Gdevops

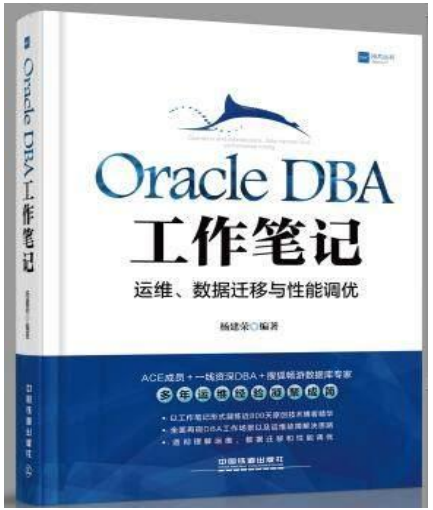
全球敏捷运维峰会

运维DBA职业发展与技术成长建议

演讲人：杨建荣

自我介绍

- 竞技世界资深DBA专家
- Oracle ACE
- DBAplus联合发起人、YEP成员
- Oracle 10g OCP,OCM ，MySQL OCP
- 对shell , Java有一定的功底
- 曾在中国数据库大会,QCon演讲,DAMS峰会演讲
- 坚持每天写点东西，已连续坚持1300多天
- 微信公众号：**jianrong-notes**



《Oracle DBA 工作笔记》作者

分享思路

01 学习周期和难度

02 工程师模型

03 数据库技术发展现状

04 数据库版本规划

05 高可用方案对比

06 解读MHA实现细节

07 剖析MySQL执行计划

08 DBA进阶之路

学习周期和难度

	Oracle	MySQL
数据库类型	商业闭源	开源
功能完善情况	非常齐全	比较齐全
学习周期	长	较短
学习难度（入门）	难	容易
学习难度（深入）	难	更难
Oracle到MySQL	NA	相对容易
MySQL到Oracle	难	NA
深度进阶	内核，调试	源码定制，改造

工程师模型

鹰眼，狮心，绣花手

自查表现	级别
独立完成工作	只是基本技能
快速解决问题	这是经验的积累和工作效率提升
避免问题	问题解决在初始阶段，这是看待需求和问题的深度
开拓创新	判别哪些不能做、哪些能做、怎么做更好

数据库技术发展现状

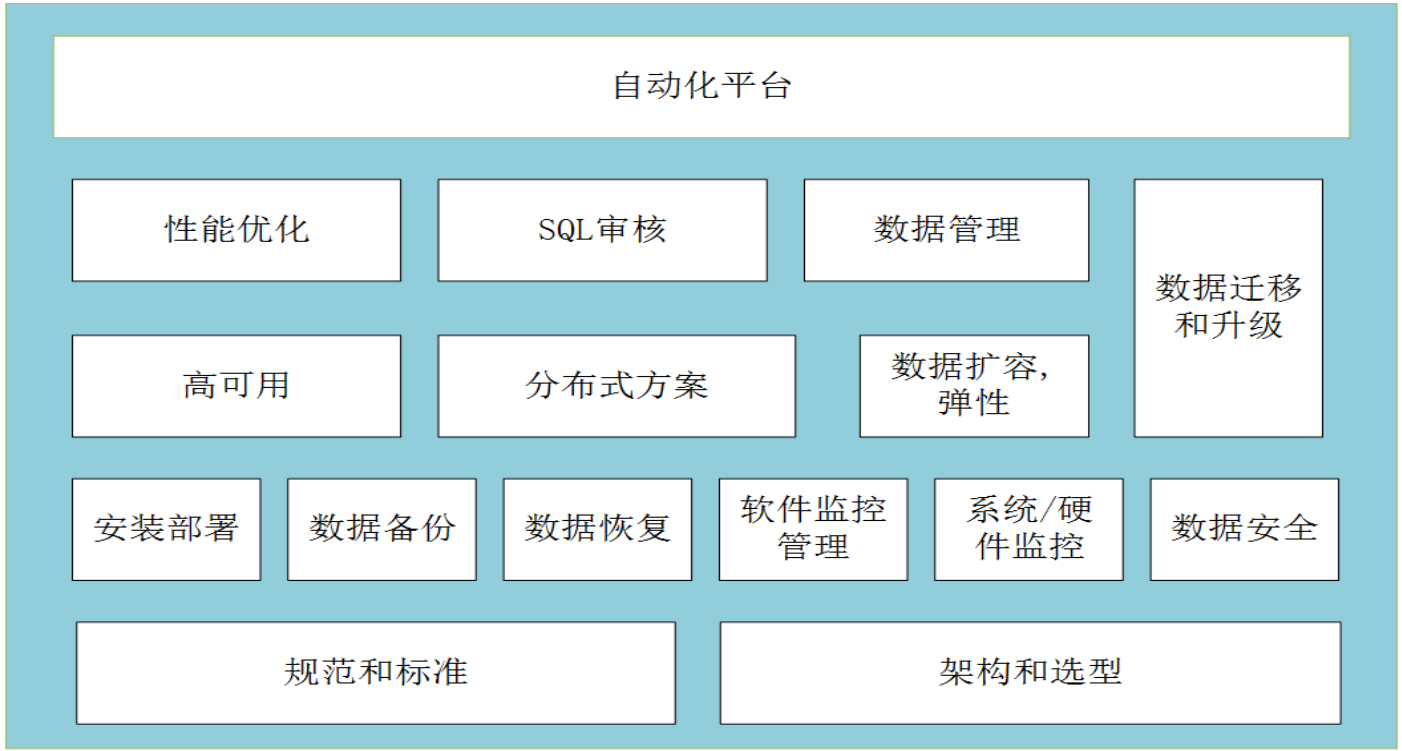
通过数据库参数了解技术变化- DB-Engines数据

The most popular database management systems

November 2017		Score
1.	Oracle	1360
2.	MySQL	1322
3.	Microsoft SQL Server	1215
4.	PostgreSQL	380
5.	MongoDB	330

