第 1 章 关系数据库模型和关系运算

要求掌握的基本概念和理论

1. 与网状和层次数据库相比，关系数据库有哪些优点？

（1）数据结构单一，不管实体还是实体之间的联系都用关系来表示；
（2）建立在严格的数学概念基础上，具有坚实的理论基础；
（3）将数据定义和数据操纵统一在一种语言中，使用方便，易学易用。

2. 试述关系模型的完整性规则

答：关系模型的完整性规则包括：实体完整性、参照完整性、用户定义的完整性 。

实体完整性规则是指若属性 A 是基本关系 R 的主属性，则属性 A 不能取空值。

参照完整性：若属性(或属性组)F 是基本关系 R 的外键，它与基本关系 S 的主码 Ks 相对应(基本关系 R 和 S 不一定是不同的关系)，则对于 R 中每个元组在 F 上的值必须为：或者取空值(F 的每个属性值均为空值)；或者等于 S 中某个元组的主码值。

用户定义的完整性指数据间应满足的语义约束关系，由用户定义，由系统检查。

3. 试述等值连接与自然连接的区别和联系

答：连接运算符是“=”的连接运算称为等值连接。它是从关系 R 与 S 的广义笛卡尔积中选取 A，B 属性值相等的那些元组

自然连接是一种特殊的等值连接，它要求两个关系中进行比较的分量必须是相同的属性组，并且在结果中把重复的属性列去掉。

4. 函数依赖、部分依赖、完全依赖、传递依赖、平凡依赖

函数依赖：对 X 中的任一值 x，$\Pi_Y(\sigma_{X=x}(r))$ 的值仅有一个元组，则有 X→Y。

设 FD X→Y，如果对任意的 X⊂′X，X′→Y 都不成立，则称 X→Y 是**完全函数依赖**；若对 X 的真子集 X′有 X⊂′X，而 X′→Y 成立，则称 FD X→Y 是**部分函数依赖**。

设关系模式 R，X、Y、Z 是 R 的属性子集，若 FD X→Y，Y ！→ X，Y→Z，则有 FD X→Z，称 FD X→Z 为**传递函数依赖。**

设 FD X→Y，如果 Y⊄X，则称 FD X→Y 为非平凡的函数依赖；否则，若 Y⊆X，称 FD X→Y 为**平凡的函数依赖**。

5. 函数依赖公理和推论

Armstrong 公理：设 r 是 R(U)上的一个关系，X、Y、Z、W⊆U。

1.自反律：若 Y⊆X⊆U，则 X→Y;2.增广律：若 X→Y 且 Z⊆U，则 XZ→YZ;3.传递律：若 X→Y，Y→Z，则 X→Z。

推论 1：若 X→Y，X→Z，则 X→YZ

推论 2：若 X→Y 且 Z⊆Y，则 X→Z

推论 3：若 X→Y，YZ→W，则 XZ→W。

6. 函数依赖的覆盖和等价

对于在模式 R 上的函数依赖集 F 和 G，如果对 G 中的每一个函数依赖 X→Y，都有 F|=X→Y，**称 F 是 G 的一个覆盖**。记为：F|= G。在模式 R 上的函数依赖集 F 和 G，若 $F^+=G^+$，则称 F 和 G **等价**，记作 F≡G。

如果函数依赖集 F 不存在真子集 F′使 F′≡F 成立，则 F 是无冗余的。如果 F 是 G 的一个覆盖且 F 是无冗余的，则 F 是 G 的一个**无冗余覆盖。**

如果函数依赖集 F 是 G 的一个覆盖，F 中的每个 FD 都具有 X→A 形式而且 F 是左化简的和无冗余的，称 F 是 G 的一个**规范覆盖**。

要求掌握的基本算法

1. 求关系的并、差、交、连接、选择、投影、除运算。

除法：

2. 关系运算在查询中的应用。
3. 属性集 X 关于 F 的闭包 X+的基本算法
4. 函数依赖集的成员测试算法(MEMBER(F,X→Y)).
5. 检验分解是无损算法
6. 检验分解算法是否保持函数依赖
7. 生成 3NF 的分解算法

DECOMPOSE(R, K, F)

