

分布式数据库

银行业务管理系统

刘向昆

郑艺

王睿

赵堃宇

马昊聪

01 需求分析

02 功能设计

03 数据库设计

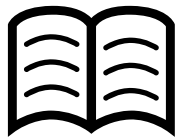
04 实现

目录 CONTENTS

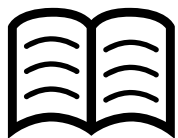
A stylized, layered mountain range graphic in shades of gray and black, spanning the top half of the image. The mountains are composed of several overlapping, wavy horizontal bands of varying heights and shades, creating a sense of depth and texture. The topmost layer is a light gray, while the bottom layer is a dark gray, with intermediate layers in various shades of gray.

01

需求分析



银行在企业的生产、人们的生活和国家的货币政策及管理上都是不可或缺的一个重要机构。由于银行储户数量众多、交易量巨大，银行每天要面临数以亿计的存取款交易记录，这使得无论是存储上还是在交易的实时上对数据库的存取要求都很高。因此在银行管理系统的设计上，多采用分布式数据库的形式进行管理，在每个省或市建立分站点，银行总部作为总站点，人们的主要业务交易都在分站点上进行，而总站点主要负责交易汇总和账户管理等事务处理，这样就大大减少了总站点的数据库访问量



银行管理系统主要受理用户的开户、存取款、转账等业务，利用计算机网络将银行的总站点和分站点连接起来。分站点主要负责用户的开户、存取款、转账等业务功能，总站点主要负责账户信息的存取以及交易记录的汇总、存取等功能，具有在地域上分散而管理上又相对集中的特点。

A stylized, layered mountain range graphic in shades of gray and black, spanning the top half of the image. The mountains are composed of several overlapping, wavy horizontal bands of varying heights and shades, creating a sense of depth and atmospheric perspective. The topmost layer is a very light gray, while the bottom layer is a dark charcoal gray.

02

功能设计

NO.	TITLE HERE
1	取款、取款、转账
2	查询交易记录
3	职员管理
4	用户管理
5	账户管理

01 系统需求

总行

查询维护所有账户信息以及交易记录。

分行

保存在本行开户的用户信息，以及在本行进行交易的交易记录，分行将用户和交易记录的汇总信息汇总到总行进行统一管理。

分行

任何开户的用户在全国范围内都可以进行存取款、转账等交易，即要求实现异地交易。

分行

分行维护自身的职员信息且只能查找各自分行的职员信息。

A stylized, layered mountain range background. The mountains are represented by overlapping, rounded shapes in various shades of gray, from light to dark. The foreground is a solid dark gray, and the sky is white.