- Oracle最新版本12.2,18c
- MySQL 最新版本5.7.20
- Oracle亮点特性 ：自治数据库
- MySQL亮点特性：查询优化，MGR

数据库方向规划


自动化平台
性能优化
SQL审核
数据管理
数据迁移和升级
高可用
分布式方案
数据扩容，弹性
安装部署
数据备份
数据恢复
软件监控管理
系统/硬件监控
数据安全
规范和标准
架构和选型

数据库版本规划

版本选型

- 5.5
- 5.6
- **5.7**（5.7.13, **5.7.16**, 5.7.19）
- 8.0

软件选型

- 社区版
- Percona分支
- MariaDB（版本标识不同）

版本选型

- 11gR2(11.2.0.4)
- 12cR1(12.1)
- 12cR2(12.2.0.1)

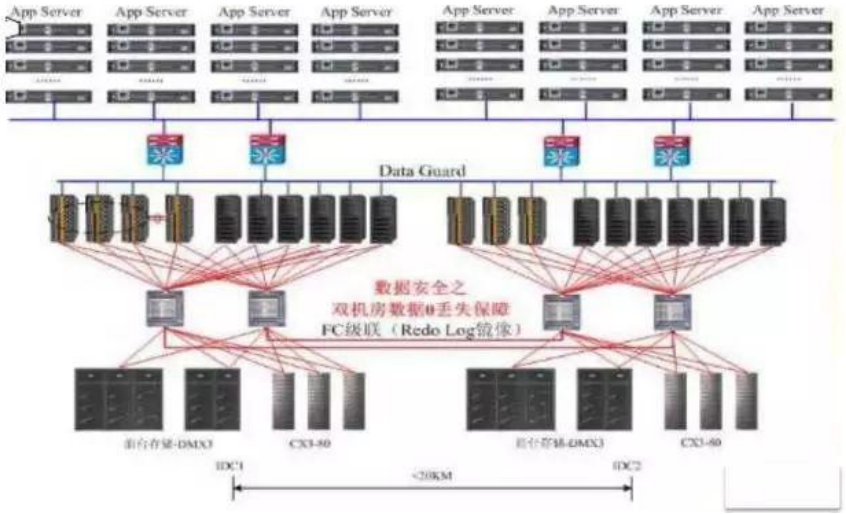
软件选型

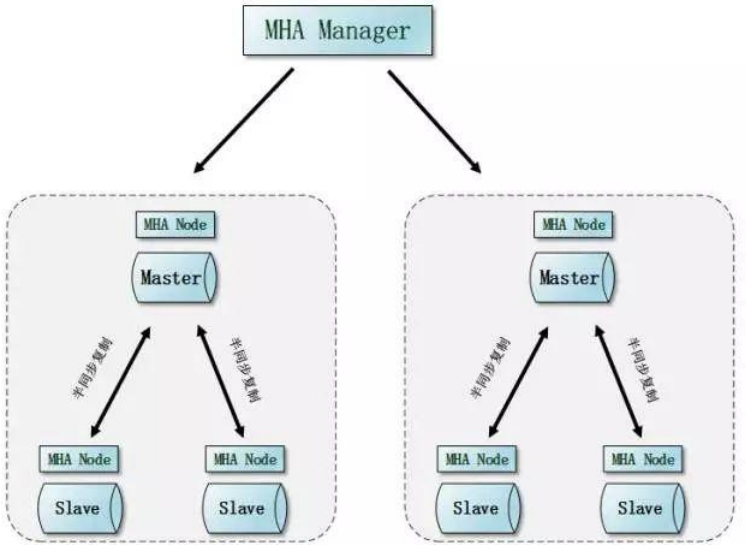
- 企业版
- 标准版

MySQL方案高可用率

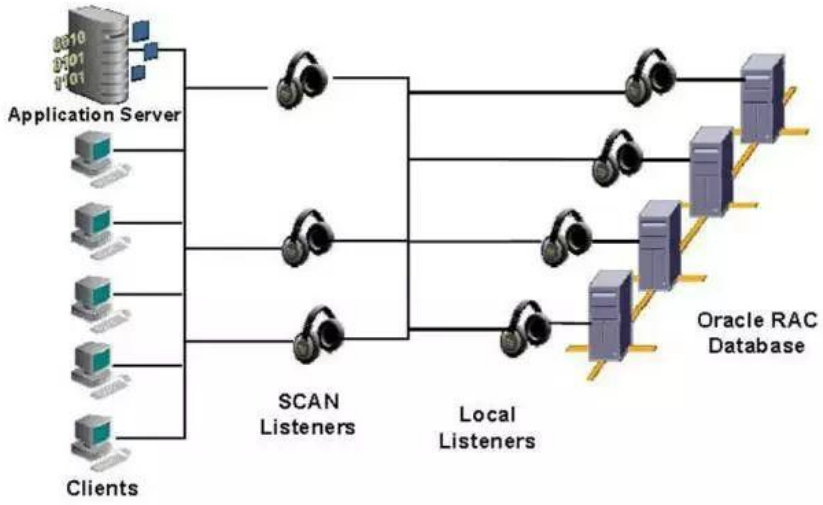
Method	Level of Availability
Simple replication	98-99.9%
Master-Master/MMM	99%
SAN	99.5-99.9%
DRBD,MHA,Tungsten Replicator	99.9%
NDBCluster,Galera Cluster	99.999%

