Tedu.cn
达内教育

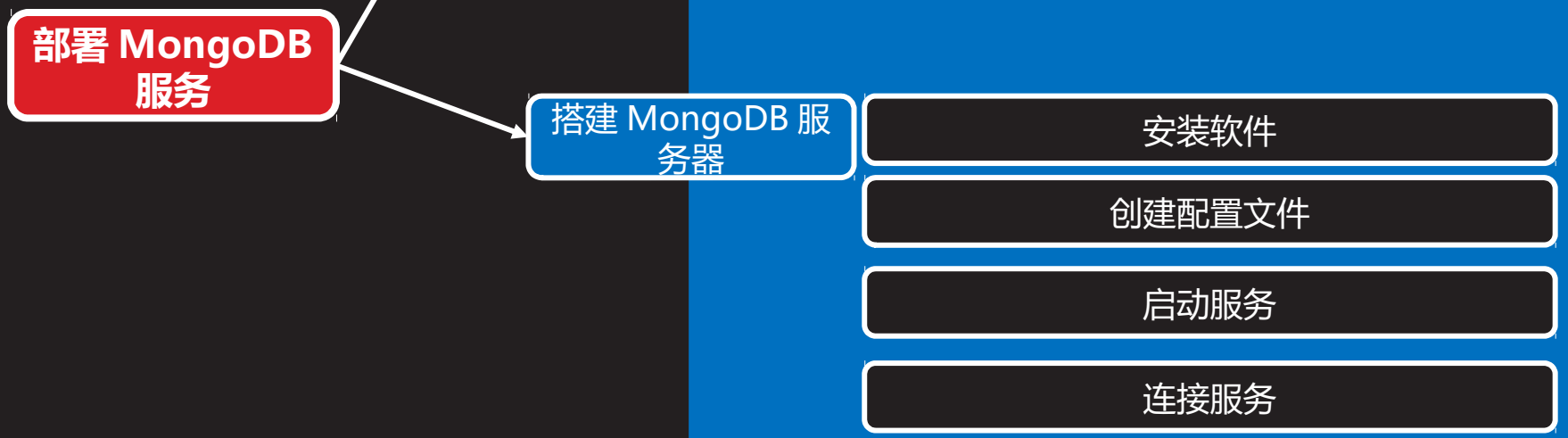
数据库管理

NSD NoSQL DAY04

内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	部署 MongoDB 服务
	10:30 ~ 11:20	
	11:30 ~ 12:00	
下午	14:00 ~ 14:50	MongoDB 基本使用
	15:00 ~ 15:50	
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑

部署 MongoDB 服务



MongoDB 概述

软件介绍

- 介于关系数据库和非关系数据库之间的产品
  - 一个基于分布式文件存储的数据库。
  - 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 将数据存储为一个文档，数据结构由键值(key=>value) 对组成。
  - MongoDB 文档类似于 JSON 对象。字段值可以包含其他文档，数组及文档数组。

