Tedu.cn
达内教育

数据库管理

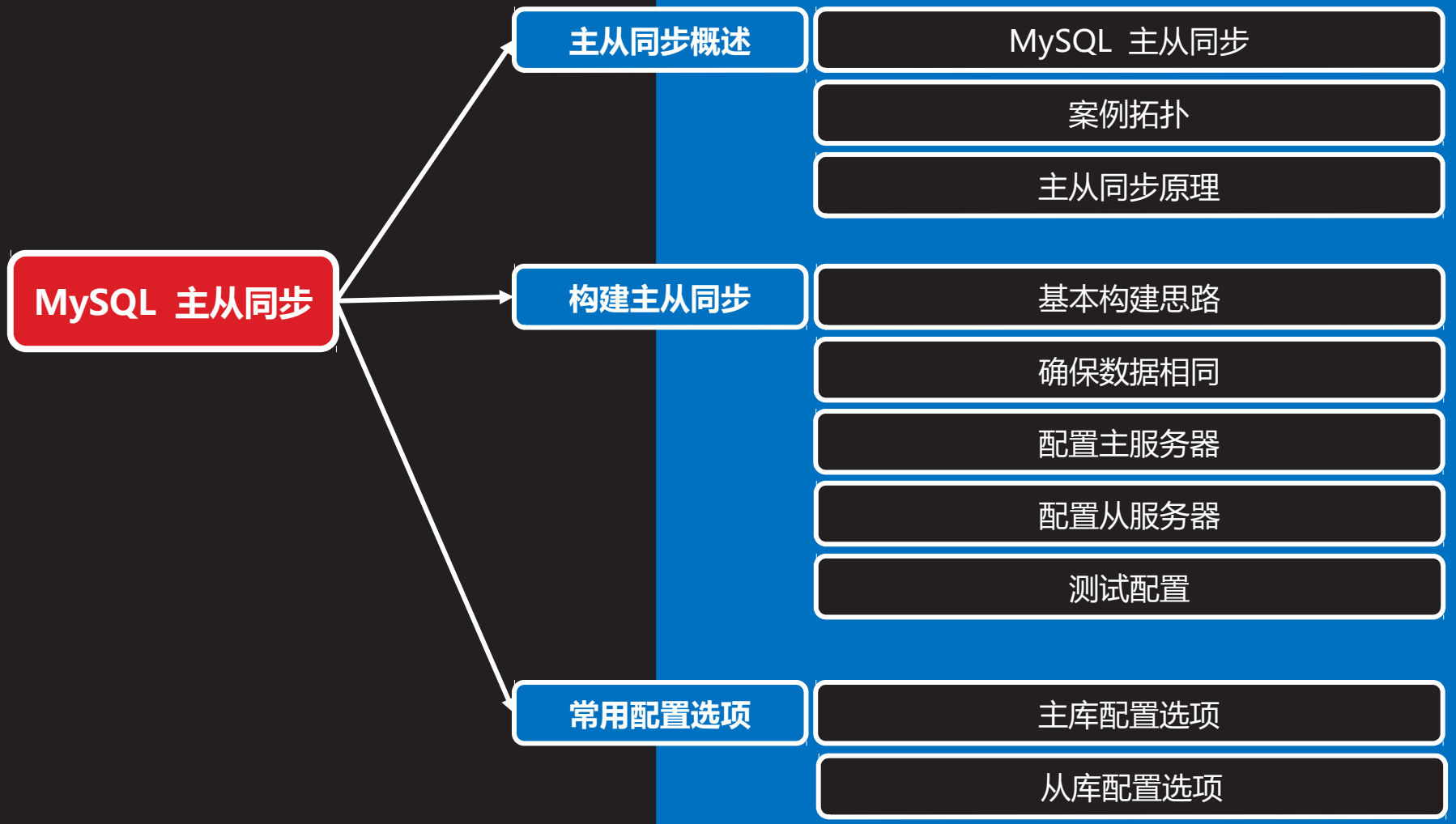
NSD DBA 进阶

DAY01

内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	MySQL 主从同步
	10:30 ~ 11:20	
	11:30 ~ 12:00	
下午	14:00 ~ 14:50	MySQL 主从同步模式
	15:00 ~ 15:50	
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑