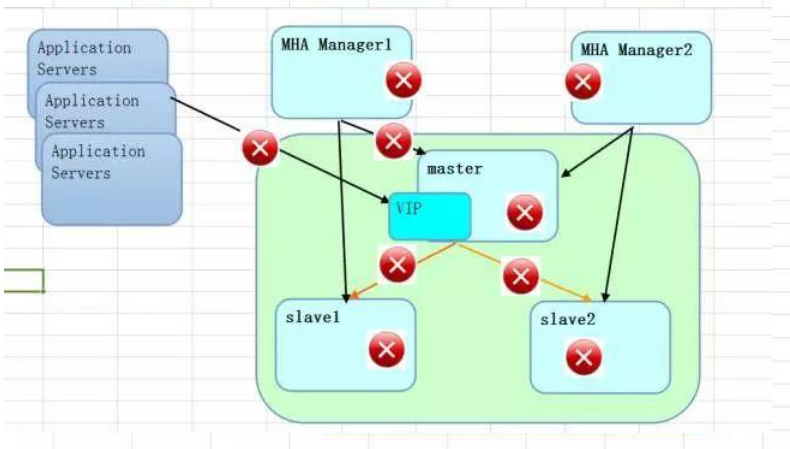
Oracle和MySQL的高可用方案

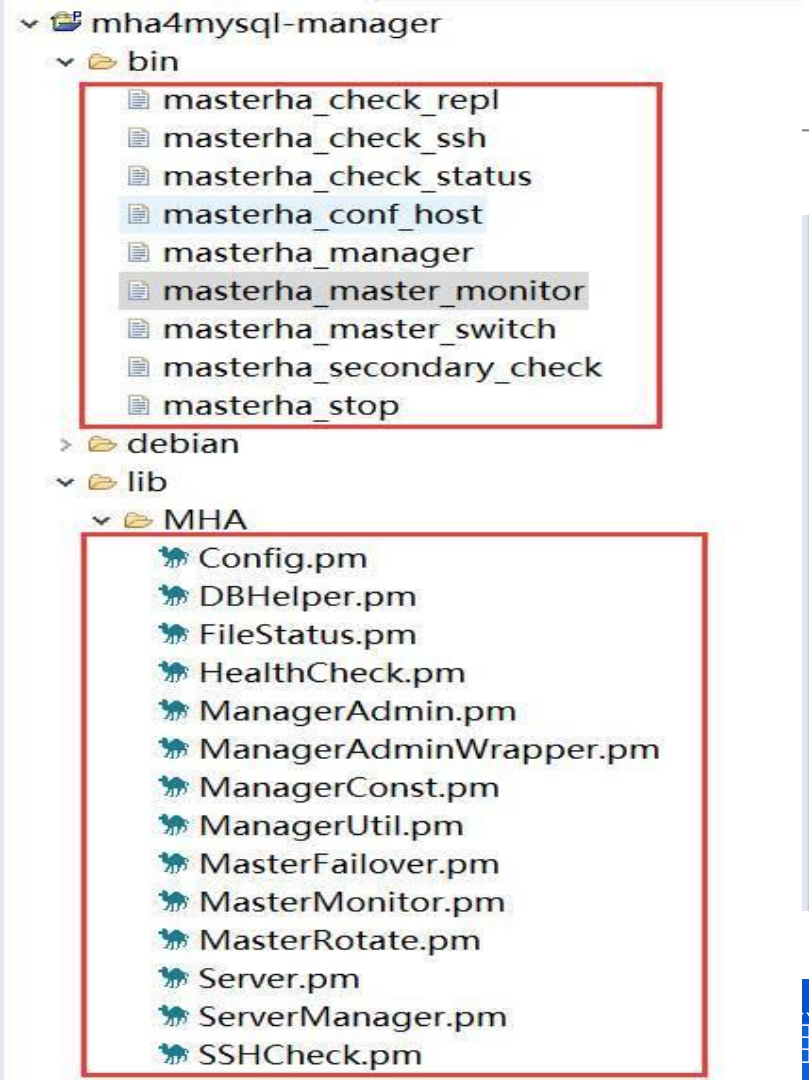


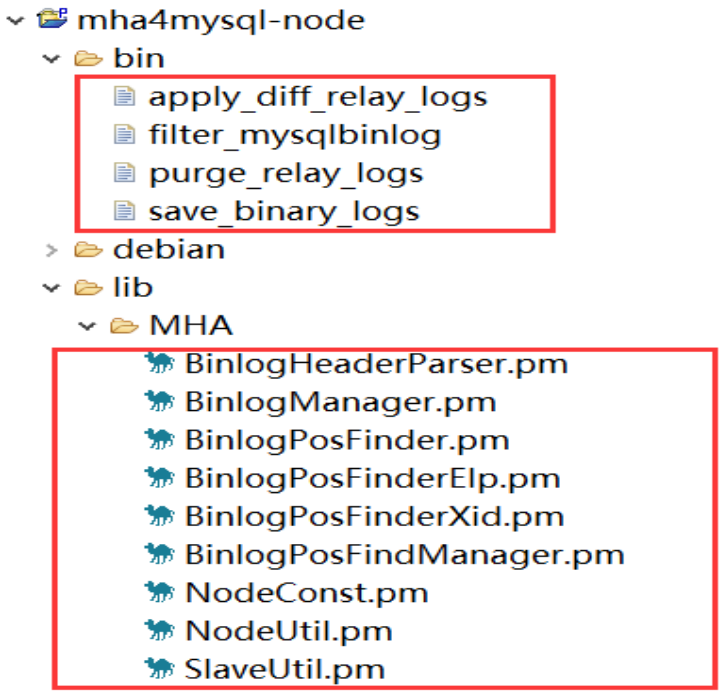


网络访问的差异

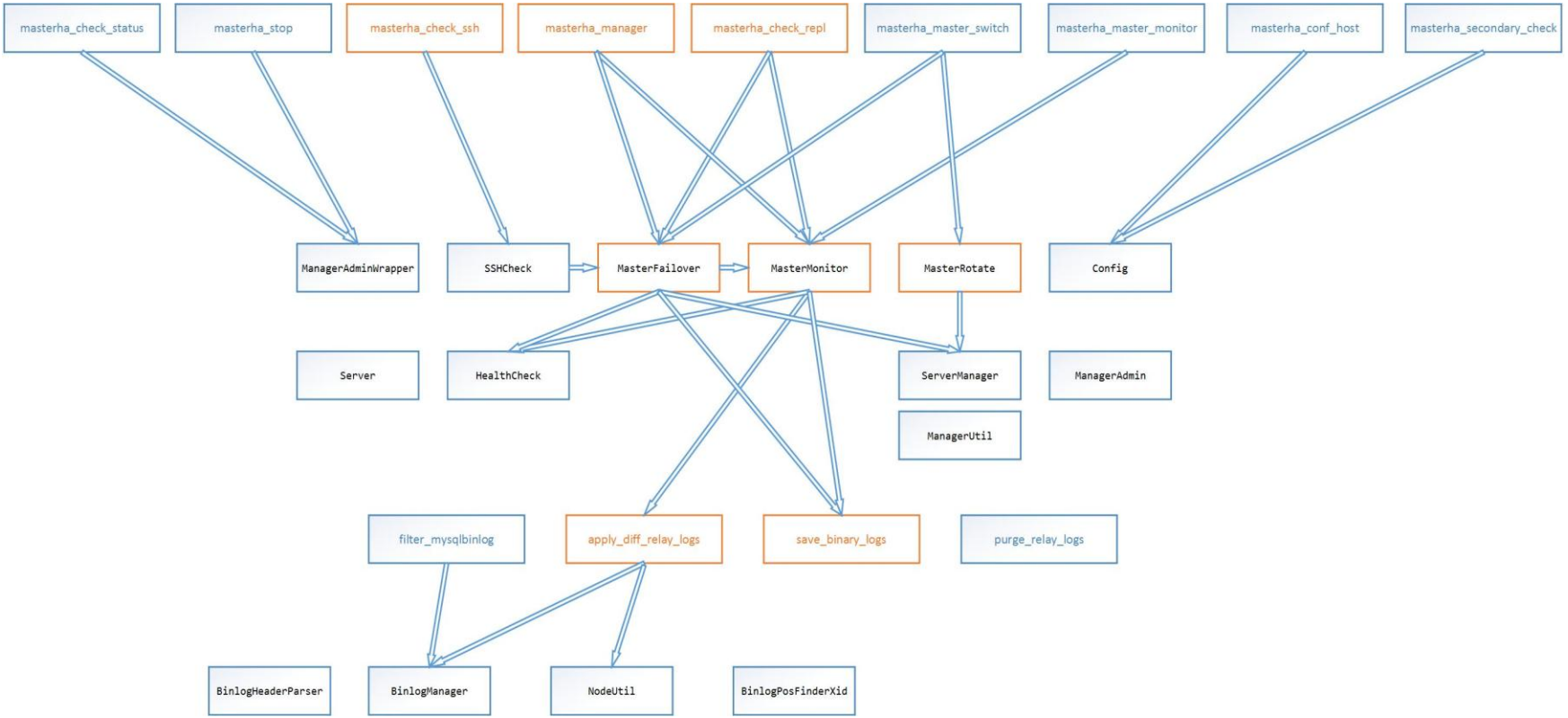

Application Server
SCAN
Listeners
Local
Listeners
Oracle RAC
Database
Clients


Application
Servers
Application
Servers
Application
Servers
MHA Manager1
MHA Manager2
master
VIP
slave1
slave2

