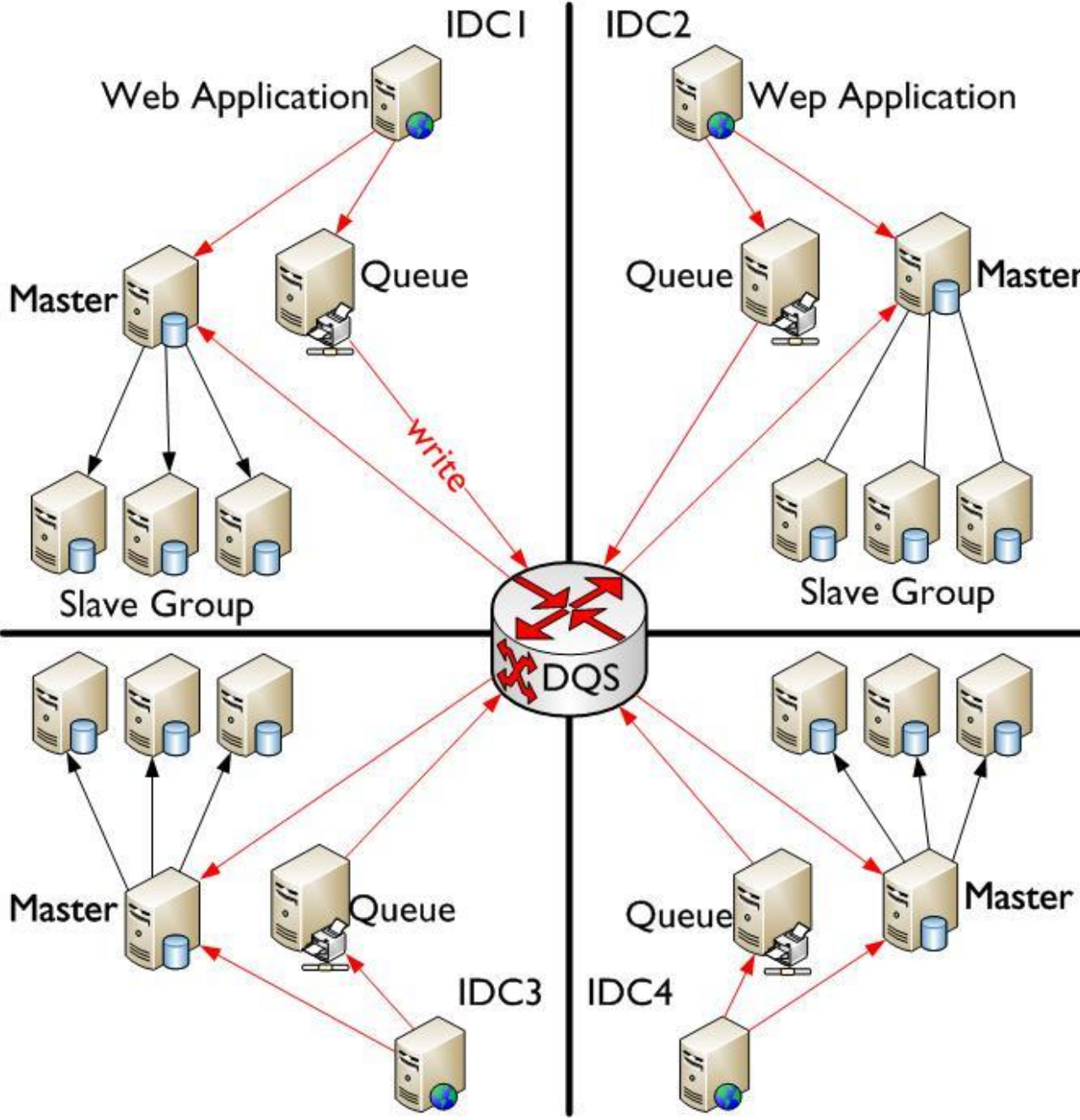
1.写节点限制在两个IDC

2.时序问题

# 尝试4

Master → Queue Service → Master

# 尝试4



# 特点

- 1.多点写入
- 2.索引和内容合并同时入队列
- 3.每个IDC完全独立



# 挑战

1.带来运维复杂

2.程序的解耦问题

# 在路上

- 1.能不分布就不要分布
- 2.部署成对IDC并且不多于4个
- 3.提高用户体验的同时解决容灾和突发流量问题
- 4.考虑好业务的时序问题
- 5.异步为王

Thank you for Coming

Q & A