文档编号：

**合同管理系统**

**数据结构说明书**

版本号：

受控编号：

编写部门：数据库实践第五小组

编写人：王宝莹

审核人：袁岗老师

审核日期：2018年12月18日

批准人：

日期：2018年12月18日

**目 录**

[1．引言](#Index1)……………………………………………………………………….. 1

[编写目的](#Index1_1)

[背景](#Index1_2)

[定义](#Index1_3)

[参考资料](#Index1_4)

[2．数据分析](#Index2)………………………………………………………………….. 1

[数据内容分析](#Index2_1)

[数据频度分析](#Index2_2)

[数据量分析](#Index2_3)

[数据结构模型](#Index2_4)

[3．数据关系分析](#Index3)…………………………………………………………….. 1

[4．整体设计](#Index4)………………………………………………………………….. 1

[采用的数据库](#Index4_1)

[数据库名称](#Index4_2)

[数据库说明](#Index4_3)

[各种数据要求的实质](#Index4_4)

[5．表格设计](#Index5)………………………………………………………………….. 1

[表格列表](#Index5_1)

[各表格内部设计](#Index5_2)

[表格之间关系](#Index5_3)

[6．数据维护方案](#Index6)…………………………………………………………….. 1

[日常维护](#Index6_1)

[扩展方式](#Index6_2)

[备份及恢复](#Index6_3)

[其它](#Index6_4)

[7．数据库结构图](#Index7)…………………………………………………………….. 1

**1．引言**

1.1) 编写目的

本文档是合同系统设计文档的组成部分.

编写数据库设计文档的目的是：明确数据库的表名、字段名等数据信息，用来指导后期的数据库脚本的开发，本文档遵循《数据库设计和开发规范》。

本文档的读者对象是需求人员、系统设计人员、开发人员、测试人员。

1.2) 背景

系统名称：合同管理系统。

项目的提出者：北京交通大学

开发者：数据库实践第五小组

用户：公司

运行地点：Windows10系统

1.3) 定义

|  |  |  |
| --- | --- | --- |
| 序号 | 术语或缩略语 | 说明性定义 |
| 1 | PM | Project Manager,项目经理 |
| 2 | DBA | Database Administrator数据库管理员 |