算法步骤：(1). 若R∈3NF，算法终止， ρ={R}。

(2).若 ρ 中有R_i∉3NF，即Y⊆R_i，Z∉KY且 K→Y， Y↛K，Y→Z, 则Z传递依赖于R_i中的键K，分解R_i为：

R_{i1}=R_i—Z和R_{i2}= YZ，用R_{i1}和R_{i2}代替 ρ 中的R_i。

(3). 若 ρ 中所有R_i ∈3NF，输出 ρ，否则转(2) 继续进行分解，直到使所有关系模式都成为3NF。

F={A→BCDEGH，BCD→A，BCE→A，DG→M，H→IJK， EG→M，IJ→K，IK→J，JK→I}。

解：R的键为K={A，BCD，BCE}，R∉3NF。

因有DG→M，而DG不是键，A→DG,即M传递依赖于A，分解R为：

R_1=ABCDEGHIJK， K_1={A，BCD，BCE}；

R_2=DGM， K_2={DG}。

R_2∈3NF，R_1∉3NF，R_1中有H→IJK， 分解R_1为：

R_{11}=ABCDEGH， K_{11}={A，BCD，BCE}；

R_{12}=HIJK， K_{12}={H}；

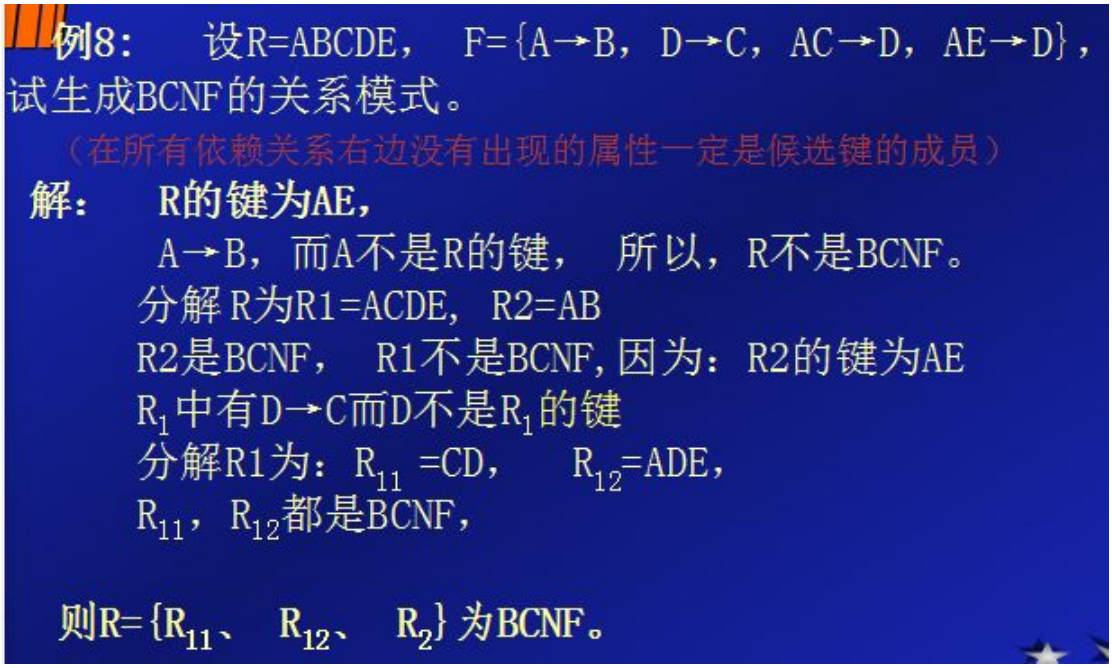
R_{12}∉3NF， 因 IJ→K，分解R_{12}为：

R_{121}=HIJ， K_{121}={H}；

R_{122}=IJK， K_{122}={IJ，IK，JK}

结果：ρ={R_{11}，R_{121}，R_{122}，R_2}。

8. 规范化关系模式为 BCNF 算法



练习 1. R(A,B,C), 其函数依赖集为 F ={ B →C, AC →B };该关系模式是否第 2 范式,并说明理由

练习 2： R(A,B,C,D), 其函数依赖集为 F ={ A →C,AD →B };
该关系模式是否第 2 范式,并说明理由

练习 3. R(A,B,C), 其函数依赖集为 F ={ B →C, AC →B };该关系模式是否第 3 范式,并说明理由

练习 4： R(A,B,C，D), 其函数依赖集为 F ={AB →C,
C →D };该关系模式是否第 3 范式,并说明理由

练习 5：假定一门课只有一个系来开，找出选课关系 elective 的键和和基本函数依赖，它是否是第 2 范式？

练习 6：假定一门课只有一个系开，一个系只有一个地址？该关系中有哪些函数

依赖？该关系的键是什么？是几范式？

1.是第三范式，不是 BCNF。 主键为 AC，非主属性为 B，B 完全依赖于 AC，而 C 传递依赖 AC。
2.不是 主键为 AD，非主属性为 B 和 C ，对于 C，C 部分依赖于 AD（A->C）
3.是 3NF.主键为 AC，非主属性为 B，不存在传递依赖
4.不是 3NF,是 2NF.主键为 AB,非主属性为 C D，D 传递依赖与 AB？？
5.SNAME、COURSE→DEPT
COURSE →DEPT
不是 2NF,为 1NF。 主键为 SNAME COURSE,非主属性为 DEPT，DEPT 部分依赖于 COURSE