MySQL 主从同步


MySQL 主从同步
主从同步概述
MySQL 主从同步
案例拓扑
主从同步原理
构建主从同步
基本构建思路
确保数据相同
配置主服务器
配置从服务器
测试配置
常用配置选项
主库配置选项
从库配置选项

Tedu.cn
达内教育

主从同步概述

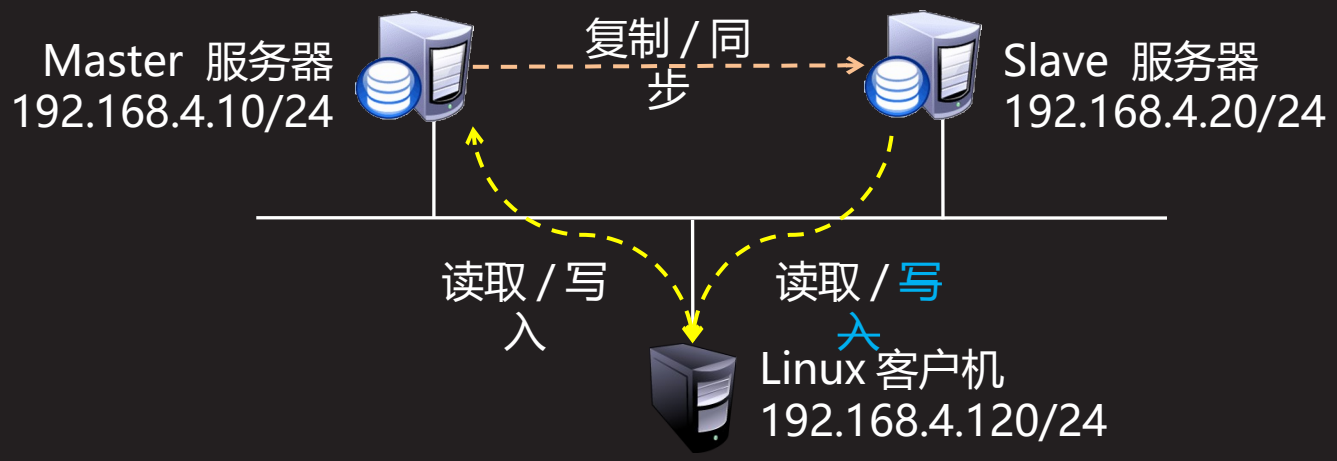
MySQL 主从同步

- 对指定库的异地同步?
- MySQL 主 --> 从复制架构的实现?
- MySQL 服务器的只读控制 ?