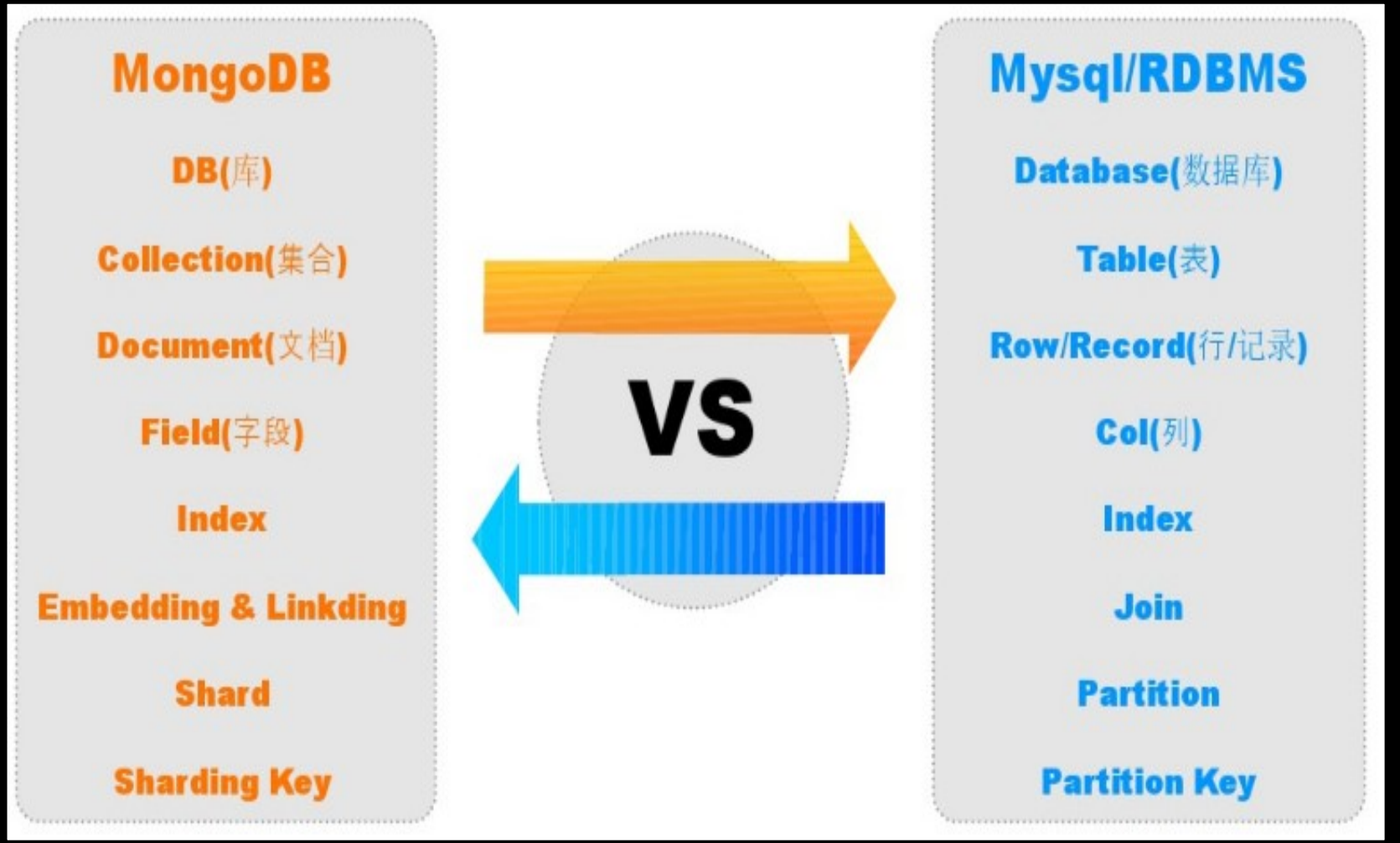
知识讲解

软件特点

- 安装简单
- 面向文档存储，操作比较简单容易
- 支持丰富的查询表达
- 可以设置任何属性的索引
- 支持主流编程语言 RUBY|PYTHON|JAVA|PHP|C++
- 支持副本集，分片

知识讲解

相关概念


MongoDB
DB(库)
Collection(集合)
Document(文档)
Field(字段)
Index
Embedding & Linkding
Shard
Sharding Key
VS
Mysql/RDBMS
Database(数据库)
Table(表)
Row/Record(行/记录)
Col(列)
Index
Join
Partition
Partition Key

知识讲解

搭建 MongoDB 服务器

装包

- 免安装，解压后即可使用

```
 [root@bogon ~]# mkdir /usr/local/mongodb
[root@bogon ~]# tar -zxf mongodb-linux-x86_64-rhel70-
3.6.3.tgz
[root@bogon ~]#
[root@bogon ~]# cp -r
mongodb-linux-x86_64-rhel70-3.6.3/bin  /usr/local/mongodb/

[root@bogon ~]# cd /usr/local/mongodb/
[root@bogon mongodb]# mkdir etc
[root@bogon mongodb]# mkdir log
[root@bogon mongodb]# mkdir -p  data/db
```

知识讲解

创建配置文件

- 手动创建服务主配置文件

知识讲解

```
[root@bogon ~]# vim mongodb.conf
logpath=/usr/local/mongodb/log/mongodb.log
logappend=true                  # 追加的方式记录日志信息
dbpath=/usr/local/mongodb/data/db  # 数据库目录
fork=true                       # 守护进程方式运行
```

启动服务

- 启动服务
  -]# ./bin/mongod -f /usr/local/mongodb/etc/mongodb.conf
- 查看进程
  -]# ps -C mongod
- 查看端口
  -]# netstat -utnlp | grep :27017

