

# DBA进阶

**NSD DBA2**

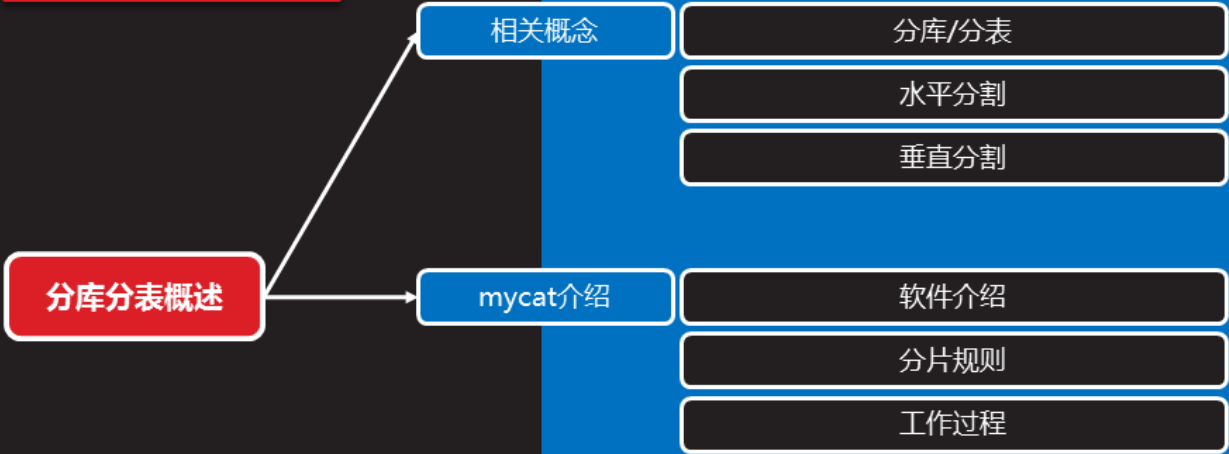
**DAY05**

# 内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	分库分表概述
	10:30 ~ 11:20	
	11:30 ~ 12:00	配置mycat
下午	14:00 ~ 14:50	
	15:00 ~ 15:50	
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑



## 分库分表概述



# 相关概念

## 分库分表

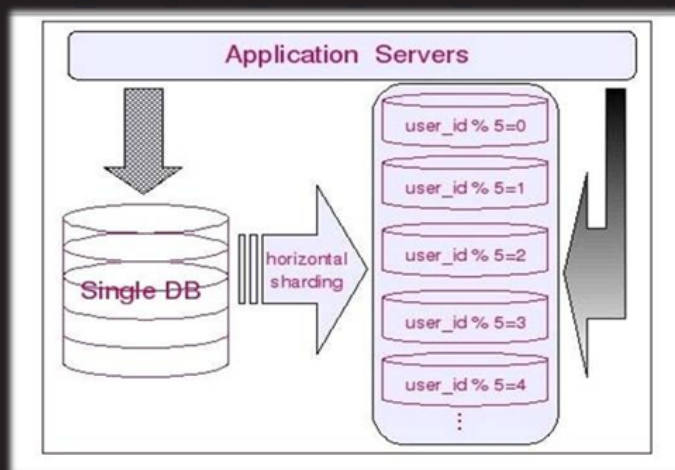
- 什么是分库分表
  - 将存放在一个数据库（主机）中的数据，按照特定方式进行拆分，分散存放多个数据库（主机）中，以达到分散单台设备负载的效果



# 水平分割

- 横向切分
  - 按照表中某个字段的某种规则，把表中的许多记录按行切分，分散到多个数据库中

知识讲解



## mycat介绍

## 软件介绍

知识讲解

- mycat 是基于Java的分布式数据库系统中间层，为高并发环境的分布式访问提供解决方案
  - 支持JDBC形式连接
  - 支持MySQL、Oracle、Sqlserver、Mongodb等
  - 提供数据读写分离服务
  - 可以实现数据库服务器的高可用
  - 提供数据分片服务
  - 基于阿里巴巴Cobar进行研发的开源软件
  - 适合数据大量写入数据的存储需求