1.4) 参考资料

项目的计划任务书、合同或批文；

项目开发计划；

需求规格说明书；

概要设计说明书；

详细设计说明书；

用户操作手册；

本测试计划中引用的其他资料、采用的软件开发标准或规范。

**2．****数据分析**

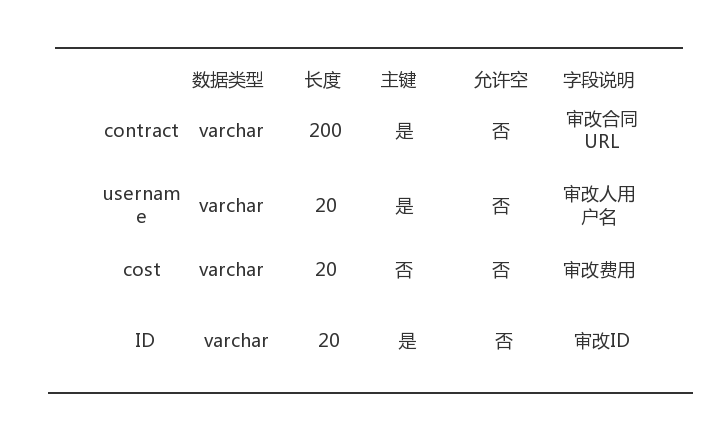
2.1) 数据内容分析

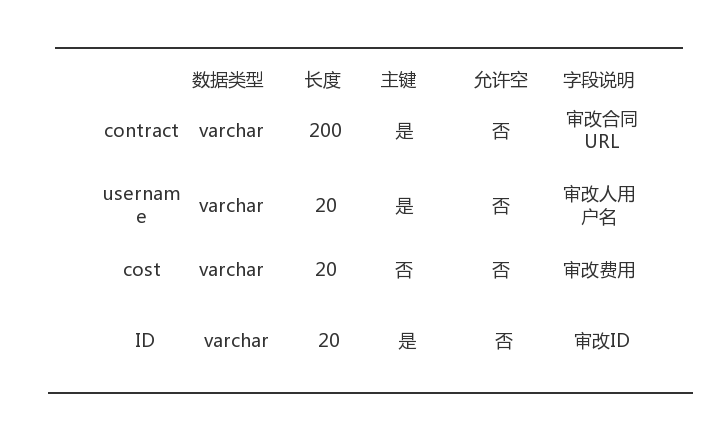
User

Contract\_write

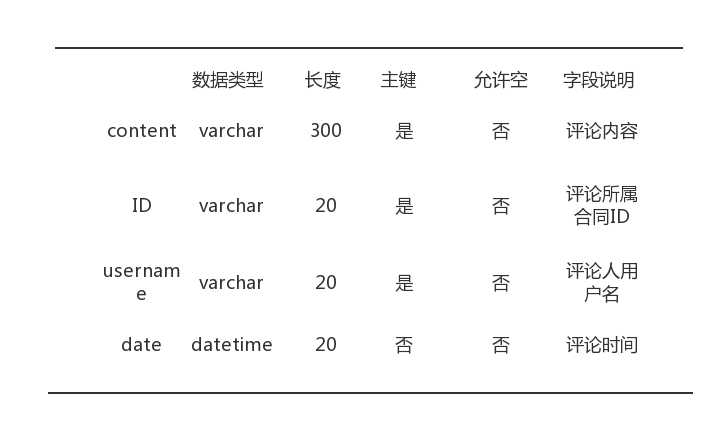


Contract\_audit





Comment



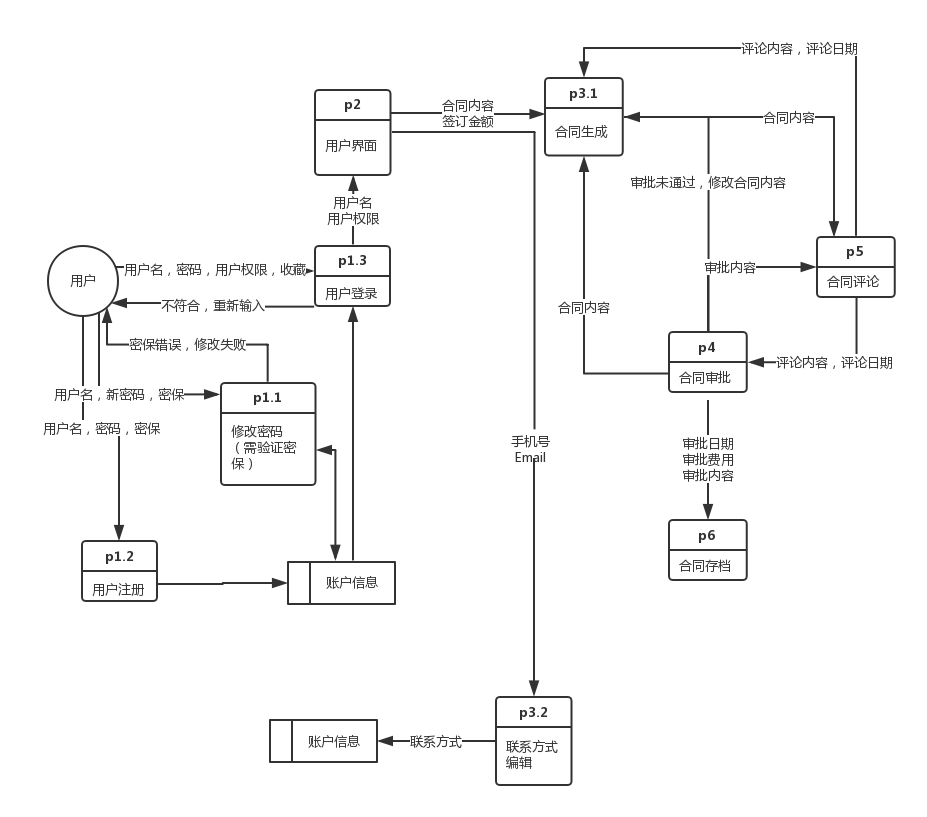
2.2) 数据频度分析

|  |  |
| --- | --- |
| 数据 | 使用频率 |
| username |  |
| password |  |
| collection |  |
| permission |  |
| content |  |
| type |  |
| phonenum |  |
| cost |  |
| email |  |
| ID |  |
| contract |  |
| date |  |
| comment |  |
| audit\_cost |  |