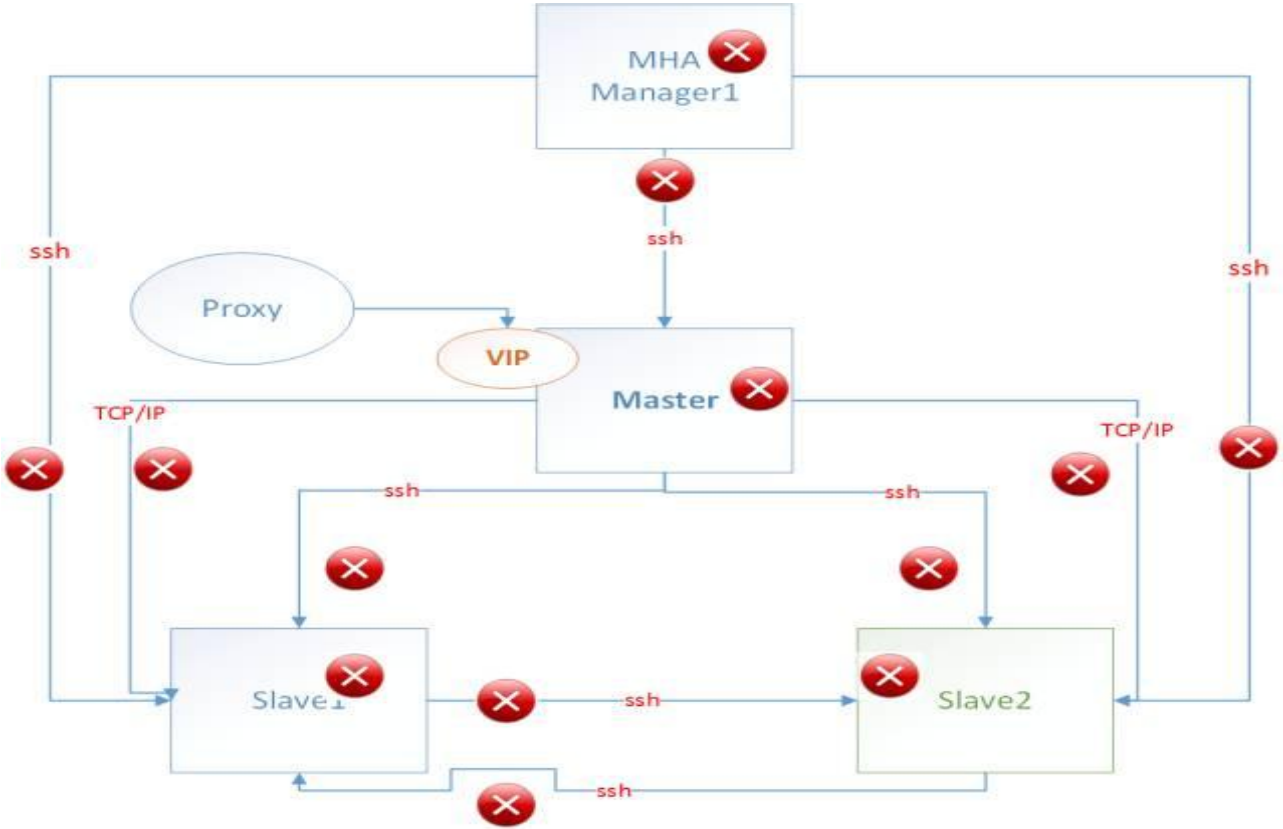

mha4mysql-manager
bin
masterha_check_repl
masterha_check_ssh
masterha_check_status
masterha_conf_host
masterha_manager
masterha_master_monitor
masterha_master_switch
masterha_secondary_check
masterha_stop
debian
lib
MHA
Config.pm
DBHelper.pm
FileStatus.pm
HealthCheck.pm
ManagerAdmin.pm
ManagerAdminWrapper.pm
ManagerConst.pm
ManagerUtil.pm
MasterFailover.pm
MasterMonitor.pm
MasterRotate.pm
Server.pm
ServerManager.pm
SSHCheck.pm


mha4mysql-node
bin
apply_diff_relay_logs
filter_mysqlbinlog
purge_relay_logs
save_binary_logs
debian
lib
MHA
BinlogHeaderParser.pm
BinlogManager.pm
BinlogPosFinder.pm
BinlogPosFinderElp.pm
BinlogPosFinderXid.pm
BinlogPosFindManager.pm
NodeConst.pm
NodeUtil.pm
SlaveUtil.pm

MHA的代码关系图


masterha_check_status
masterha_stop
masterha_check_ssh
masterha_manager
masterha_check_repl
masterha_master_switch
masterha_master_monitor
masterha_conf_host
masterha_secondary_check
ManagerAdminWrapper
SSHCheck
MasterFailover
MasterMonitor
MasterRotate
Config
Server
HealthCheck
ServerManager
ManagerAdmin
ManagerUtil
filter_mysqlbinlog
apply_diff_relay_logs
save_binary_logs
purge_relay_logs
BinlogHeaderParser
BinlogManager
NodeUtil
BinlogPosFinderXid

MHA的异常测试


MHA
Manager1
ssh
ssh
ssh
Proxy
VIP
Master
TCP/IP
TCP/IP
ssh
ssh
Slave1
ssh
Slave2
ssh

主从复制检查的阈值

my $g_seconds_behind_master = 30;

MasterMonit... | Config.pm | HealthCheck.pm | *Config2.pm | ManagerAdmin.pm

```
405         if ( !$_server_manager->is_gtid_auto_pos_enabled()
406           && check_binlog($current_master) )
407         {
408           $log->error("Master configuration failed.");
409           croak;
410         }
411       }
412       $_status_handler->set_master_host( $current_master->{hostname} )
413         unless ($g_check_only);
414
415       if ( !$_server_manager->is_gtid_auto_pos_enabled() && check_slave_env() ) {
416         $log->error("Slave configuration failed.");
417         croak;
418       }
419       $_server_manager->print_servers_ascii($current_master);
420       $_server_manager->check_replication_health($g_seconds_behind_master)
421         if ($g_check_repl_health);
422       check_scripts($current_master);
423       $_server_manager->disconnect_all();
424       $func_rc = 0;
425     };
426     if ($@) {
427       $log->error("Error happened on checking configurations. $@") if ($log);
428       undef $@;
429       return $func_rc;
430     }
431     return $func_rc if ($g_check_only);
432
```