6.COURSE→DEPT， DEPT→BUILDING
为第二范式，主键为 COURSE，非主属性为 DEPT,BUILDING，存在非主属性的传递依赖，不符合 3NF。

7.指出下列关系模式是第几范式,并说明理由
（1） R(A,B,C), 其函数依赖集为 F ={ B →C, AC →B };
（2） R(A,B,C), 其函数依赖集为 F ={AB →C};
（3） R(A,B,C), 其函数依赖集为 F ={ A →B , A→C };
（4） R(A,B,C,D), 其函数依赖集为 F ={ A →C , AD→B };
（5） R(A,B,C), 其函数依赖集为 F ={ B →C, B →A , A →BC }

7.（1）第三范式，存在主属性的传递依赖，主属性为 AC，非主属性 B，AC→B→C，不满足 BCNF。而满足 3NF，即不存在非主属性的传递依赖。
（2）BCNF 范式，主属性 AB,非主属性 C，不存在传递依赖和部分依赖，故为 BCNF 范式。
（3）BCNF 范式，主属性 A，非主属性 BC，不存在传递依赖和部分依赖，故为 BCNF 范式。
（4）1NF，主属性 AD，非主属性 BC，存在非主属性的部分依赖 A→C，则为第一范式。
（5）BCNF 范式，主键 A（或 B），非主属性为 BC（或 AC），不存在传递依赖和部分依赖，故为 BCNF 范式。这里没有传递依赖， 虽然 A →B→C，但是由于 B→A，这就违背了传递依赖的条件$B \nrightarrow A$。

第 2 章 关系数据库设计和数据库管理系统

要求掌握的基本概念和理论

1. 试述数据库设计过程，及每个阶段的任务。
答：各阶段的设计要点如下：
(1) 需求分析：准确了解与分析用户需求（包括数据与处理）。
(2) 概念结构设计：通过对用户需求进行综合、归纳与抽象，形成一个独立于具

体 DBMS 的概念模型。
(3) 逻辑结构设计：将概念结构转换为某个 DBMS 所支持的数据模型，并对其进行优化。
(4) 数据库物理设计：为逻辑数据模型选取一个最适合应用环境的物理结构（包括存储结构和存取方法）。
(5）数据库实施：设计人员运用 DBMS 提供的数据语言、工具及宿主语言，根据逻辑设计和物理设计的结果建立数据库，编制与调试应用程序，组织数据入库，并进行试运行。
(6) 数据库运行和维护：在数据库系统运行过程中对其进行评价、调整与修改。
这是一个完整的实际数据库及其应用系统的设计过程。不仅包括设计数据库本身，还包括数据库的实施、运行和维护。设计一个完善的数据库应用系统往往是上述六个阶段的不断反复。

2．什么是数据库的逻辑结构设计？试述其设计步骤。
答：数据库的逻辑结构设计就是把概念结构设计阶段设计好的基本 E 一 R 图转换为与选用的 DBMS 产品所支持的数据模型相符合的逻辑结构。设计步骤为
(1 ）将概念结构转换为一般的关系、网状、层次模型；
(2 ）将转换来的关系、网状、层次模型向特定 DBMS 支持下的数据模型转换；
(3 ）对数据模型进行优化。