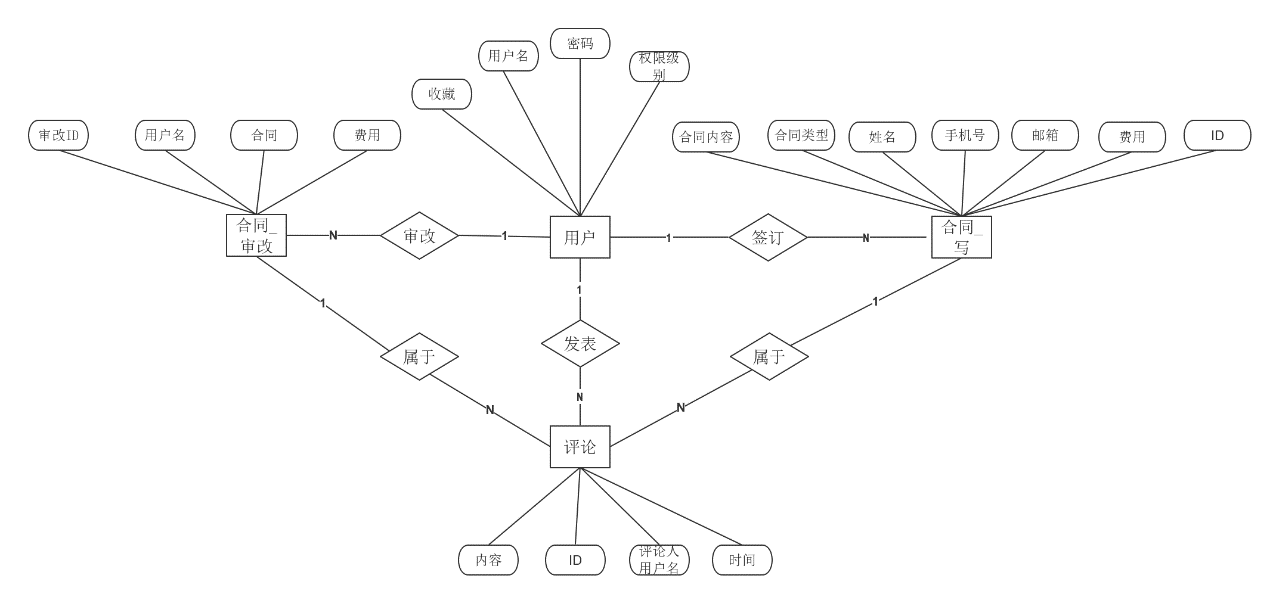
2.3) 数据量分析

|  |  |
| --- | --- |
| 数据 | 数据量 |
| username | 20000 |
| password | 20000 |
| collection | 20000 |
| permission | 20000 |
| content | 100000 |
| type | 25 |
| phonenum | 20000 |
| cost | 100000 |
| email | 20000 |
| ID | 100000 |
| contract | 100000 |
| date | 300000 |
| comment | 300000 |
| audit\_cost | 100000 |

2.4) 数据结构模型



**3．数据关系分析**



**4．整体设计**

4.1) 采用的数据库

Mysql5.7

4.2) 数据库名称

Contract.db

4.3) 数据库说明

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据库实例 | 数据库系统 | 数据库部署环境 | 数据库设计工具 | 数据库存放位置 | 说明 |
| Contract.db | MySql 5.7 | CentOS  1核 - 2GB内存 - 系统40GB SSD云盘 | Visio，  Process On | /db/contract.db | 合同管理系统数据库存取文件 |

**5．表格设计**

5.1) 表格列表

|  |  |
| --- | --- |
| 表名 | 功能说明 |
| User | 存储用户的基本资料和权限信息 |
| contract\_write | 存储写合同时的各种资料 |
| contract\_review | 存储审改合同时的各种资料 |
| comment | 存储评论的各种信息 |