03

数据库设计

分布设计

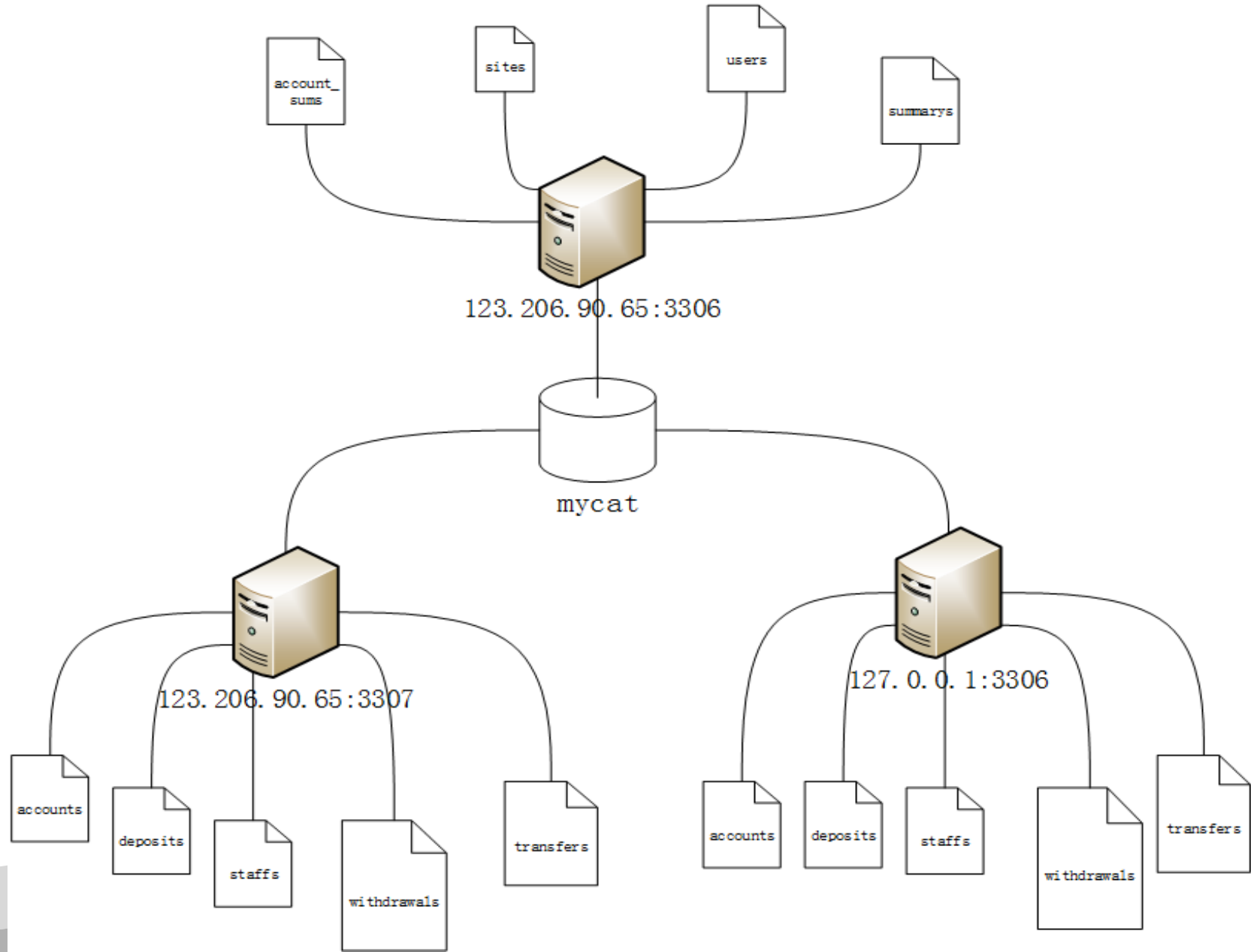
非冗余分配

职员信息：各分行自行管理
用户信息：用户信息放到总行管理
站点信息：站点信息放到总行管理

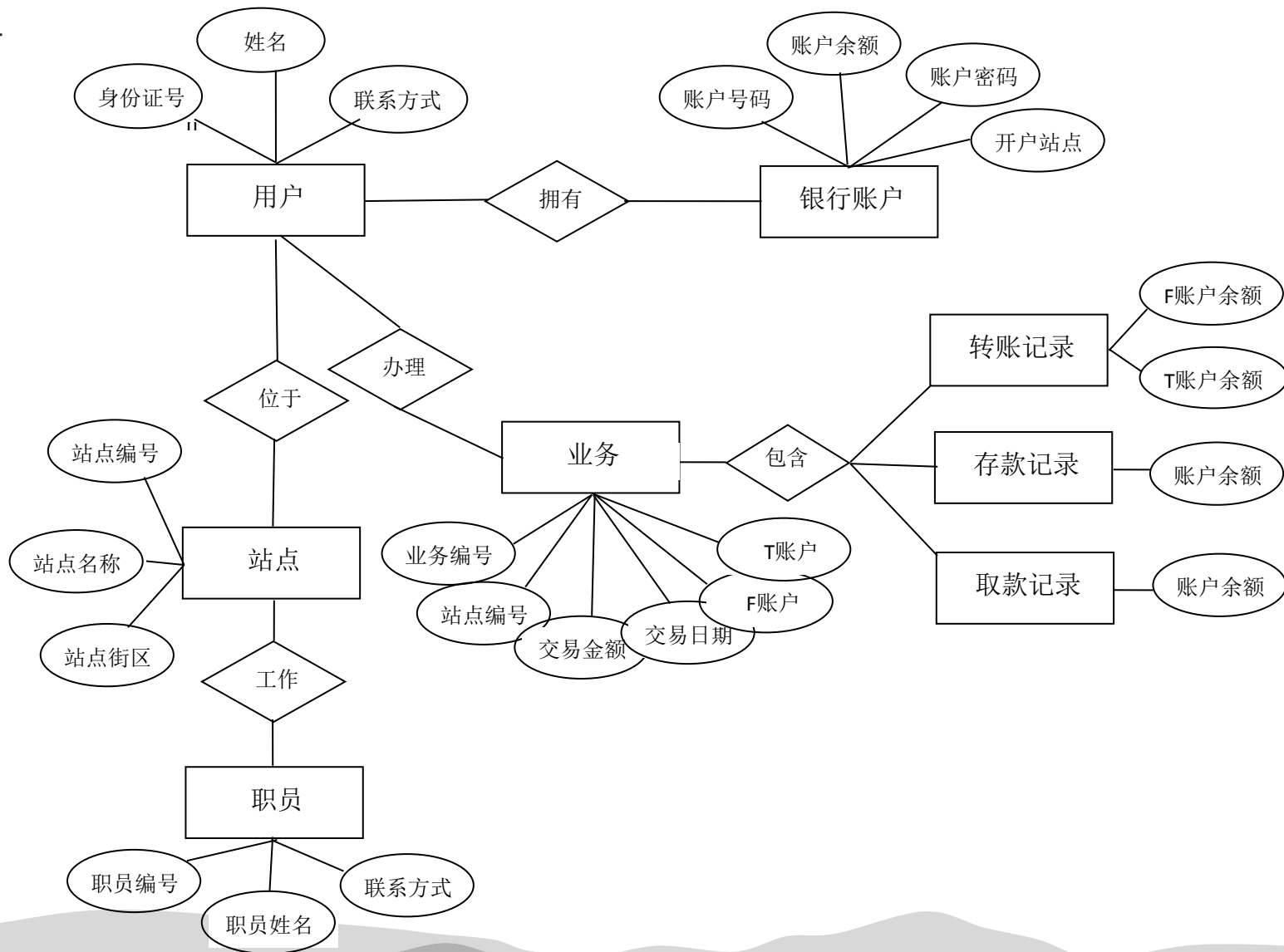
冗余分配

交易信息：各分行存储自己站点的交易信息，同时
汇总到总行
账户信息：各分行存储自己站点的账户信息，同时
汇总到总行

03 数据库设计

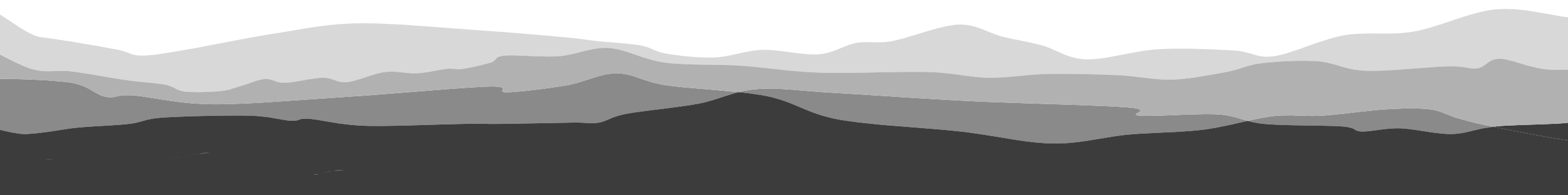


03 数据库设计



03 数据库设计

```
<table name="account_sums" primaryKey="account_id" dataNode="node_main"/>
<table name="accounts" primaryKey="acconut_id" dataNode="node_01,node_02" rule="siteidMod"/>
<table name="deposits" primaryKey="deposits_id" dataNode="node_01,node_02" rule="siteidMod"/>
<table name="sites" primaryKey="site_id" dataNode="node_main"/>
<table name="staffs" primaryKey="staff_id" dataNode="node_01,node_02" rule="siteidMod"/>
<table name="summarys" primaryKey="summary_id" dataNode="node_main"/>
<table name="transfers" primaryKey="transfer_id" autoIncrement="true" dataNode="node_01,node_02" rule="siteidMod"/>
<table name="users" primaryKey="u_id" dataNode="node_main"/>
<table name="withdrawals" primaryKey="withdrawals_id" dataNode="node_01,node_02" rule="siteidMod"/>
<!-- <table name="oc_call" primaryKey="ID" dataNode="dn1$0-743" rule="latest-month-callldate"
| /> -->
```



03 数据库设计

```
<dataNode name="node_main" dataHost="data_host_main" database="bank" />  