## 分片规则

知识讲解

- mycat支持提供10种分片规则

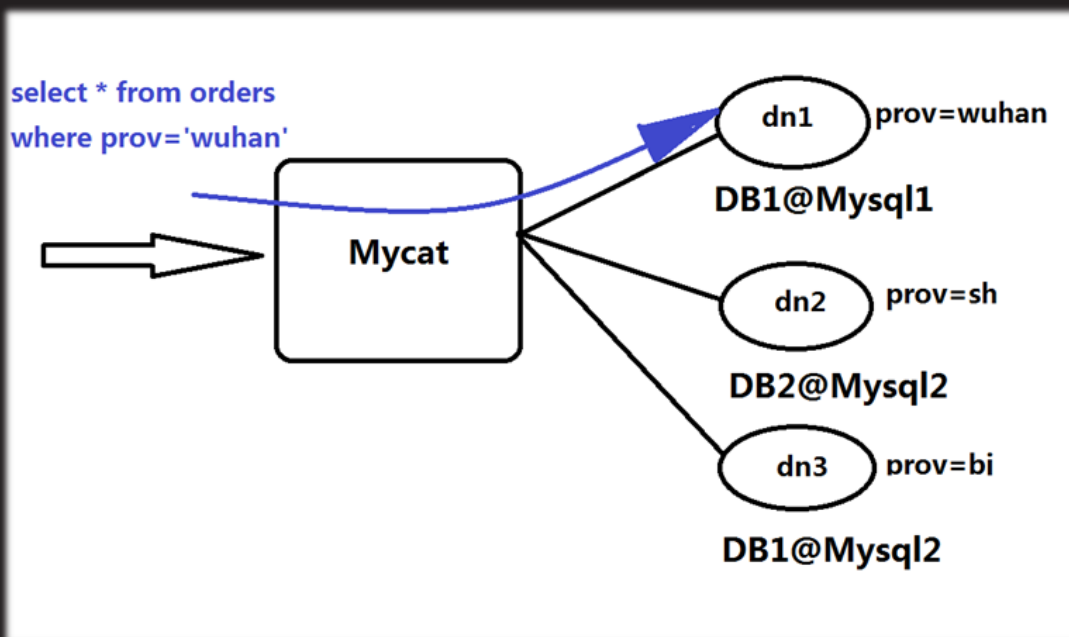
- 1 枚举法 sharding-by-intfile
- 2 固定分片 rule1
- 3 范围约定 auto-sharding-long
- 4 求模法 mod-long
- 5 日期列分区法 sharding-by-date

- 6 通配取模 sharding-by-pattern
- 7 ASCII码求模通配 sharding-by-prefixpattern
- 8 编程指定 sharding-by-substring
- 9 字符串拆分hash解析 sharding-by-stringhash
- 10 一致性hash sharding-by-murmur