5.2) 各表格内部设计

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表名 | | User | | | | | |
| 主键 | | Username, | | | | | |
| 其他排序字段 | | Password, collection, permission | | | | | |
| 索引字段 | | 无 | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | username | Varchar（20） | N | Y | 高 |  | 主键/用户用户名 |
| 2 | password | Varchar(20) | N | Y | 高 |  | /用户密码 |
| 3 | Collection | Varchar（20） | Y | N | 高 |  | /用户收藏 |
| 4 | Permission | Int（2） | N | Y | 高 |  | /用户权限 |
| Mysql脚本 | | CREATE TABLE use  (  username varchar（20）primary   key  ,password varchar（20） NOT NULL  , collection varchar(20)  , permission int(2) NOT NULL  ); | | | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表名 | | Contract\_write | | | | | |
| 主键 | | Content, ID, username | | | | | |
| 其他排序字段 | | Type, username, phone\_number, email, cost | | | | | |
| 索引字段 | | ID | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | content | Varchar（5000） | N | Y | 高 |  | 主键/合同内容 |
| 2 | Type | Varchar(20) | N | N | 高 |  | /合同类型 |
| 3 | Username | Varchar(20) | N | N | 高 |  | 主键、外键/签订者用户名 |
| 4 | Phone\_number | Varchar(20) | Y | N | 低 |  | /签订者手机号 |
| 5 | Email | Varchar(20) | Y | N | 低 |  | /签订者邮箱 |
| 6 | Cost | Varchar(20) | N | N | 中 |  | /签订费用 |
| 7 | ID | Varchar(20) | N | Y | 高 |  | /合同ID |
| Mysql脚本 | | CREATE TABLE contract\_write  (  content varchar(5000）primary   key  ,type varchar（20） NOT NULL  , username varchar(20) primary key  , phone\_number varchar(20)  ,email varchar(20)  ,cost varchar(20) NOT NULL  , ID varchar(20) primary key  ,foreign key(username) references user(username)  ); | | | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表名 | | Contract\_review | | | | | |
| 主键 | | Contract, username, ID | | | | | |
| 其他排序字段 | | cost | | | | | |
| 索引字段 | | 无 | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | contract | Varchar（200） | N | Y | 高 |  | 主键/审改合同URL |
| 2 | username | Varchar(20) | N | Y | 高 |  | 主键，外键/审改人用户名 |
| 3 | Cost | Varchar(20) | N | N | 高 |  | /审改费用 |
| 4 | ID | Int(2) | N | Y | 高 |  | 主键/审改ID |
| Mysql脚本 | | CREATE TABLE Contract\_review  (  contract varchar（200）primary   key  ,username varchar（20） primary key  , cost varchar(20) NOT NULL  , ID varchar(20) primary key  Foreign key(username) references user(username)  ); | | | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表名 | | Comment | | | | | |
| 主键 | | Content,ID,username | | | | | |
| 其他排序字段 | | date | | | | | |
| 索引字段 | | 无 | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | Content | Varchar（300） | N | Y | 高 |  | 主键/评论内容 |
| 2 | ID | Varchar(20) | N | Y | 高 |  | 主键，外键/评论所属合同ID |
| 3 | Username | Varchar(20) | N | N | 高 |  | 主键，外键/评论人用户名 |
| 4 | Date | Datetime | N | Y | 高 |  | /评论时间 |
| Mysql脚本 | | CREATE TABLE Comment  (  content varchar（300）primary   key  ,ID varchar（20） primary key  , username varchar(20) primary key  , date datetime NOT NULL  foreign key(username) references user(username),  foreign key(ID) references contract\_write(ID)  ); | | | | | |

5.3) 表格间关系

无。

**6．数据维护方案**

6.1) 日常维护

一、备份系统数据

　　SYBASE 系统的备份与恢复机制保证了在系统失败时重新获取数据的可能性。SQL Server 提供了两种不同类型的恢复机制：一类是系统自动完成的恢复，这种措施在每次系统启动时都自动进行，保证了在系统瘫痪前完成的事务都写到数据库设备上，而未完成的事务都被回退；另一类是人工完成的恢复，这是通过 DUMP 和 LOAD 命令来执行人工备份和恢复工作。因此定期备份事务日志和数据库是一项十分重要的日常维护工作。