3. 试述数据库物理设计的内容和步骤。
答：数据库在物理设备上的存储结构与存取方法称为数据库的物理结构，它依赖于给定的 DBMS 。为一个给定的逻辑数据模型选取一个最适合应用要求的物理结构，就是数据库的物理设计的主要内容。数据库的物理设计步骤通常分为两步：
(1 ）确定数据库的物理结构，在关系数据库中主要指存取方法和存储结构；
(2 ）对物理结构进行评价，评价的重点是时间效率和空间效率。

4．数据库管理系统的主要功能有哪些？
（1）数据库定义
(2) 数据操纵
(3) 数据库控制
(4) 数据库维护

5．数据库管理系统有哪几部分组成？
（1）数据和元数据存储
（2）存储管理器
（3）查询处理器
（4）事务管理器
（5）输入模块---模式修改、查询和修改

6．开发一个数据库管理系统的主要技术难点在哪里？对中国如何尽快开发自己的数据库管理系统，给出你的建议。
1、综合统一
SQL 语言将数据定义语言 DDL、数据操纵语言 DML、数据控制语言 DCL

的功能集于一体，语言风格统一，可以独立完成数据库生命周期中的全部活动.
高度非过程化
2. 对用户的透明性：
用 SQL 语言进行数据操作时，只要提出“做什么”，而无需指明“怎么做”。
3、面向集合的操作方式
SQL 语言操作的对象和操作的结果都用关系表示。
4、一种语法，两种使用方式
SQL 语言既是自含式语言，又是嵌入式语言。
5、语言简捷，易学易用
完成核心功能只用 9 个动词，SQL 语言接近英语句子。
6、支持三级模式结构

数据库设计

要求：给出 E---R 图，将其转换为关系模型、指出转换结果中每个关系的候选键。

第 3-4 章 分布式数据库和面向对象数据库

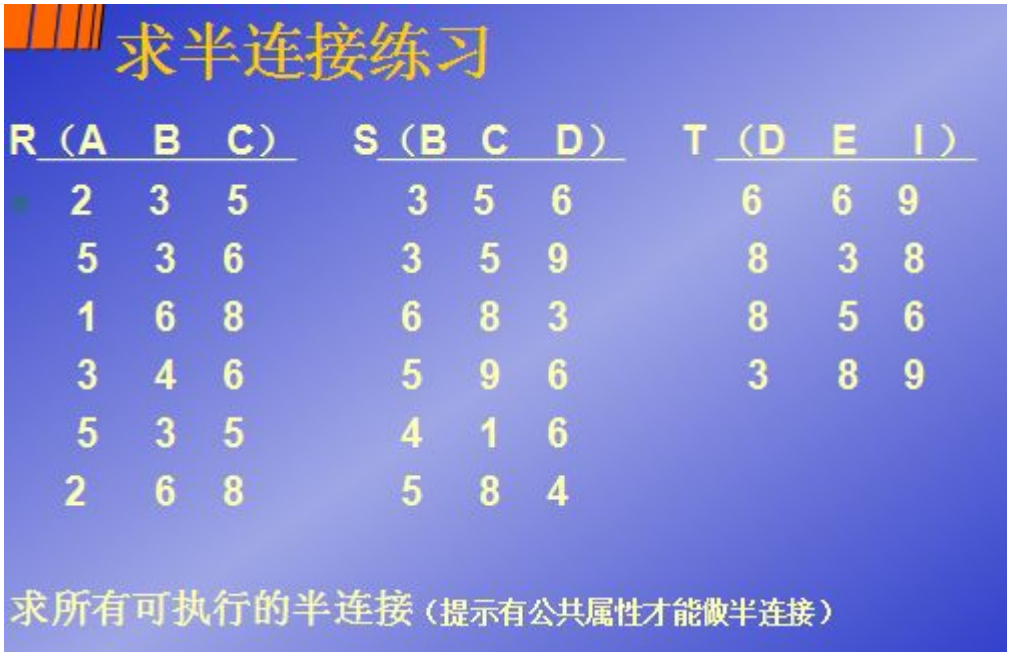
要求掌握的基本概念和理论

1. 分布式数据库的有哪些特点？
特点：1. 数据是分布的 2. 数据是逻辑相关的 3. 结点自治性

2. 分布式数据库管理系统有哪几部分组成？
组成：局部数据库管理系统 LDBMS；
全局数据库管理系统 GDBMS；
全局数据字典 GDD；
网络通信管理 CM

3. 分布式数据库系统能够提供哪些分布透明性？不同透明性对应用程序的编程有什么影响？
分片透明性：关系如何分片对用户是透明的，指用户不必关心数据是如何分片的。其应用程序的编写与集中式数据库相同。
位置透明性：用户需知道数据在哪个片段，而不必知道所操作的数据放在哪个节点。 数据在结点间的转移不会影响应用程序。
局部映象透明性：该透明性提供数据到局部数据库的映象。在编程时不但需要了解全局关系的分片模式，还需要了解各片段存放的站点。