知识讲解

案例拓扑

- 一主、一从
  - 单向复制时，建议将从库设为只读



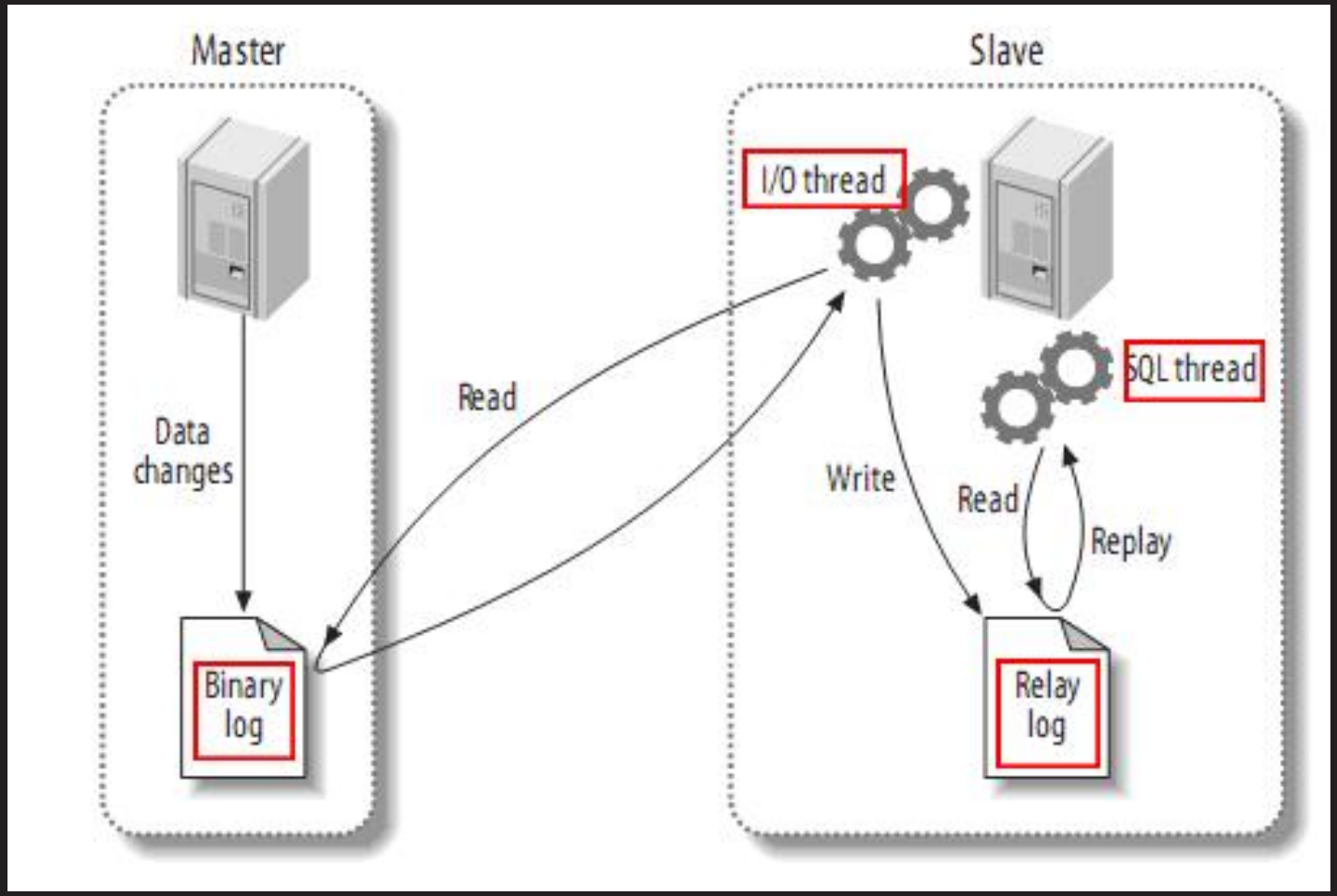
知识讲解

主从同步原理

- Master， 记录数据更改操作
  - 启用 binlog 日志
  - 设置 binlog 日志格式
  - 设置 server_id
- Slave 运行 2 个线程
  - Slave_IO： 复制 master 主机 binlog 日志文件里的 SQL 到本机的 relay-log 文件里。
  - Slave_SQL： 执行本机 relay-log 文件里的 SQL 语句，重现 Master 的数据操作。

知识讲解

主从同步原理（续 1 ）



知识讲解

Tedu.cn
达内教育

构建主从同步

基本构建思路

1. 确保数据相同
   - 从库必须要有主库上的数据。
2. 配置主服务器
   - 启用 binlog 日志及设置格式，设置 server_id，授权用户
3. 配置从服务器
   - 设置 server_id，指定主数据库服务器信息
4. 测试配置
   - 客户端连接主库，写入的数据，在连接从库的时候也能够访问到。