　　1、备份数据库

　　每一个数据库都应在创建之后卸出，从而提供一个装入基点。在此之后按排定的时间周期表卸出。比如每周五卸出数据库。对一般数据库系统卸出数据库周期建议为每周一次。

　　除了按计划周期卸出数据库之外，还需在每次运行没有日志的操作后卸出数据库。例如：

　　·每次强制地运行了 DUMP TRAN WITH NO\_LOG （因为数据库的磁盘空溢出）；

　　·每次用 sp\_dboption 允许 select into/bulkcopy 做快速拷贝，或用 SELECT INTO 命令创建一个永久性的表，或使用了 WRITETEXT 命令。

　　卸出数据库的命令为：

　　DUMP DATABASE database\_name

　　TO dump\_device

　　database\_name 是要卸出的数据库名称，dump\_device 是卸出设备的名称。用系统过程 sp\_helpdevice 可以获得设备的信息。

　　下面一条命令用来卸出数据库 my\_db ：

　　DUMP DATABASE my\_db

　　TO db\_bk\_dev

　　2、备份事务日志

　　如果事务日志与数据库放在同一个设备上，则事务日志不应与数据库分开备份。master 数据库和小于 4M 的用户数据库就是这种情况。一般数据库系统的数据库和日志分别放在不同的设备上，因此，可以用 DUMP TRAN 命令单独备份日志。

　　备份事务日志的周期直接影响数据的恢复程度，因此建议每天备份。

　　备份事务日志的命令格式为：

　　DUMP TRANsaction database\_name

　　[TO dump\_device]

　　[WITH TRUNCATE\_ONLYWITH NO\_LOGWITH NO\_TRUNCATE]

　　其中 database\_name 是要备份事务的数据库名称，dump\_device 是备份设备名称，仅当包含了 WITH TRUNCATE\_ONLY 或 WITH NO\_LOG 子句时，才可以备份到设备。

　　注意：如果总是用 DUMP DATEBASE （备份数据库及其日志），而不用 DUMP TRAN ，事务日志将不会刷新，而变得非常庞大。

　　对于 master 数据库和小型数据库每次运行 DUMP DATEBASE 之后应当运行 DUMP TRANsaction 命令刷新日志 。

　　下面一条命令备份数据库 db160 的事务日志到备份设备上：

　　DUMP TRANsaction db160

　　TO db\_log\_bk\_dev

　　WITH TRUNCATE\_ONLY

　　3、备份数据库及其日志间的相互作用

　　在至少卸出一次数据库前，卸出事务日志是毫无意义的。下图显示了备份数据库及其日志间的关系

　　如果在星期二下午5：01出现非硬件故障，需要做的所有工作是装入磁带5（参见下一节：数据恢复），由于磁带5是下午5：00刚备份的，因此只有备份和装入之间的一分钟内的数据损失。

　　但是，如果在星期二下午4：49失效会怎么样呢？在这种情况下，要装入磁带1（在星期五下午5：00的卸出）。然后，依次装入磁带2，3以及4。这样，系统将恢复到星期二上午10：00点的状态，星期二的大部分工作丢失了。此例显示了经常卸出事务的重要性。

　　二、万一系统失败时恢复数据库系统

　　如果用户数据库存储的设备失效，从而数据库被破坏或不可存取，通过装入最新的数据库备份以及后来的事务日志备份可以恢复数据库。假设当前的事务日志存在于一个并没有毁坏的设备上，带着 WITH NO\_TRUNCATE 选项的 DUMP TRANsaction 命令卸出它。

　　要恢复数据库按如下步骤去做：

　　1、如果日志存在于一个分离的设备上，用带着 NO\_TRUNCATE 选项的 DUMP TRANsaction 命令卸出被毁坏的或者不可存取的用户数据库事务日志。