4. 半连接在分布式查询优化中的作用？会计算简单的半连接。
在分布式数据库的查询中半连接的作用：减少传送的数据量，提高查询效率。半连接把笛卡尔乘积和其后的选择运算合并成为连接运算，以避免扫描笛卡尔乘积的中间结果。



$R \propto S =$

$(A$	B	$C)$
2	3	5
1	6	8
2	6	8
5	3	5

$S \propto R =$

$(B$	C	$D)$
3	5	6
3	5	9
6	8	3

$S \propto T =$

$(B$	C	$D)$
3	5	6
5	9	6
4	1	6
6	8	3

$T \propto S =$

$(D$	E	$I)$
6	6	9
3	8	9

5．试述事务的概念及事务的 4 个特性。

事务是用户定义的一个数据库操作序列，这些操作要么全做要么全不做，是一个不可分割的工作单位。

事务具有 4 个特性：原子性（Atomicity ）、一致性（consistency ）、隔离性(Isolation ）和持续性（Durability ）。这 4 个特性也简称为 ACID 特性。

原子性：事务是数据库的逻辑工作单位，事务中包括的诸操作要么都做，要么都不做。

一致性：事务执行的结果必须是使数据库从一个一致性状态变到另一个一致性状态。

隔离性：一个事务的执行不能被其他事务干扰。即一个事务内部的操作及使用的数据对其他并发事务是隔离的，并发执行的各个事务之间不能互相干扰。

持续性也称永久性（Perfnanence），指一个事务一旦提交，它对数据库中数据的改变就应该是永久性的。接下来的其他操作或故障不应该对其执行结果有任何影响。

6. 在数据库管理系统中为什么要采用并发控制技术？常用并发控制技术有哪些？

答：数据库是共享资源，通常有许多个事务同时在运行。当多个事务并发地存取数据库时就会产生同时读取和 / 或修改同一数据的情况。若对并发操作不加控制就可能会存取和存储不正确的数据，破坏数据库的一致性。所以数据库管理系统必须提供并发控制机制。

封锁技术使一组事务的并发执行(即交叉执行)同步，使它等价于这些事务的某一种串行操作；

时戳技术也使一组事务的交叉执行同步，但它等价于这些事务的一个特定的串行操作，即由时戳的时序所确定的一个串行操作执行。

7．什么是两段封锁协议？

两段锁协议是指所有事务必须分两个阶段对数据项加锁和解锁。

在对任何数据进行读、写操作之前，首先要申请并获得对该数据的封锁；

在释放一个封锁之后，事务不再申请和获得任何其他封锁。

“两段”的含义是，事务分为两个阶段：

第一阶段是获得封锁，也称为扩展阶段。在这阶段，事务可以申请获得任何数据项上的任何类型的锁，但是不能释放任何锁。

第二阶段是释放封锁，也称为收缩阶段。在这阶段，事务释放已经获得的锁，但是不能再申请任何锁。

8．数据库恢复的基本技术有哪些？

数据转储和**登录日志**文件是数据库恢复的基本技术。当系统运行过程中发生故障，利用转储的数据库后备副本和日志文件就可以将数据库恢复到故障前的某个一致性状态。

① 转储：数据库管理员定期将整个数据库复制到磁带或另一个磁盘上保存起来的过程。

② 日志：保存每一次对数据库进行更新操作的有关信息的文件，由 DBMS

自动建立和记录。

③ 检查点机制：为了便于恢复，在日志中每隔一定时间（如 10 分钟）写一个检查点，以标识检查点前已经执行完的事务是正确的。检查点记录包括检查点时刻执行的所有事务的标识以及这些事务最近一个运行记录在日志中的地址。