<dataNode name="node_01" dataHost="data_host_1" database="bank" />  
<dataNode name="node_02" dataHost="data_host_2" database="bank" />  
<!--<dataNode name="node_db04" dataHost="data_host_1" database="db04" /> -->
```

03 数据库设计

```
<dataHost name="data_host_main" maxCon="1000" minCon="10" balance="0" writeType="0" dbType="mysql" dbDriver="native">  
  <heartbeat>select user()</heartbeat>  
  <writeHost host="server1" url="123.206.90.65:3306" user="root" password=" " />  
</dataHost>
```

```
<dataHost name="data_host_1" maxCon="1000" minCon="10" balance="0" writeType="0" dbType="mysql" dbDriver="native">  
  <heartbeat>select user()</heartbeat>  
  <writeHost host="server1" url="123.206.90.65:3307" user="root" password=" " />  
</dataHost>
```

```
<dataHost name="data_host_2" maxCon="1000" minCon="10" balance="0" writeType="0" dbType="mysql" dbDriver="native">  
  <heartbeat>select user()</heartbeat>  
  <writeHost host="server1" url="127.0.0.1:3306" user="root" password=" " />  
</dataHost>
```

A stylized, layered mountain range background. The mountains are represented by overlapping, wavy shapes in various shades of gray, from light to dark. The foreground is a solid dark gray.

04

实现

04 实现

⚙ 银行管理系统

🏠 主页

💰 存款

🏧 取款

📄 交易查询

💳 转账

📋 职员信息

👤 用户管理

👤 账户管理

存款业务

账号

金额

职员号

提交

获取所有账户

感谢聆听！