知识讲解

连接服务

- 本地连接，默认没有密码

```
[root@bogon ~]# /usr/local/mongodb/bin/mongo
MongoDB shell version v3.6.3
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.6.3
……
> show dbs   // 显示已有的库
admin   0.000GB
config  0.000GB
local   0.000GB
> exit  # 断开连接
bye
[root@bogon ~]#
```

知识讲解

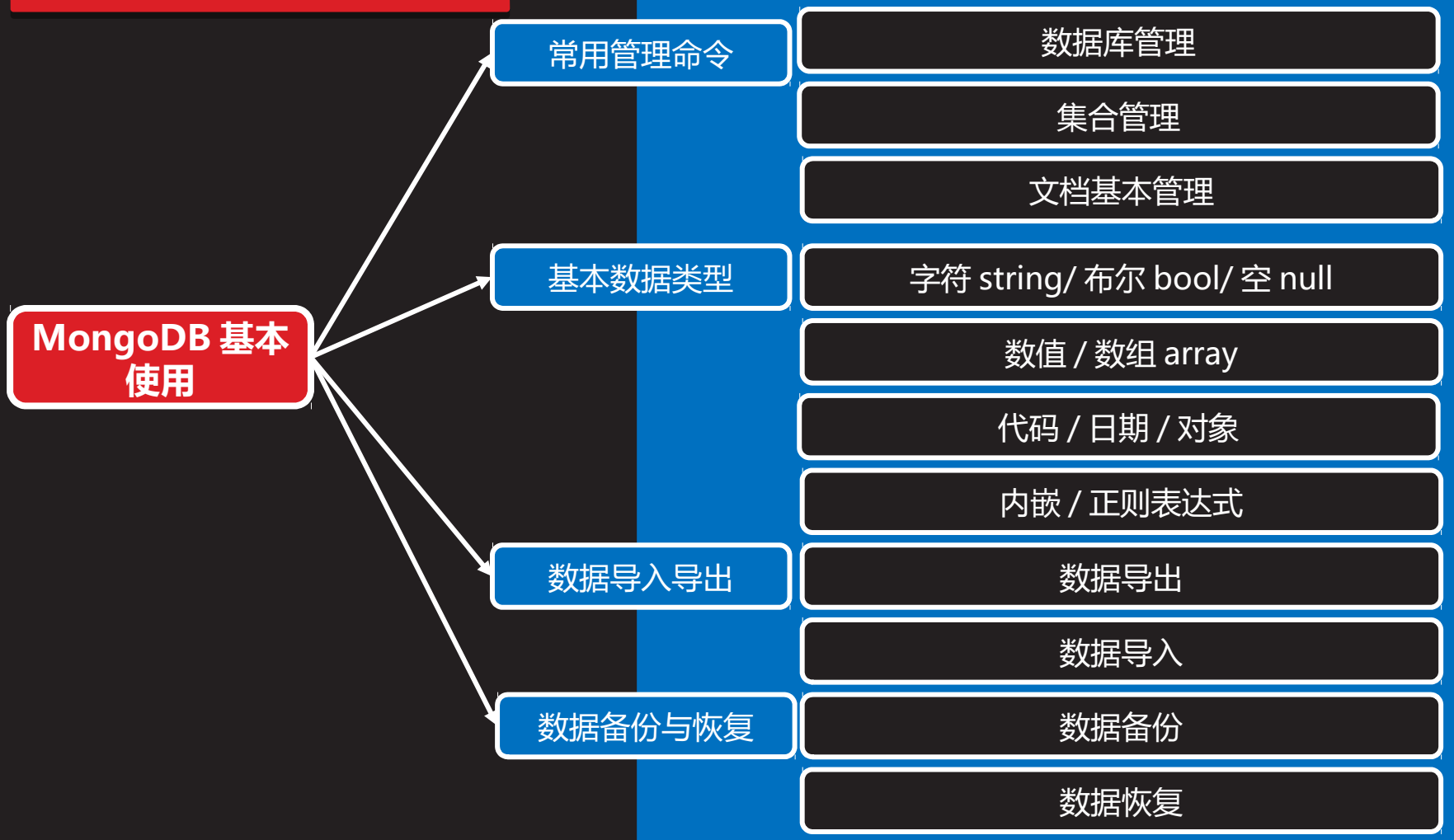
案例 1：搭建 MongoDB 服务器

满足以下要求：

- 在主机 192.168.4.51 上部署 MongoDB 服务

课堂练习

MongoDB 基本使用



Tedu.cn
达内教育

常用管理命令

数据库管理

知识讲解

- 库管理命令： 查看 创建 切换 删除
  - show dbs 查看已有的库
  - db 显示当前所在的库
  - use 库名 切换库，若库不存在的话 自动延时创建库
  - show collections 或 show tables 查看库下已有的集合
  - db.dropDatabase() 删除当前所在的库