9．什么是日志文件？为什么要设立日志文件？

日志文件是用来记录事务对数据库的更新操作的文件。设立日志文件的目的是：进行事务故障恢复；进行系统故障恢复；协助后备副本进行介质故障恢复。

先写日志文件，即首先把日志记录写到日志文件中，然后写数据库的修改。

10．数据库运行中可能产生的故障有哪几类？

在**集中式数据库**系统发生的故障，大致可以分以下几类：

（1）事务内部的故障；（2）系统故障；（3）介质故障；

在**分布数据库**运行中，除了上面的三种故障外，还有：

（1） 信息丢失；（2）网络分割

11．试述实现数据库安全性控制的常用方法和技术。

答：实现数据库安全性控制的常用方法和技术有：

(1 ）**用户标识和鉴别**：该方法由系统提供一定的方式让用户标识自己的名字或身份。每次用户要求进入系统时，由系统进行核对，通过鉴定后才提供系统的使用权。

(2 ）**存取控制**：通过用户权限定义和合法权检查确保只有合法权限的用户访问数据库，所有未被授权的人员无法存取数据。

(3 ）**视图机制**：为不同的用户定义视图，通过视图机制把要保密的数据对无权存取的用户隐藏起来，从而自动地对数据提供一定程度的安全保护。

(4 ）**审计**：建立审计日志，把用户对数据库的所有操作自动记录下来放入审计日志中，DBA 可以利用审计跟踪的信息，重现导致数据库现有状况的一系列事件，找出非法存取数据的人、时间和内容等。

(5 ）**数据加密**：对存储和传输的数据进行加密处理，从而使得不知道解密算法的人无法获知数据的内容。

12．给出下列名次的含义

对象、类、封装、继承、多态、对象标识、子类、超类。

对象：在面向对象程序中，一切都是对象，从一个数据元素到一个大的文件以及一个数据结构，一个可执行程序段等都是对象。

类：具有相同特征对象的集合；对象为类中的实例。

继承：继承只有在类按层次排列时才有意义。一个类可以从另一个类中继承其特征，包括数据和方法。

封装： 是一种信息隐蔽技术，它把对象的特征和行为隐蔽起来，使得一个对象在程序中可以作为一个独立的整体使用而不用担心对象的功能受到影响。

多态：表现为同一操作允许有不同的实现细节。

对象标识：每个对象都有一个内部标识符 OID，OID 在整个系统中 是唯一的，

一旦生成就不能改变。
子类：
子类继承超类的属性和方法。
超类：
被继承的类称为超类，也叫做父类，

13. 叙述面向对象模型中“对象标识”与关系模型中的“键”的相同点和不同点。

在面向对象数据库系统中，每个对象都有一个内部标识 OID，用来标识一个对象，对象标识在整个系统中是唯一的。

相同点：都是构成数据库操作的基本单位

不同点：

① “对象标识”可以支持复杂数据类型，而“键”不能很好模拟复杂对象；

② “对象标识”可以支持面向对象的数据模型，“键”构成的数据类型简单，没有定义抽象数据类型的能力

③ “对象标识”利用面向对象的思想，将结构与行为统一；“键”导致了数据的结构与行为完全分离，使数据库中的信息仅能由识别他们的应用程序解释执行；

④ “对象标识”可以让查询方便高效；“键”导致了查询实现复杂，连接优化降低了存取效率。

第 5—8 章 新型数据库

1. 叙述 key/value 的数据结构。

key/value 的数据结构为：域(Domain)＋数据项(Item)

域类似于“表”，但无结构；作用是容纳数据项。

数据项用 Key 定义，所有与一个数据项相关的内容都存储到该数据项中，数据属性全部是字符串类型。

可以将 Key-value 数据存储系统理解为面向数据项的系统，所有与一个数据项相关的内容都存储带该数据项中。在同一个域中存储的数据项可以存在很大的差异。

由于与数据项相关的内容都存储在一个单独的数据项中，因此要获取一个数据项的相关内容无需多个表之间的 Jion 操作。

2. Key/Value 数据模式与关系数据库的比较有哪些优点和缺点？

Key/Value 的优点：

- 便于**扩展**，适于**云计算**的环境
- 与应用程序代码的**兼容性**更好

Key/Value 的缺点：

- 数据**完整性约束转移**至应用程序
- 目前的很多 Key/Value 数据**存储系统**之间**不兼容**
- 在云环境中，很多用户和应用使用同一个系统。为了避免一个进程使共享环境超载，往往**严格限制一个单独的查询**所能够产生的全局影响。