## 工作过程

知识讲解

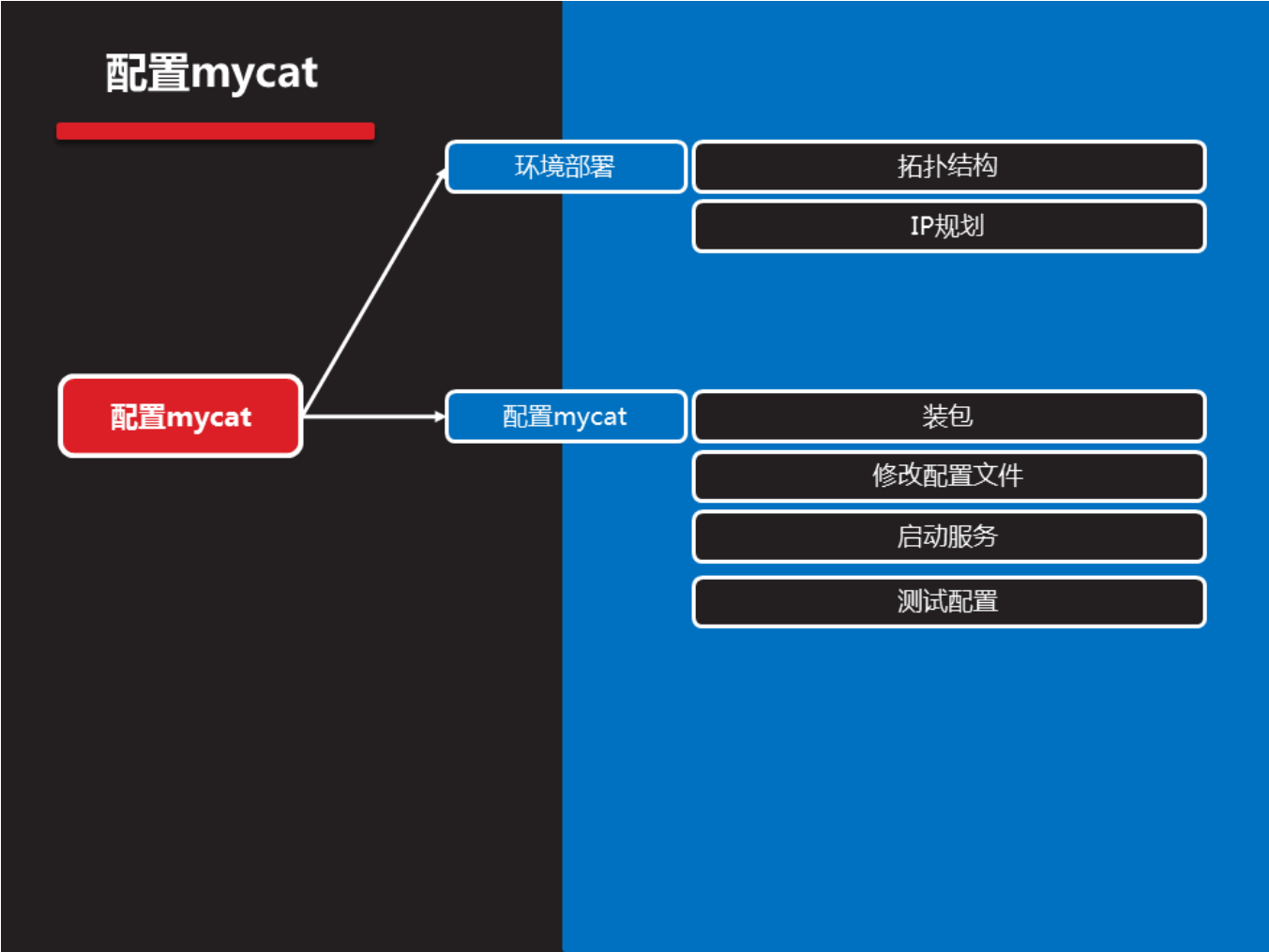


## 工作过程（续1）

知识讲解

- 当mycat收到一个SQL查询时
  - 先解析这个SQL查找涉及到的表
  - 然后看此表的定义，如果有分片规则，则获取SQL里分片字段的值，并匹配分片函数，获得分片列表
  - 然后将SQL发往这些分片去执行
  - 最后收集和处理所有分片结果数据，并返回到客户端



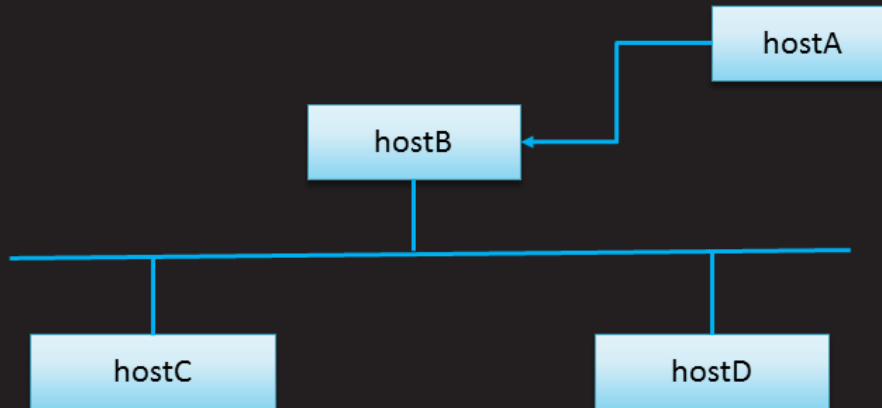


# 环境部署

# 拓扑结构

- 数据分片拓扑结构

知识讲解





# 配置mycat

## 装包

- 安装JDK
  - 系统自带的即可

```
[root@localhost ~]# rpm -qa | grep -i jdk
java-1.8.0-openjdk-1.8.0.65-3.b17.el7.x86_64
java-1.8.0-openjdk-headless-1.8.0.65-3.b17.el7.x86_64
```