数据库管理（续 1 ）

知识讲解

- 数据库名可以是满足以下条件的任意 UTF-8 字符串。
  - 不能是空字符串（""）。
  - 不得含有' '（空格）、 .、 $、 /、 \和\0（空字符）。
  - 应全部小写。
  - 最多 64 字节。

集合管理

- 集合管理命令：查看 创建 删除
  - show collections 或 show tables # 查看集合
  - db.集合名.drop() # 删除集合
  - db.集合名.save({'','}) # 创建集合，集合不存在时，创建并添加文档

```
> db.user.save({'name':'bob','age':'21'})
WriteResult({ "nInserted" : 1 })
```

知识讲解

集合管理（续 1 ）

- 合法的集合名
  - 集合名不能是空字符串 "" 。
  - 集合名不能含有 \0 字符（空字符），这个字符表示集合名的结尾。
  - 集合名不能以 "system." 开头，这是为系统集合保留的前缀。
  - 用户创建的集合名字不能含有保留字符。

知识讲解

文档基本管理

- 文档 ： 类似于 MySQL 表里的记录

知识讲解

id	user_name	email	age	city
1	Mark Hanks	mark@abc.com	25	Los Angeles
2	Richard Peter	richard@abc.com	31	Dallas

```
{
    "_id": ObjectId("5146bb52d8524270060001f3"),
    "age": 25,
    "city": "Los Angeles",
    "email": "mark@abc.com",
    "user_name": "Mark Hanks"
}
{
    "_id": ObjectId("5146bb52d8524270060001f2"),
    "age": 31,
    "city": "Dallas",
    "email": "richard@abc.com",
    "user_name": "Richard Peter"
}
```

文档基本管理 (续 1)

- 文档管理命令： 查看 统计 添加 删除
  - db. 集合名 .find()
  - db. 集合名 .count()
  - db. 集合名 .insert({ “name” :” jim” })
  - db. 集合名 .find({ 条件 })
  - db. 集合名 .findOne() # 返回一条文档
  - db. 集合名 .remove({}) # 删除所有文档
  - db. 集合名 .remove({ 条件 }) # 删除与条件匹配的所有文档

知识讲解

文档管理 (续 2)

- 插入记录

```
>db.col.insert(
{ title: 'MongoDB 教程 ',
  description: 'MongoDB 是一个 Nosql 数据库 ',
  by: 'MongoDB 中文网 ',
  url: 'http://www.mongodb.org.cn',
  tags: ['mongodb', 'database', 'NoSQL'],
  likes: 100
}
)
```

>db.col.remove({‘title’:‘MongoDB 教程’ }) 删除记录

知识讲解

案例 2：常用管理命令练习

要求如下：

- 练习库的创建、查看、切换、删除
- 练习集合的创建、查看、删除
- 练习文档的查看、插入、删除

课堂练习

Tedu.cn
达内教育

基本数据类型

字符 string/ 布尔 bool/ 空 null

知识讲解

- 字符串 string
  - UTF-8 字符串都可以表示为字符串类型的数据
  - {name:" 张三" } 或 { school:"tarena" }
- 布尔 bool
  - 布尔类型有两个值 true 和 false， {x:true}
- 空 null
  - 用于表示空值或者不存在的字段， {x:null}

数值 / 数组 array

- 数值
  - shell 默认使用 64 为浮点型数值。 {x： 3.14} 或 {x： 3} 。
  - NumberInt （4 字节整数） {x:NumberInt(3)}
  - NumberLong （8 字节整数） {x:NumberLong(3)}

- 数组 array
  - 数据列表或数据集可以表示为数组
  - {x： [“a “， “b”，” c”]}

知识讲解

代码 / 日期 / 对象

- 代码
  - 查询和文档中可以包括任何 JavaScript 代码
  - {x: function( ){/*　代码　*/}}
- 日期
  - 日期被存储为自新纪元依赖经过的毫秒数，不存储时区
  - {x:new Date( )}
- 对象
  - 对象 id 是一个 12 字节的字符串，是文档的唯一标识
  - {x: ObjectId() }