Ping insert的逻辑

infra数据库，表chk_masterha

```
sub ping_insert($) {
  my $self = shift;
  my $log  = $self->{logger};
  my $dbh  = $self->{dbh};
  my ( $query, $sth, $href );
  eval {
    $dbh->{RaiseError} = 1;
    $dbh->do("CREATE DATABASE IF NOT EXISTS infra");
    $dbh->do(
"CREATE TABLE IF NOT EXISTS infra.chk_masterha (`key` tinyint NOT NULL primary key,`val` int(10) unsigned !
    );
    $dbh->do(
"INSERT INTO infra.chk_masterha values (1,unix_timestamp()) ON DUPLICATE KEY UPDATE val=unix_timestamp()"
    );
  };
```

```
[root@oel643 n2]# sdiff a.log  /tmp/test/slavediff_tmp_127.0.0.1_10002.log
/*!50530 SET @@SESSION.PSEUDO_SLAVE_MODE=1*/;                  /*!50530 SET @@SESSION.PSEUDO_SLAVE_MODE=1*/;
/*!50003 SET @OLD_COMPLETION_TYPE=@@COMPLETION_TYPE,COMPLETIO   /*!50003 SET @OLD_COMPLETION_TYPE=@@COMPLETION_TYPE,COMPLETIO
DELIMITER /*!*/;                                              DELIMITER /*!*/;
# at 4                                                      <
#171022  8:30:54 server id 10002  end_log_pos 123       Start <
# This format_description_event appears in a relay log and wa <
# at 123                                                    <
#171022  8:30:54 server id 10002  end_log_pos 150       Previ <
# [empty]                                                   <
# at 150                                                    <
#700101  8:00:00 server id 10001  end_log_pos 0         Rotat <
# at 190                                                    <
#171022  8:30:29 server id 10001  end_log_pos 123       Start <
BINLOG '                                                    <
pebrWQ8RJwAAdwAAAHsAAAAAAAQANS43LjE5LWwvZwAAAAAAAAAAAAAAAAAAA <
AAAAAAAAAAAAAAAAAAAAAAAAAAAEzgNAAgAEgAEBAQEEgAAXwAEGggAAAAICAgCA <
AKRCklw=                                                    <
'/*!*/;                                                     <
# at 309                                                    <
#171022  8:30:54 server id 0  end_log_pos 349   Rotate to bin <
# at 349                                                    <
#171022  8:30:54 server id 0  end_log_pos 389   Rotate to bin <
# at 389                                                    <
#171022  8:33:32 server id 10001  end_log_pos 974       GTID  <
SET @@SESSION.GTID_NEXT= '31a67400-b6c0-11e7-83e8-000c291264f <
# at 450                                                    <
#171022  8:33:32 server id 10001  end_log_pos 1150      Query <
SET TIMESTAMP=1508632412/*!*/;                              <
SET @@session.pseudo_thread_id=7/*!*/;                      <
SET @@session.foreign_key_checks=1, @@session.sql_auto_is_nul <
SET @@session.sql_mode=1436549152/*!*/;                     <
SET @@session.auto_increment_increment=1, @@session.auto_incr <
/*!\C utf8 *//*!*/;                                         <
SET @@session.character_set_client=33,@@session.collation_con <
SET @@session.lc_time_names=0/*!*/;                         <
SET @@session.collation_database=DEFAULT/*!*/;              <
CREATE USER 'test'@'%' IDENTIFIED WITH 'mysql_native_password <
/*!*/;                                                      <
# at 626                                                    <
#171022  8:33:42 server id 10001  end_log_pos 1211      GTID  <
SET @@SESSION.GTID_NEXT= '31a67400-b6c0-11e7-83e8-000c291264f <
# at 687                                                    <
#171022  8:33:42 server id 10001  end_log_pos 1335      Query <
SET TIMESTAMP=1508632422/*!*/;                              <
GRANT ALL PRIVILEGES ON *.* TO 'test'@'%'                   <
/*!*/;                                                      <
SET @@SESSION.GTID_NEXT= 'AUTOMATIC' /* added by mysqlbinlog   SET @@SESSION.GTID_NEXT= 'AUTOMATIC' /* added by mysqlbinlog
DELIMITER ;                                                 DELIMITER ;
# End of log file                                           # End of log file
/*!50003 SET COMPLETION_TYPE=@OLD_COMPLETION_TYPE*/;        /*!50003 SET COMPLETION_TYPE=@OLD_COMPLETION_TYPE*/;
/*!50530 SET @@SESSION.PSEUDO_SLAVE_MODE=0*/;               /*!50530 SET @@SESSION.PSEUDO_SLAVE_MODE=0*/;
[root@oel643 n2]#
```