- 安装mycat服务软件包

```
[root@localhost ~]# tar -zxf mycat-server-1.4-beta-20150604171601-linux.tar.gz //免安装，解压即可使用
[root@localhost ~]# mv mycat/ /usr/local/
```

```
[root@localhost ~]# ls /usr/local/mycat/
bin catlet conf lib logs version.txt
```

# 修改配置文件

知识讲解

- 目录结构说明
  - bin //mycat命令, 如 启动 停止 等
  - catlet //扩展功能
  - conf //配置文件
  - lib //mycat使用的jar
  - log //mycat启动日志和运行日志
  - wrapper.log //mycat服务启动日志
  - mycat.log //记录SQL脚本执行后的报错内容



# 修改配置文件 ( 续1 )

知识讲解

- 重要配置文件说明
  - server.xml //设置连mycat的账号信息
  - schema.xml //配置mycat的真实库表
  - rule.xml //定义mycat分片规则
- 配置标签说明
  - <user>.. ..</user> //定义连mycat用户信息
  - <datanode>.. ..</datanode>  
//指定数据节点
  - <datahost>.. ..</datahost>  
//指定数据库地址及用户信息



## 修改配置文件（续2）

- 修改配置文件/usr/local/mycat/conf/server.xml

```
<user name= "test" >                                //连mycat的用户名
  <property name= "password" >test</property>        //对应密码
  <property name="schemas">TESTDB</property>
</user>
<user name="user">
  <property name="password">user</property>
  <property name="schemas">TESTDB</property>
  <property name="readOnly">true</property>          //定义只读
</user>
```

知识讲解



## 修改配置文件（续5）

知识讲解

- 修改数据库服务器配置文件
  - 添加对应设置后重启mysqld服务
  - 添加授权用户
  - 创建存储数据对应的库db1 、 db2

```
# vim /etc/my.cnf
[mysqld]
...
lower_case_table_names = 1           //表名忽略大小写

[root@localhost ~]# systemctl restart mysqld

mysql> grant all on *.* to admin@ "%" identified by "123456";
//添加授权访问用户
```



## 启动服务

- 启动服务
  - 指定java路径、添加PATH路径、启动服务

知识讲解

```
[root@localhost ~]# sed -n '4,5p' /usr/local/mycat/conf/wrapper.conf
# Java Application
wrapper.java.command=java
[root@localhost ~]#
[root@localhost ~]# which java
/usr/bin/java

[root@localhost ~]# echo "export PATH=/usr/local/mycat/bin" >> /etc/profile
[root@localhost ~]# source /etc/profile

[root@localhost ~]# mycat --help
Usage: /usr/local/mycat/bin/mycat { console | start | stop | restart | status | dump }
[root@localhost ~]#

[root@localhost ~]# mycat start
Starting Mycat-server...
[root@localhost ~]#
[root@localhost ~]# netstat -tnalp | grep :8066
tcp6      0      0 :::8066          :::*              LISTEN     4524/java
[root@localhost ~]#
```



# 测试配置

- 在客户端连接mycat服务器

– mysql -h服务器地址 -P端口 -u用户名 -p密码

```
[root@room9pc17 ~]# mysql -h192.168.4.56 -P8066 -utest -ptest
MySQL [(none)]> show databases;
```

```
+-----+
| DATABASE |
+-----+
| TESTDB   |
+-----+
1 row in set (0.00 sec)
```

```
[root@room9pc17 ~]# mysql -h192.168.4.56 -P8066 -utest -ptest
MySQL [(none)]> show databases;
+-----+
| DATABASE |
+-----+
| TESTDB   |
+-----+
1 row in set (0.00 sec)
MySQL [(none)]> use TESTDB;
```

知识讲解

案例1 搭建mycat公共服务器

# 总结和答疑



# 分片规则

# 分片规则

- mycat支持10种分片规则

知识讲解

- 1 枚举法 sharding-by-intfile
- 2 固定分片 rule1
- 3 范围约定 auto-sharding-long
- 4 求模法 mod-long
- 5 日期列分区法 sharding-by-date

- 6 通配取模 sharding-by-pattern
- 7 ASCII码求模通配 sharding-by-prefixpattern
- 8 编程指定 sharding-by-substring
- 9 字符串拆分hash解析 sharding-by-stringhash
- 10 一致性hash sharding-by-murmur