知识讲解

确保数据相同

- Master 服务器
  - 应包括希望同步的所有库
  - 对采用 MyISAM 的库，可离线备份

知识讲解

```
mysql> RESET MASTER;                    // 重置 binlog 日志
.. ..
[root@dbsvr1 ~]# mysqldump -u root -p -B mysql test >
mytest.sql
Enter password:                         // 验证口令
```

确保数据相同（续 1 ）

- Slave 服务器
  - 离线导入由 Master 提供的备份
  - 清空同名库（若有的话）

```
mysql> DROP DATABASE test;          // 先清理目标库
.. ..
[root@dbsvr2 ~]# scp dbsvr1:/root/mytest.sql ./
.. ..                               // 直接 scp 远程拷贝
[root@dbsvr2 ~]# mysql -u root -p < mytest.sql
Enter password:                     // 验证口令
```

知识讲解

配置主服务器

- 调整运行参数
  - 启用 binlog 及允许同步

知识讲解

```
[root@dbsvr1 mysql]# vim /etc/my.cnf
[mysqld]
log_bin=dbsvr1-bin                    // 启用 binlog 日志
server_id = 10                        // 指定服务器 ID 号
binlog_format=mixed                   // 指定日志格式
.. ..
[root@dbsvr1 mysql]# systemctl  start  mysqld        // 启动服务
```

配置主服务器（续 1 ）

- 授权备份用户
  - 允许 replicater 从 192.168.4.0/24 网段访问
  - 对所有库（默认不允许对单个库）有同步权限

知识讲解

```
[root@dbsvr1 ~]# mysql  -u  root  -p
Enter password:
.. ..
mysql> GRANT  REPLICATION  SLAVE  ON  *.*  TO
    > 'replicater'@'192.168.4.%'  IDENTIFIED  BY  'pwd123';
```

配置主服务器（续 2 ）

- 查看 Master 状态
  - 记住当前的日志文件名、偏移位置

```
mysql> SHOW  MASTER  STATUS\G
*********************** 1. row ***********************
            File: dbsvr1-bin.000004        // 日志文件名
        Position: 334                      // 偏移位置
    Binlog_Do_DB:
Binlog_Ignore_DB:
Executed_Gtid_Set:
.. ..
```

知识讲解

配置从服务器

- 调整运行参数
  - 启用 binlog 及允许同步，启用只读模式

```
[root@dbsvr2 mysql]# vim  /etc/my.cnf
[mysqld]
log_bin=dbsvr2-bin                    // 启用 binlog 日志
server_id = 20                        // 指定服务器 ID 号
sync-binlog=1                         // 允许日志同步
read_only=1              // 只读模式，同步及 SUPER 权限用户例外
.. ..
[root@dbsvr2 mysql]# systemctl   start  mysqld    // 启动服务
```

知识讲解

配置从服务器（续 1 ）

- 发起同步操作
  - 指定 Master 相关参数

```
mysql> CHANGE MASTER TO MASTER_HOST='192.168.4.10',
    -> MASTER_USER='replicater',
    -> MASTER_PASSWORD='pwd123',
    -> MASTER_LOG_FILE='dbsvr1-bin.000004',      // 日志文件
    -> MASTER_LOG_POS=334;                       // 偏移位置
.. ..
mysql> START SLAVE;                              // 启动复制
.. ..
```

1. Master 信息会自动保存到 /var/lib/mysql/master.info 文件
2. 以后要更改 Master 信息时，应先 STOP SLAVE;

知识讲解

配置从服务器（续 2 ）

- 查看 Slave 状态
  - 确认 IO 线程、 SQL 线程都已运行

```
mysql> SHOW  SLAVE  STATUS\G
*********************** 1. row ***********************
 Slave_IO_State: Waiting for master to send event
                Master_Host: 192.168.4.10
                Master_User: replicater
                .. ..
          Slave_IO_Running: Yes          //IO 线程已运行
          Slave_SQL_Running: Yes         //SQL 线程已运行
```