3. 在数据切分机制中，**一致性哈希**算法的基本原理是什么？

一致性哈希算法：哈希函数的输出范围被看作一个固定的“环”。系统中的每个**节点被赋予环中的一个随机值**，该随机值用来表示其在“环”中的位置。每个“键值”对应一个数据项，根据该键值的哈希值可生成数据项在环中的位置 position = hash(key)，然后顺时针沿着环找到 value 大于 position 的第一个节点，这个节点就是该数据项的存储节点。

4．云计算按照服务类型可以分为哪几类？

(1).IaaS(基础架构即服务)

将硬件设备等基础资源封装成服务供用户使用

(2)PaaS（平台即服务）

对资源的抽象层次更进一步，提供用户应用程序运行环境

(3).SaaS（软件即服务）

针对性更强，它将某些特定应用软件功能封装成服务

5. Google 云计算中分布式结构化数据表 Bigtable 的设计动机是什么?

Bigtable 的设计动机主要表现在以下三个方面:

(1)需要存储的数据种类繁多：Google 目前向公众开放的服务很多，需要处理的数据类型也非常多。

(2)海量的服务请求：Google 运行着目前世界上最繁忙的系统，它每时每刻处理的客户服务请求数量是普通的系统根本无法承受的 .

(3)商用数据库无法满足 Google 的需求：一方面现有商用数据库设计着眼点在于通用性，根本无法满足 Google 的苛刻服务要求；另一方面对于底层系统的完全掌控会给后期的系统维护、升级带来极大的便利.

6. 试比较 Hadoop 中的数据库 HBase 和传统关系数据库的不同.

Hbase 和传统关系数据库的不同主要体现在：

（1） 数据类型

Hbase 只有简单的**字符串类型**，所有类型都是交由用户自己处理，它只保存字符串。而关系数据库有丰富的类型选择和存储方式、数据操作，**Hbase 操作只有很简单的插入、查询、删除、清空**等，**表和表之间是分离**的，没有复杂的表和表之间的关系，所以也不能也没有必要实现表和表之间的关联等操作。而传统的关系数据通常有各种各样的函数、连接操作。

（2）存储模式

Hbase 是**基于列存储**的，**每个列族都有几个文件保存，不同列族的文件是分离的**。传统的关系数据库是**基于表格结构和行模式**保存的。

（3）数据维护

Hbase 的更新正确来说应该不叫更新，而是一个主键或者列对应的新的版本，而它旧有的版本仍然会保留，所以它实际上是插入了新的数据，而不是传统关系数据库里面的替换修改。

（4） 可伸缩性

Hbase 和 Bigtable 这类分布式数据库就是直接为了这个目的开发出来的，能够轻易的增加或者减少（在硬件错误的时候）硬件数量，而且对错误的兼容性比较高。而传统的关系数据库通常需要增加中间层才能实现类似的功能。

7. 了解 MapReduce 的基本工作原理。

工作原理:一种**简化**的**并行编程模型**，借用**函数式编程中**的 **map** 和 **reduce** 函数，将复杂的运行于**大规模集群上**的**并行计算**过程**高度的抽象**到了两个阶段:借用了 Lisp 中相似功能的名称，将这两个阶段分别用 Map 函数和 Reduce 函数命名，并将此计算模型命名为 MapReduce，然后**自动分布**到一个由普通机器组成的**超大集群上**并发执行。

MapReduce:通过把对数据集的大规模操作分发给网络上的每个节点实现可靠性（Map）；每个节点会周期性地把完成的工作和状态的更新报告回来（Reduce）。

MapReduce:Map 是把输入 Input 分解成中间的 Key/Value 对，Reduce 把 Key/Value 合成最终输出 Output。

MapReduce 原理的要点：

数据分割、任务调度、故障处理等细节**对程序员透明**；

利用**资源无关性**的原理，提高处理效率；

合理的**任务粒度**，优化容错处理和整体效率；

本地计算:充分利用数据的空间局部性来减少网络传输，节省带宽资源；

减少中间数据的产生，优化网络传输。

8. 了解 Hadoop 中的分布式数据库--- Hbase 的逻辑模型和物理模型。

逻辑模型:

