**一、MongoDB介绍**

MongoDB 是一个基于分布式文件存储的数据库（开源）。由 C++ 语言编写。旨在为 WEB 应用提供可扩展的高性能数据存储解决方案。

MongoDB 是一个介于关系数据库和非关系数据库之间的产品，是非关系数据库当中功能最丰富，最像关系数据库的。

**二、监控工具**

1、MongoDB提供了网络和系统监控工具Munin，它作为一个插件应用于MongoDB中。

2.Gangila是MongoDB高性能的系统监视的工具，它作为一个插件应用于MongoDB中。

3.基于图形界面的开源工具 Cacti, 用于查看CPU负载, 网络带宽利用率,它也提供了一个应用于监控 MongoDB 的插件。

**三、MongoDB安装**

1.在Windows下安装

2.在Linux下安装

①、导入公共键

sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv EA312927

②、给mongodb创建一个列表文件（以Ubuntu 14.04为例）

echo "deb http://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.2 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.2.list

③、重载本地包数据库

sudo apt-get update

④、安装mongodb

sudo apt-get install -y mongodb

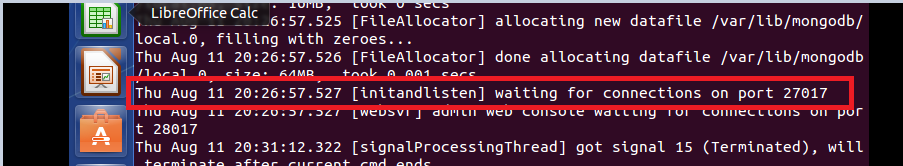
⑤、启动mongodb

sudo service mongod start

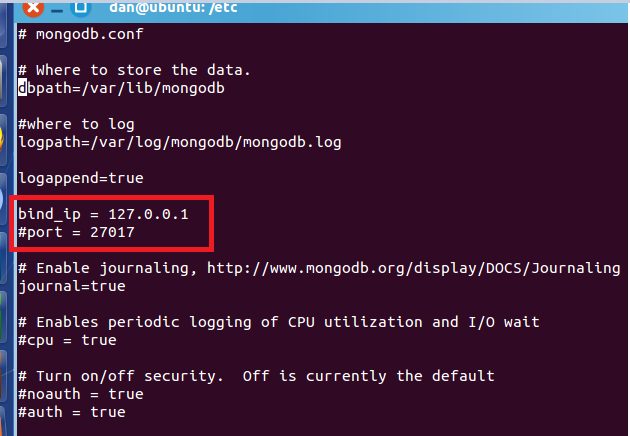
sudo ./mongod --config /etc/mongodb.conf  #启动MongoDB

⑥、如何验证mongodb是否启动成功

进入/var/log/mongodb/mongod.log，查看内容[initandlisten] waiting for connections on port <port>，默认端口为27017.如下图：



注：安装完成默认地址为127.0.0.1，端口为27017，可以通过/etc/mongodb.conf文件修改



mongo+IP号进入数据库

**四、MongoDB与MySQL，Oracle的区别以及优缺点**

**1、区别**

Oracle和MySQL是传统的关系型数据库，MongoDB则是非关系型数据库，也叫文档型数据库。

下表将帮助您更容易理解Mongo中的一些概念：

|  |  |  |
| --- | --- | --- |
| **SQL术语/概念** | **MongoDB术语/概念** | **解释/说明** |
| database | database | 数据库 |
| table | collection | 数据库表/集合 |
| row | document | 数据记录行/文档 |
| column | field | 数据字段/域 |
| index | index | 索引 |
| table joins |  | 表连接,MongoDB不支持 |
| primary key | primary key | 主键,MongoDB自动将\_id字段设置为主键 |

**2.关系型数据库优点和瓶颈**

优点：

①容易理解：二维表结构是非常贴近逻辑世界的一个概念，关系模型相对网状、层次等其他模型来说更容易理解

②使用方便：通用的SQL语言使得操作关系型数据库非常方便

③易于维护：丰富的完整性(实体完整性、参照完整性和用户定义的完整性)大大减低了数据冗余和数据不一致的概率

④事务一致性，数据的持久存储

瓶颈：

①高并发读写需求：网站的用户并发性非常高，往往达到每秒上万次读写请求，对于传统关系型数据库来说，硬盘I/O是一个很大的瓶颈

②海量数据的高效率读写:网站每天产生的数据量是巨大的，对于关系型数据库来说，在一张包含海量数据的表中查询，效率是非常低的

③高扩展性和可用性:在基于web的结构当中，数据库是最难进行横向扩展的，当一个应用系统的用户量和访问量与日俱增的时候，数据库却没有办法像web server和app server那样简单的通过添加更多的硬件和服务节点来扩展性能和负载能力。对于很多需要提供24小时不间断服务的网站来说，对数据库系统进行升级和扩展是非常痛苦的事情，往往需要停机维护和数据迁移。