知识讲解

配置从服务器（续 3 ）

- 相关文件

文件名	说明
master.info	连接主服务器信息
relay-log.info	中继日志信息
主机名 -relay-bin.xxxxxx	中继日志
主机名 -relay-bin.index	中继日志索引文件

知识讲解

测试配置

- 在 Master 上操纵数据
  - 新建 newdb 库、 newtbl 表
  - 任意插入几条表记录
- 在 Slave 上查看数据更改情况
  - 确认新建的 newdb 库、 newtbl 表
  - 列出 newtbl 表的所有记录

知识讲解

案例 1： MySQL 一主一从

1. 构建 主 --> 从 复制结构
2. 将主机 192.168.4.20 配置为主机 192.168.4.10 的从库

课堂练习

Tedu.cn
达内教育

常用配置选项

主库配置选项

- 适用于 Master 服务器

选 项	用 途
binlog_do_db=name	设置 Master 对哪些库记日志
binlog_ignore_db=name	设置 Master 对哪些库不记日志

知识讲解

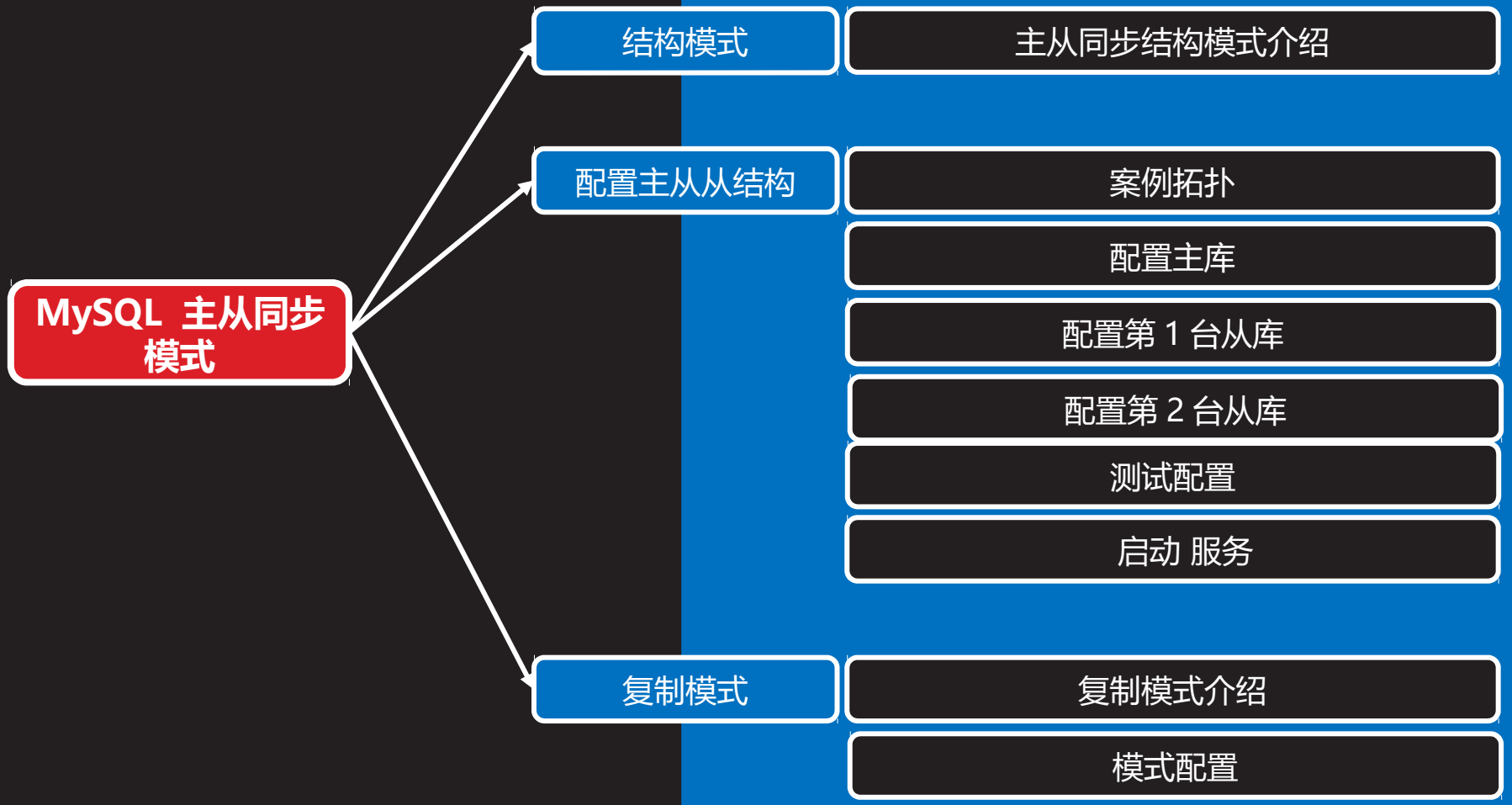
从库配置选项

- 适用于 Slave 服务器

选 项	用 途
log_slave_updates	记录从库更新，允许链式复制（A-B-C）
relay_log=dbsvr2-relay-bin	指定中继日志文件名
replicate_do_db=mysql	仅复制指定库，其他库将被忽略，此选项可设置多条（省略时复制所有库）
replicate_ignore_db=test	不复制哪些库，其他库将被忽略， ignore-db 与 do-db 只需选用其中一种

知识讲解

MySQL 主从同步模式



Tedu.cn
达内教育

结构模式

主从同步结构模式介绍

- 基本应用
  - 单向复制：主 --> 从
- 扩展应用
  - 链式复制：主 --> 从 --> 从