表格里存储一系列的数据行，每行包含一个可排序的行关键字、一个可选的时间戳及一些可能有数据的列（稀疏）

数据行有三种基本类型的定义：

行关键字是数据行在表中唯一标识，时间戳是每次数据操作对应关联的时间戳，列定义为：<family>:<label>（<列族>:<标签>）

物理模型:

物理模型实际上就是把概念模型中的一个行进行分割，并按照列族存储

查询时间戳为 t7 的“contents:”将返回空值，查询时间戳为 t8，“anchor:”值为“look.ca”的项也返回空值 （空的单元格不存储 ）

查询“contents:”而不指明时间戳，将返回 t5 时刻的数据；查询“anchor:”的“look.ca”而不指明时间戳，将返回 t7 时刻的数据 （未指明时间戳，则返回指定列的最新数据值 ）

9. 了解在亚马逊的分布式 Key/value 数据存储与管理系统 Dynamo 中，采用的哪些技术来保证数据的可伸缩性和最终一致性。

数据划分(data partitioned)和使用**一致性哈希**的复制(replicated)，并通过对象版本(object versioning)提供一致性。

问题	技术	优势
划分	一致性哈希	增量可伸缩性
写的高可用性	矢量时钟与读取过程中的协调	版本大小与更新操作速率脱钩。
暂时性的失败处理	法定数量协议和基于提示的切换技术	提供高可用性和耐用性的保证，即使一些副本不可用时。
永久故障恢复	使用 Merkle 树的反熵	在后台同步不同的副本。
会员和故障检测	Gossip 的成员和故障检测协议。	保持对称性并且避免了一个用于存储会员和节点活性信息的集中注册服务节点。

10. 阐述 SQL Azure 和 SQL Server 的相同点和不同点。

（1）. 物理管理和逻辑管理

SQL Azure 在管理上突出强调了物理管理，能够自动复制所有存储数据以提供高可用性，同时还可以管理负载均衡、故障转移等功能

用户不能管理 SQL Azure 的物理资源

SQL Azure 不能使用 SQL Server 备份机制，所有的数据都是自动复制备份

（2）. 服务提供

部署本地 SQL Server 时，需要准备和配置所需要的硬件和软件

用户在 Windows Azure 平台上创建了账户后，便可以使用 SQL Azure 数据库，同时还可以访问所有提供的服务

每个 SQL Azure 订阅都会绑定到微软数据中心的某个 SQL Azure 服务器上

SQL Azure 服务器上的数据库通常会在数据中心其他物理机上进行备份

（3）. Transact-SQL 支持

SQL Azure 中由微软进行物理资源的管理，因而这些类型的参数并不适用于 SQL Azure

（4）. 特征和类型

SQL Azure 不支持 SQL Server 的所有特征和数据类型。在现今版本的 SQL Azure 中，不支持分析、复制、报表和服务代理等服务

11. 大数据的 4V 特征是什么？

大量化(Volume)、多样化(Variety)、快速化(Velocity)、价值密度低（Value）

12. 分布式数据系统的 CAP 原理的三要素是什么？

分布式数据系统的 CAP 原理的三要素：

一致性(Consistency)

可用性(Availability)

分区容忍性(Partition tolerance)

一致性（Consistency）是指执行了一次成功的写操作之后，未来的读操作一定可以读到这个写入的值。

可用性（Availability）(指的是快速获取数据)
每一次操作总是能够在确定的时间返回 。

分区容忍性（Partition-tolerance）系统中任意信息的丢失或失败不会影响系统的继续运作。

13. 几种主流 NoSQL 数据库包括哪些？
(1) BigTable
(2) Dynamo
(3) Cassandra
(4) HBase
(5) Redis
(6) MongoDB

14. 数据仓库数据的基本特征是什么？
四个基本特征是：
数据仓库的数据是面向主题的
数据仓库的数据是集成的
数据仓库的数据是不可更新的
数据仓库的数据是随时间不断变化的

15. 什么是数据挖掘？数据挖掘常用的技术方法哪几种？
数据挖掘是从超大型数据库（VLDB）或数据仓库中发现并提取隐藏在内的模式的过程，这些模式是有效的、新颖的、有潜在使用价值的和易于理解的。目的是帮助决策者寻找数据间潜在的关联，发现经营者被忽略的要素，而这些要素对预测趋势、决策行为也许是十分有用的信息。
常用的技术方法有：人工神经网络、遗传算法、决策树方法、粗集方法等。