　　2、用下面的查询检查设备分配已毁坏数据库的设备使用情况。必须为同一目的赋同样的空间块。

　　下面的查询显示了分配给数据库 mydb 设备使用和尺寸情况：

　　SELECT segmap，size FROM　sysusages

　　WHERE dbid =

　　（ SELECT dbid FROM sysdatabases WHERE name = “mydb”）

　　3、检查查询的输出。在 segmap 列的 ‘3’代表数据分配，‘4’代表日志分配。size 列代表 2K 数据块的数目。注意此信息的次序、使用和尺寸部分。例如，输出为：

　　segmapSize

　　--------------------

　　310240//实际尺寸为：20M

　　35120//实际尺寸为：10M

　　45120//实际尺寸为：10M

　　31024//实际尺寸为：2M

　　42048//实际尺寸为：4M

　　4、用 DROP DATABASE 命令删除毁坏设备上的数据库。如果系统报错，用DBCC DBREPAIR 命令的 DROPDB 选项。

　　5、删除数据库后，用 sp\_dropdevice 删除毁坏了的设备。

　　6、用 DISK INIT 初始化新的数据库设备。

　　7、重建数据库。用 CREATE DATABASE 命令从老的 sysusages 表拷贝所有的行，并包含第一逻辑设备。

　　对上例，命令为：

　　CREATE DATABASE mydb

　　ON datadev1=20，datadev2=10

　　LOG ON logdev1=10

　　8、用 ALTER DATABASE 命令重建其余入口。在此例中，在datadev1上分配更多的空间，命令为：

　　ALTER DATABASE mydb ON datadev1=2

　　9、用 LOAD DATABASE 重新装入数据库，然后用 LOAD TRAN 装入前面卸出的日志。

　　LOAD DATABASE 命令语法是：

　　LOAD DATABASE database\_name

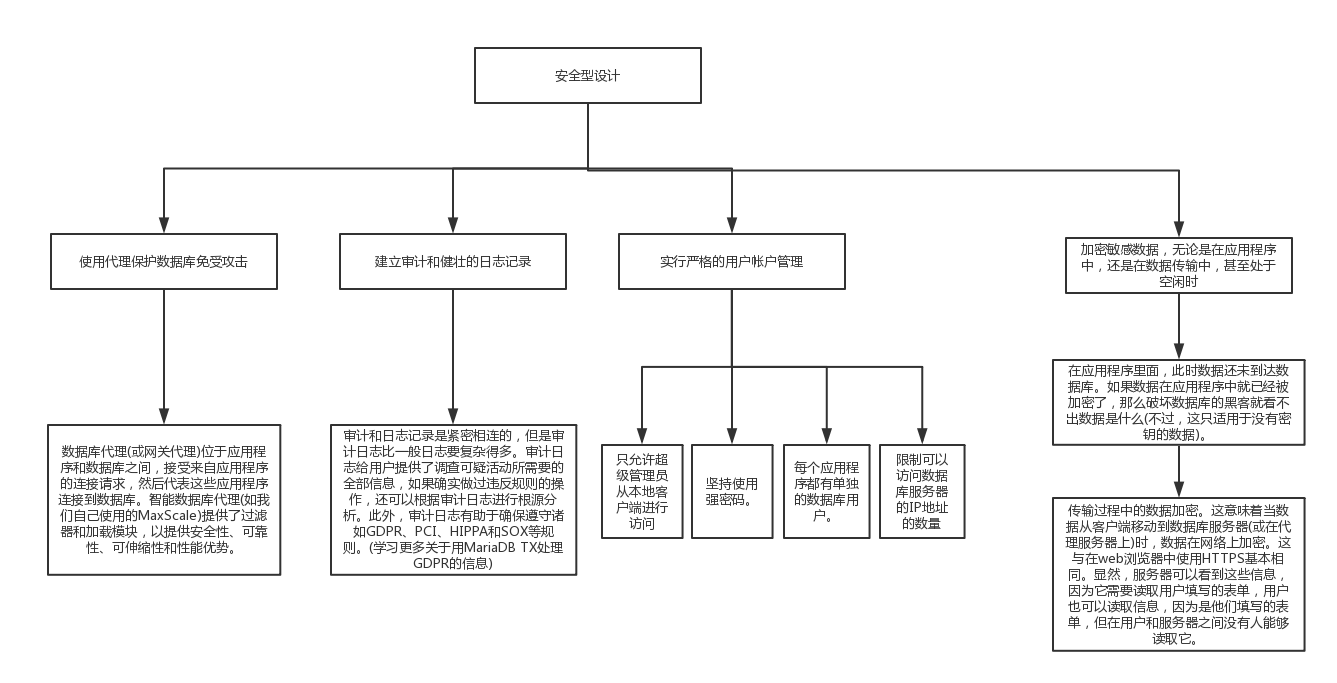
　　FROM dump\_device

　　LOAD TRANsaction 命令的语法是：

　　LOAD TRANsaction database\_name

　　FROM dump\_device

　　卸出数据库和事务日志的缺省权限归数据库所有者，且可以传递给其他用户；装载数据库和事务的权限也归数据库所有者，但不能传递。



6.4) 其它

数据库系统在信息化建设中的重要地位和作用告诉我们，数据库的日常管理与维护不容小视。为保证数据库数据的安全，企业应该做到未雨绸缪。

　　（一）完善管理制度，强化监管力度。数据库系统的安全与企业自身内部的安全机制、内外网络环境、从业人员素质等密切相关。因此，企业应该完善网络系统安全规章制度，防范因制度缺陷带来的风险；企业应该规范操作流程和故障处理流程，减少人为失误与故障，提高故障处理速度，缩短故障处理时间；企业应该通过建立科学合理的责任追究机制，防止出现由于工作态度、工作作风等各种人为因素导致的数据库安全事故。