  - 双向复制：主 <--> 从
  - 放射式复制：从 <-- 主 --> 从

从

知识讲解

Tedu.cn
达内教育

配置主从从结构

拓扑结构

- 主从从

知识讲解

配置主库

- 用户授权
- 启用 binlog 日志
- 重启服务

知识讲解

```
Mysql> grant replication slave on *.* to 用户名
@" 从库 IP 地址" identified by "密码";

]# Vim /etc/my.cnf
[mysqld]
log-bin= 日志名
Server_id=id 号
binlog_format= "mixed"
:wq

]# systemctl restart mysqld
```

配置第 1 台从库

- 修改配置文件
- 用户授权
- 指定主库信息
- 启动 slave 进程
- 查看状态信息

```
]# Vim  /etc/my.cnf
[mysqld]
Server_id=id 号
log-bin= 日志名
Binlog_format= “mixed”
log_slave_updates
:wq
]# systemctl  restart  mysqld
```

```
Mysql>  grant  replication  slave  on  *.*  to  用户名 @” 第 2 台从库的 IP 地址”
  identified  By  “密码”;

  mysql> CHANGE MASTER TO MASTER_HOST= ‘主库 IP 地
  址 ',
      -> MASTER_USER= ‘用户名 ',
      -> MASTER_PASSWORD= ‘密码 ',
      -> MASTER_LOG_FILE= ‘binlog 日志文件名 ',
      -> MASTER_LOG_POS= 偏移量 ;
  Mysql>  start  slave;
  Mysql>  show  slave  status\G;
```