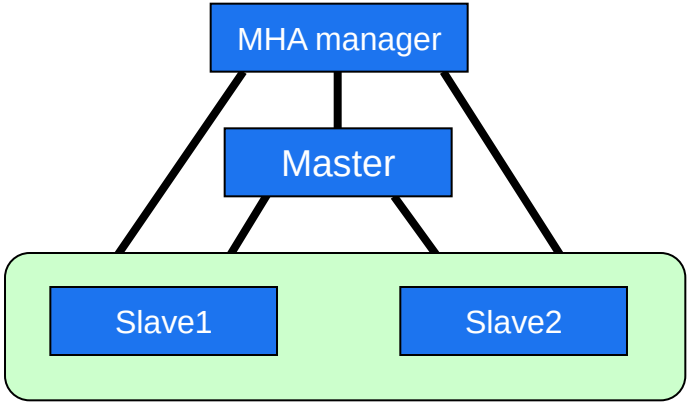
延迟的基准

延迟不是根据时间来算，而是根据relay的大小来算

```
# check slave is too behind master or not
# 0: no or acceptable delay
# 1: unacceptable delay (can not be a master)
sub check_slave_delay($$$) {
  my $self   = shift;
  my $target = shift;
  my $latest = shift;
  my $log    = $self->{logger};
  $log->debug(
    sprintf( "Checking replication delay on %s.. ", $target->get_hostinfo() ) );
  if (
    ( $latest->{Master_Log_File} gt $target->{Relay_Master_Log_File} )
    || ( $latest->{Read_Master_Log_Pos} >
      $target->{Exec_Master_Log_Pos} + 100000000 )
    )
  {
```

MHA的缺点

- 部署复杂度
  - 额外的perl安装包和模块
  - 节点信任关系
  - manager节点
  - node节点
- perl语言的“复杂度”
- 定制的复杂度
- secondary_check的逻辑
- 更新的周期，近些年来已经不维护了



MySQL执行计划剖析-初始化数据

```
CREATE TABLE `tmp_users` (
`id` int(11) NOT NULL
AUTO_INCREMENT,
`uid` int(11) NOT NULL,
`l_date` datetime NOT NULL,
`data` varchar(32) DEFAULT NULL,
PRIMARY KEY (`id`),
KEY `ind_uidldate` (`uid`,`l_date`)
) ENGINE=InnoDB DEFAULT CHARSET=gbk;
```

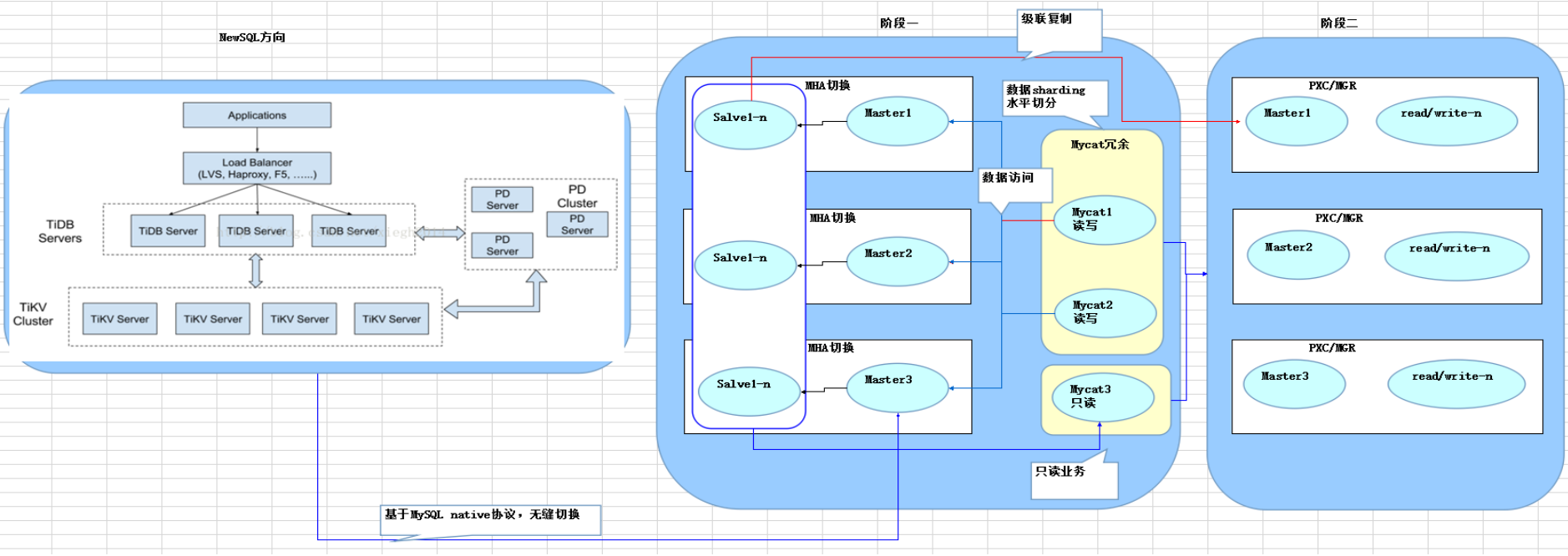
查看执行计划

```
explain select * from tmp_users where uid = 9527 and l_date >= '2012-12-10 10:13:17'\G
*************************** 1. row ***************************
           id: 1
  select_type: SIMPLE
        table: tmp_users
         type: range
  possible_keys: ind_uidldate
          key: ind_uidldate
      key_len: 9
          ref: NULL
         rows: 1
        Extra: Using index condition
  1 row in set (0.07 sec)
```

Data Typez	Storage Required Before MySQL 5.6.4	Storage Required as of MySQL 5.6.4
YEAR	1 byte	1 byte
DATE	3 bytes	3 bytes
TIME	3 bytes	3 bytes + fractional seconds storage
DATETIME	8 bytes	5 bytes + fractional seconds storage
TIMESTAMP	4 bytes	4 bytes + fractional seconds storage