**3.非关系型数据库优点和瓶颈**

优点：

1）成本：nosql数据库简单易部署，基本都是开源软件，不需要像使用oracle那样花费大量成本购买使用，相比关系型数据库价格便宜。  
2）查询速度：nosql数据库将数据存储于缓存之中，关系型数据库将数据存储在硬盘中，自然查询速度远不及nosql数据库。  
3）存储数据的格式：nosql的存储格式是key,value形式、文档形式、图片形式等等，所以可以存储基础类型以及对象或者是集合等各种格式，而数据库则只支持基础类型。

4)可扩展性高,尤其是水平横向扩展

（纵向扩展是指用更强的机器；横向扩展是指把数据分散到多个机器）

瓶颈：

1）维护的工具和资料有限，因为nosql是属于新的技术，不能和关系型数据库10几年的技术同日而语。  
2）不提供对sql的支持，如果不支持sql这样的工业标准，将产生一定用户的学习和使用成本。  
3）不提供关系型数据库对事物的处理。

 mongodb以BSON结构（二进制）进行存储，对海量数据存储有着很明显的优势。下面是Mongodb与Mysql的操作命令的对比。

|  |  |  |
| --- | --- | --- |
| 作用 | MySQL | MongoDB |
|  |  |  |
| 服务器守护进程 | mysqld | mongod |
| 客户端工具 | mysql | mongo |
| 逻辑备份工具 | mysqldump | mongodump |
| 逻辑还原工具 | mysql | mongorestore |
| 数据导出工具 | mysqldump | mongoexport |
| 数据导入工具 | source | mongoimport |
| 新建用户并授权 | grant all on \*.\* to username@'localhost'  identified by 'passwd'; | db.addUser("user","psw") db.auth("user","psw") |
| 显示库列表 | show databases; | show dbs |
| 进去库 | use dbname; | use dbname |
| 显示表列表 | show tables; | show collections |
| 查询主从状态 | show slave status; | rs.status |
| 创建库 | create database name; | 无需单独创建，直接use进去 |
| 创建表 | create table tname(id int); | 无需单独创建，直接插入数据 |
| 删除表 | drop table tname; | db.tname.drop() |
| 删除库 | drop database dbname; | 首先进去该库，db.dropDatabase() |
| 插入记录 | insert into tname(id) value(2); | db.tname.insert({id:2}) |
| 删除记录 | delete from tname where id=2; | db.tname.remove({id:2}) |
| 修改/更新记录 | update tname set id=3 where id=2; | db.tname.update({id:2}, {$set:{id:3}},false,true) |
| 查询所有记录 | select \* from tname; | db.tname.find() |
| 查询所有列 | select id from tname; | db.tname.find({},{id:1}) |
| 条件查询 | select \* from tname where id=2; | db.tname.find({id:2}) |
| 条件查询 | select \* from tname where id < 2; | db.tname.find({id:{$lt:2}}) |
| 条件查询 | select \* from tname where id >=2; | db.tname.find({id:{$gte:2}}) |
| 条件查询 | select \* from tname where id=2 and name='steve'; | db.tname.find({id:2, name:'steve'}) |
| 条件查询 | select \* from tname where id=2 or name='steve'; | db.tname.find($or:[{id:2}, {name:'steve'}]) |
| 条件查询 | select \* from tname limit 1; | db.tname.findOne() |
| 模糊查询 | select \* from tname where name like "%ste%"; | db.tname.find({name:/ste/}) |
| 模糊查询 | select \* from tname where name like "ste%"; | db.tname.find({name:/^ste/}) |
| 获取表记录数 | select count(id) from tname; | db.tname.count() |
| 获取有条件 的记录数 | select count(id) from tname where id=2; | db.tname.find({id:2}).count() |
| 查询时去掉 重复值 | select distinct(last\_name) from tname; | db.tname.distinct('last\_name') |
| 正排序查询 | select \*from tname order by id; | db.tname.find().sort({id:1}) |
| 逆排序查询 | select \*from tname order by id desc; | db.tname.find().sort({id:-1}) |
| 取存储路径 | explain select \* from tname where id=3; | db.tname.find({id=3}).explain() |

特别要注意的是：mongodb插入多个字段语法

> db.user.insert({id:1,name:'steve',sex:'male'})  正确

> db.user.insert({id:2},{name:'bear'},{sex:'female'})  错误

**参考资料：**

https://docs.mongodb.com/manual/tutorial/install-mongodb-on-ubuntu/#packages

http://www.runoob.com/mongodb/mongodb-linux-install.html

http://blog.csdn.net/robinjwong/article/details/18502195/

http://www.osyunwei.com/archives/7629.html