知识讲解

配置第 2 台从库

- 修改配置文件
- 指定主库信息
- 启动 slave 进程
- 查看状态信息

```
]# vim  /etc/my.cnf
[mysqld]
server_id=id 号
:wq
]# systemctl   restart  mysqld
```

```
mysql> CHANGE MASTER TO MASTER_HOST= ‘第 1 台从库 IP 地址 ',
    -> MASTER_USER= ‘用户名 ',
    -> MASTER_PASSWORD= ‘密码 ',
    -> MASTER_LOG_FILE= ‘binlog 日志文件名 ',
    -> MASTER_LOG_POS= 偏移量 ;
Mysql>  start  slave;
Mysql>  show   slave  status\G;
```

知识讲解

测试配置

- 在主库授权访问数据的用户
- 在客户端访问主库，建库、建表、插入记录
- 在客户端访问第 1 台从库服务器可以看到和主库同样的记录。
- 在客户端访问第 2 台从库服务器可以看到和主库同样的记录。

知识讲解

```
Mysql>  grant  all  on  库 .*  to  用户 @“ 客户端地址”  identified by  “ 密码” ;

]# mysql  -h 数据库 ip 地址  -u 用户名  -p  密码
Mysql> select  *  from  库 . 表;
```

案例 2： 配置主从从同步结构

具体要求如下：

- 配置主机 192.168.4.51 为主数据库服务器
- 配置主机 192.168.4.52 为 51 主机的从库服务器
- 配置主机 192.168.4.53 为 52 主机的从库服务器
- 客户端连接主数据库服务器 51 主机创建的数据，连接 52 和 53 主机时，也可以访问到库、表、记录。

课堂练习

Tedu.cn
达内教育

复制模式

复制模式介绍

- 异步复制（Asynchronous replication）
  - 主库在执行完客户端提交的事务后会立即将结果返给客户端，并不关心从库是否已经接收并处理。
- 全同步复制（Fully synchronous replication）
  - 当主库执行完一个事务，所有的从库都执行了该事务才返回给客户端。
- 半同步复制（Semisynchronous replication）
  - 介于异步复制和全同步复制之间，主库在执行完客户端提交的事务后不是立刻返回给客户端，而是等待至少一个从库接收到并写到 relay log 中才返回给客户端。

知识讲解

模式配置

- 查看是否允许动态加载模块
  - 默认允许

mysql> show variables like "have _ dynamic _ loading";

```
mysql> show variables like "have_dynamic_loading";
+----------------------+-------+
| Variable_name        | Value |
+----------------------+-------+
| have_dynamic_loading | YES   |
+----------------------+-------+
1 row in set (0.00 sec)
```

知识讲解

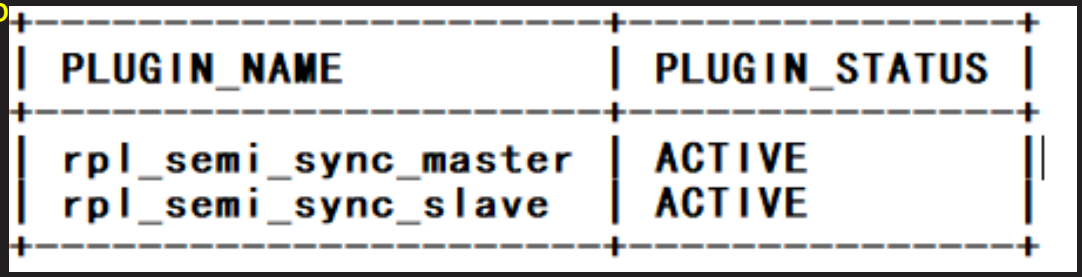
模式配置（续 1 ）

- 命令行加载插件
  - 用户需有 SUPER 权限

主库： mysql> INSTALL PLUGIN rpl _ semi _ sync _ master SONAME 'semisync_master.so';

从库： mysql> INSTALL PLUGIN rpl _ semi _ sync _ slave SONAME 'semisync_slave.so';

查看： mysql> SELECT PLUGIN_NAME, PLUGIN_STATUS FROM INFORMATION_SCHEMA.PLUGINS WHERE PLUGIN_NAME LIKE '%semi%';