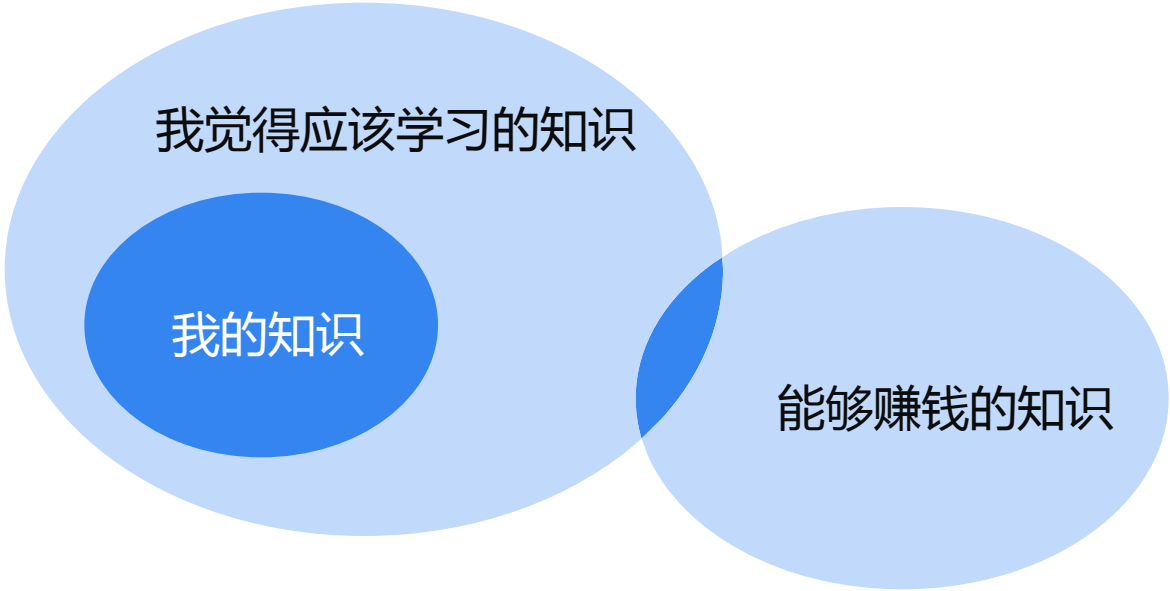
DBA进阶之路

DBA进阶之路-HTAP方案

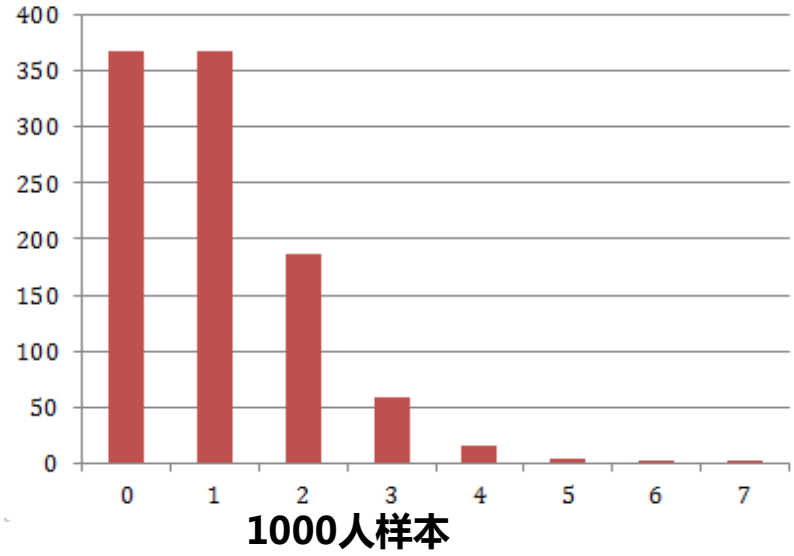


DBA进阶之路

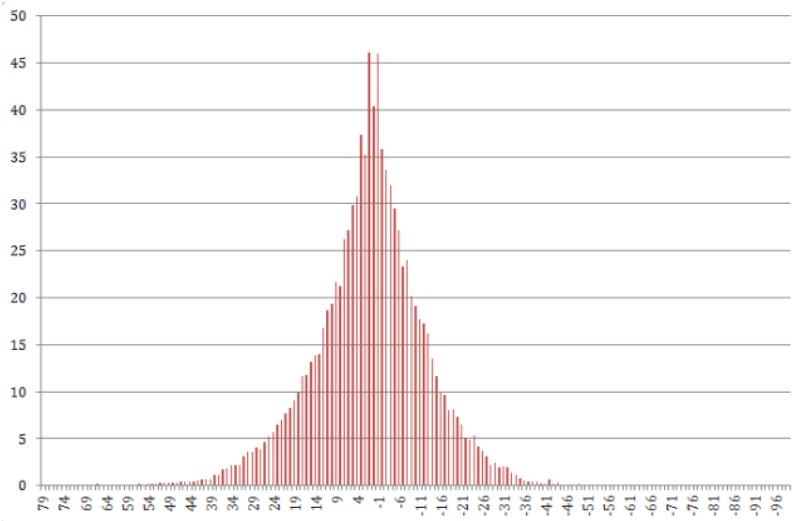
矛盾和困扰并存，用心掌握



通过SQL来解读财富分配



1000人样本



允许透支，样本100

欢迎志同道合的你

JJworld 竞技世界

简历投递
zhangkl@mail.jj.cn

DBA

人工智能　机器学习

大数据（开发、运维）

全球敏捷运维峰会

THANK YOU!