知识讲解

内嵌 / 正则表达式

- 内嵌
  - 文档可以嵌套其他文档，被嵌套的文档作为值来处理
  - {tarena: {address: “Beijing”,tel: “888888”,person:” hanshaoyun”
  - }}
- 正则表达式
  - 查询时，使用正则表达式作为限定条件
  - {x:/ 正则表达式 /}

知识讲解

Tedu.cn
达内教育

数据导入导出

数据导出

- 语法格式 1
  - #mongoexport [--host IP 地址 --port 端口]
    -d 库名 -c 集合名 -f 字段名 1, 字段名 2
    --type=csv > 目录名 / 文件名 .csv
- 语法格式 2
  - #mongoexport --host IP 地址 --port 端口
    - 库名 -c 集合名 -q '{ 条件 }' -f 字段名 1 , 字段名 2
    --type=csv > 目录名 / 文件名 .csv

注意：导出为 csv 格式必须使用 -f 指定字段名列表 !!!

知识讲解

数据导出（续 1 ）

- 语法格式 3

#mongoexport [--host IP 地址 --port 端口]
-d 库名 -c 集合名 [-q '{ 条件 }' –f 字段列表]
--type=json > 目录名 / 文件名 .json

```
[root@host50 bin]#
[root@host50 bin]# ./mongoexport -d mdb -c t1 --type=json > /root/t1.json
2018-05-27T14:22:48.005+0800    connected to: localhost
2018-05-27T14:22:48.006+0800    exported 1 record
[root@host50 bin]#
```

```
[root@host50 bin]#
[root@host50 bin]# ./mongoexport -d mdb -c t1 -q '{uid:{$lt:5}}' \
> -f name,shell --type=json > /root/user2.json
2018-05-27T15:03:52.962+0800    connected to: localhost
2018-05-27T15:03:52.962+0800    exported 7 records
[root@host50 bin]#
```

知识讲解

数据导入

- 语法格式 1
  - #mongoimport –host IP 地址 – port 端口
    -d 库名 – c 集合名
    --type=json 目录名 / 文件名 .json
- 语法格式 2
  - #mongoimport –host IP 地址 – port 端口
    -d 库名 – c 集合名
    --type=csv --headerline [--drop] 目录名 / 文件名 .csv

注意：导入数据时库和集合不存在时，会创建库和集合后导入数据
反之以追加的方式导入数据到集合里，使用— drop 选项可以删除原有数据后导入新数据 --headerline 忽略标题

Tedu.cn
达内教育

数据备份恢复

数据备份

- 备份数据所有库到当前目录下的 dump 目录下

mongodump [--host ip 地址 --port 端口]

- 备份时指定备份的库和备份目录

mongodump [--host ip 地址 --port 端口] -d 数据库名 -c 集合名 -o 目录

目录无需事先创建 备份时指定即可！！！

- 查看 bson 文件内容

#bsondump ./dump/bbs/t1.bson

知识讲解

数据恢复

- 语法格式
  - mongorestore --host IP 地址 --port 端口 -d 数据库名 [-c 集合名] 备份目录名

知识讲解

```
[root@host50 ~]#
[root@host50 ~]# /usr/local/mongodb/bin/mongodump -d bbs3 -c user3 -o /bbs3bak
2018-05-27T15:51:58.990+0800    writing bbs3.user3 to
2018-05-27T15:51:58.990+0800    done dumping bbs3.user3 (43 documents)
[root@host50 ~]#
```

```
[root@host50 ~]# ls /bbs3bak/
bbs3
[root@host50 ~]# ls /bbs3bak/bbs3/
user3.bson  user3.metadata.json
[root@host50 ~]#
[root@host50 ~]# /usr/local/mongodb/bin/mongorestore -d bbs3 /bbs3bak/bbs3/
```

```
[root@host50 ~]#
[root@host50 ~]# /usr/local/mongodb/bin/mongorestore -d bbs3 \
> -c user3 /bbs3bak/bbs3/user3.bson
```