　　（二）采取措施，确保数据库数据的安全。保证数据库数据的安全是数据库日常管理与维护工作的首要任务，企业需要采取的安全措施主要有：

　　确保网络及操作系统安全。网络系统是数据库应用的外部环境和基础，网络系统安全是数据库安全的第一道屏障。从技术角度讲，网络系统层次的安全防范技术有很多种，大致可以分为防火墙、数字签名与认证、入侵检测等。操作系统是数据库系统的运行平台，能够为数据库系统提供一定程度的安全保护。操作系统的安全控制方法主要是采用隔离控制、访问控制、信息加密和审计跟踪。主要安全技术有操作系统安全策略、安全管理策略等。

　　加强用户身份验证。用户身份验证是数据库系统的重要防线。利用窗体身份验证数据库程序的漏洞，进而获取存储在数据库中的用户身份验证密码，这是目前对网络数据库攻击最常见的方式。对此，企业信息部门通常使用带有salt值的单向密码哈希值，以避免用户密码在数据库中以明文形式存储，减轻字典攻击带来的威胁。

　　对重要数据加密。数据加密交换又称密码学，是计算机系统对信息进行保护的一种最可靠的办法。它利用密码技术对信息进行交换，实现信息隐蔽，从而有效保护信息的安全不受侵犯。数据库加密要求加解密的粒度是每个记录的字段数据。采用库外口加密的方式，对密钥的管理较为简单，只需借用文件加密的密钥管理方法，将加密后的数据块纳入数据库，在算法或数据库系统中做些必要的改动就行。这样有利于公共数据字典的使用和维护系统的完整性。

做好数据库备份与恢复。数据备份是备份数据库某个时刻的数据状态，当系统出现意外时用来恢复系统。依靠网络办公的企业，其信息系统很可能随时被破坏而丢失数据。因此，数据库管理系统必须具备把数据库从错误状态恢复到某一已知的正确状态的功能，这就是数据库的恢复技术。

（三）开展数据库健康检查。为及时发现数据库系统存在的问题，在日常管理与维护中，数据管理员要对数据库开展健康检查。当前，烟草行业内大部分企业使用的数据库是IBM DB2，对该数据库进行检查时，检查内容主要包括以下七个方面：

1、系统环境：操作系统版本、文件系统容量、内存交换区使用率、系统性能。

2、数据库环境：数据库和补丁版本、是否有僵尸数据库进程、数据库节点数、是否有其他数据库产品及版本。

3、日志记录：db2diag.log报错、db2inst1.nfy报错、是否有需要处理的DUMP文件。

4、数据库健康状况：表空间利用率和状态、表空间容器利用率和状态、排序溢出、是否需要收集统计信息、是否需要数据重组、活动日志和日志所在文件系统利用率、死锁发生率、锁升级发生率、锁等待的百分比、编目Cache命中率、包Cache命中率、监视堆利用率、数据库堆利用率、数据库缓冲池命中率。

5、数据库维护内容：最近一次统计信息收集时间、最近一次表数据重组时间、最近一次绑定包时间、最近一次数据库备份时间。

6、权限管理：public组的权限是否取消。

7、数据库基本信息记录：数据库内存使用、环境变量。

**7．数据库结构图**