PLUGIN_NAME	PLUGIN_STATUS
rpl_semi_sync_master	ACTIVE
rpl_semi_sync_slave	ACTIVE

知识讲解

模式配置（续 2 ）

- 启用半同步复制
  - 在安装完插件后，半同步复制默认是关闭的

主： mysql> SET GLOBAL rpl_semi_sync_master_enabled = 1;

从： mysql> SET GLOBAL rpl_semi_sync_slave_enabled = 1;

查看： mysql> show variables like "rpl_semi_sync_%_enabled";

```
mysql> show variables like "rpl_semi_sync_%_enabled";
+-------------------------------+-------+
| Variable_name                 | Value |
+-------------------------------+-------+
| rpl_semi_sync_master_enabled  | ON    |
| rpl_semi_sync_slave_enabled   | ON    |
+-------------------------------+-------+
```

知识讲解

模式配置（续 3 ）

- 配置文件永久启用半同步复制
  - 命令配置临时配置，重启服务会失效
  - 修改后需要重启服务
  - 写在主配置文件 /etc/my.cnf 的 [mysqld] 下方

主:
```
plugin-load=rpl_semi_sync_master=semisync_master.so
rpl_semi_sync_master_enabled=1
```

从:
```
plugin-load=rpl_semi_sync_slave=semisync_slave.so
rpl_semi_sync_slave_enabled=1
```

知识讲解

模式配置（续 4 ）

- 在有的高可用架构下， master 和 slave 需同时启动
  - 以便在切换后能继续使用半同步复制

知识讲解

```
plugin-load =
"rpl_semi_sync_master=semisync_master.so;rpl_semi_sync_slave=semis
ync_slave.so  "

rpl-semi-sync-master-enabled = 1
rpl-semi-sync-slave-enabled = 1
```

```
mysql> show variables like "rpl_semi_sync_%_enabled";
+-------------------------------+-------+
| Variable_name                 | Value |
+-------------------------------+-------+
| rpl_semi_sync_master_enabled  | ON    |
| rpl_semi_sync_slave_enabled   | ON    |
+-------------------------------+-------+
2 rows in set (0.00 sec)
```

案例 3： 配置半同步复制模式

具体要求如下：

- 开启案例 1 主库 192.168.4.51 半同步复制模式
- 开启案例 1 从库 192.168.4.52 半同步复制模式
- 开启案例 1 从库 192.168.4.53 半同步复制模式
- 查看半同步复制模式是否开启。

课堂练习

总结和答疑

Tedu.cn
达内教育

配置主从同步

问题现象 1

- Slave_IO 线程没有运行
  - 报错： Slave_IO_Running: No

```
[root@room1 ~]# mysqld -uplj -p123
mysql > show slave status \G;

    Slave_IO_Running: No

        Last_IO_Error : 报错信息 .....
```

知识讲解

故障分析及排除

- 原因分析
  - 连接不上 master 数据库服务器
- 解决办法
  - 检查物理连接（ping）、检查授权用户
  - 检查是否有防火墙规则 （service iptables stop）
  - 关闭 SElinux（setenforce 0)
  - 或是 binlog 日志文件指定错误（日志名或 pos 节点）
  - stop slave；change master to 选项 = 值 ；start slave ;

知识讲解

问题现象 2

- Slave_SQL 线程没有运行
  - 报错： Slave_SQL_Running: No

```
[root@room1 ~]# mysqld  -uplj  -p123
mysql >  show  slave  status \G;

          Slave_SQL_Running: No

     Last_SQL_Error :  报错信息 .....
```

知识讲解

故障分析及排除

- 原因分析
  - 执行本机中继日志里的 sql 命令时， sql 命令使用的库、表 或记录在本机不存在。
- 解决办法
  - 1 、 stop slave;
  - 2 、创建或恢复需要用到的库或表
  - 3 、 start slave;

知识讲解