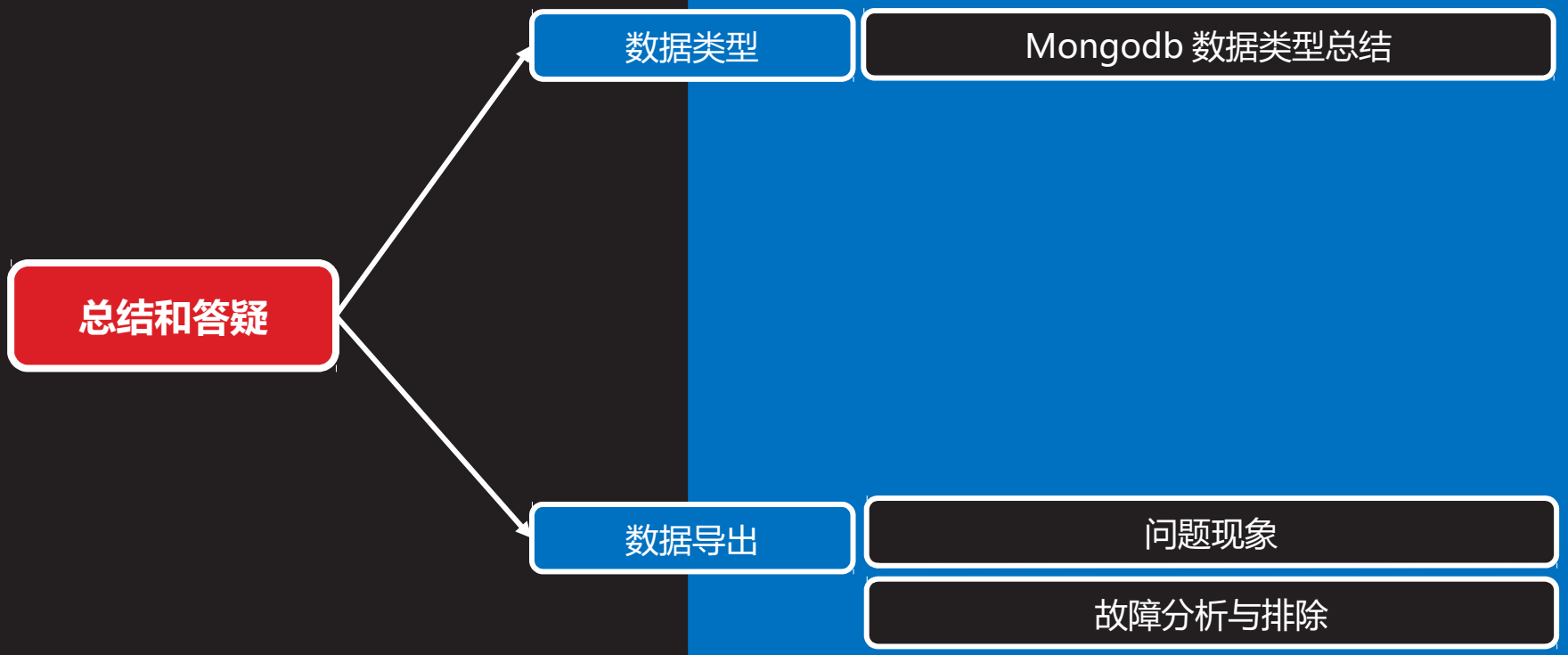
案例 3： 数据导入导出 / 备份 / 恢复

要求如下：

- 练习数据导入导出
- 练习数据备份恢复

课堂练习

总结和答疑



Tedu.cn
达内教育

数据类型

数据类型总结

- mongodb 数据类型总结
  - 字符串类型
  - 数值类型
  - 布尔类型
  - 空 / 正则 / 代码
  - 数组
  - 数值
  - 日期
  - 对象
  - 内嵌

知识讲解

Tedu.cn
达内教育

数据导出

问题现象

- 数据导出为 csv 格式时报错

```
[root@host50 bin]# ./mongoexport  -d mdb -c t1  --t
ype=csv  > /root/t1.csv
2018-05-27T14:36:04.084+0800  Failed: CSV mode r
equires a field list
```

知识讲解

故障分析及排除

- csv 格式导出数据时，必须指定导出的字段名

[root@host50 bin]# ./mongoexport -d mdb -c t1
-f name,shell,uid --type=csv > /root/t1.csv

```
[root@host50 bin]#
[root@host50 bin]# ./mongoexport -d mdb -c t1 -f name,uid,shell --type=csv > /root/t1.csv
2018-05-27T14:41:30.225+0800    connected to: localhost
2018-05-27T14:41:30.226+0800    exported 1 record
[root@host50 bin]#
```

知识讲解