

数据管理技术的发展



讲授内容

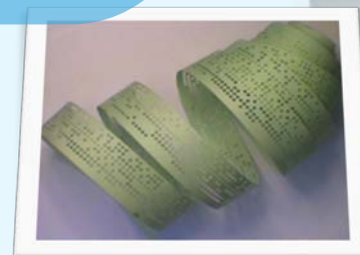
▶ 数据管理的三个阶段

- 1 人工管理
- 2 文件系统管理
- 3 数据库系统管理



人工管理

- 20世纪50年代中期以前
- 数据处理方式：批处理
- 计算机主要用于科学计算
- 硬件：外部存储器只有磁带、卡片和纸带等
- 软件：没有出现操作系统，无数据管理方面的软件





人工管理

数据管理特点:

- 数据面向应用

数据需要由应用程序自己设计、说明（定义）和管理，程序员在编写程序时自己规定数据的存储结构、存取方法和输入方式等。

例：用C语言编程求10个数据之和

```
/*程序 1： 求 10 个数之和*/  
#include <stdio.h>  
main ()  
{  
    int i,s=0;  
    int a[10]={66,55,75,42,86,77,96,89,78,56};  
    for (i=0;i<10;i++)  
        s=s+a[i];  
    printf("%d",s);  
}
```



人工管理

数据管理特点:

- 数据面向应用
- 数据不保存

内存

```
/*程序 1: 求 10 个数之和*/  
#include <stdio.h>  
main ()  
{  
    int i,s=0;  
    int a[10]={66,55,75,42,86,77,96,89,78,56};  
    for (i=0;i<10;i++)  
        s=s+a[i];  
    printf("%d",s);  
}
```

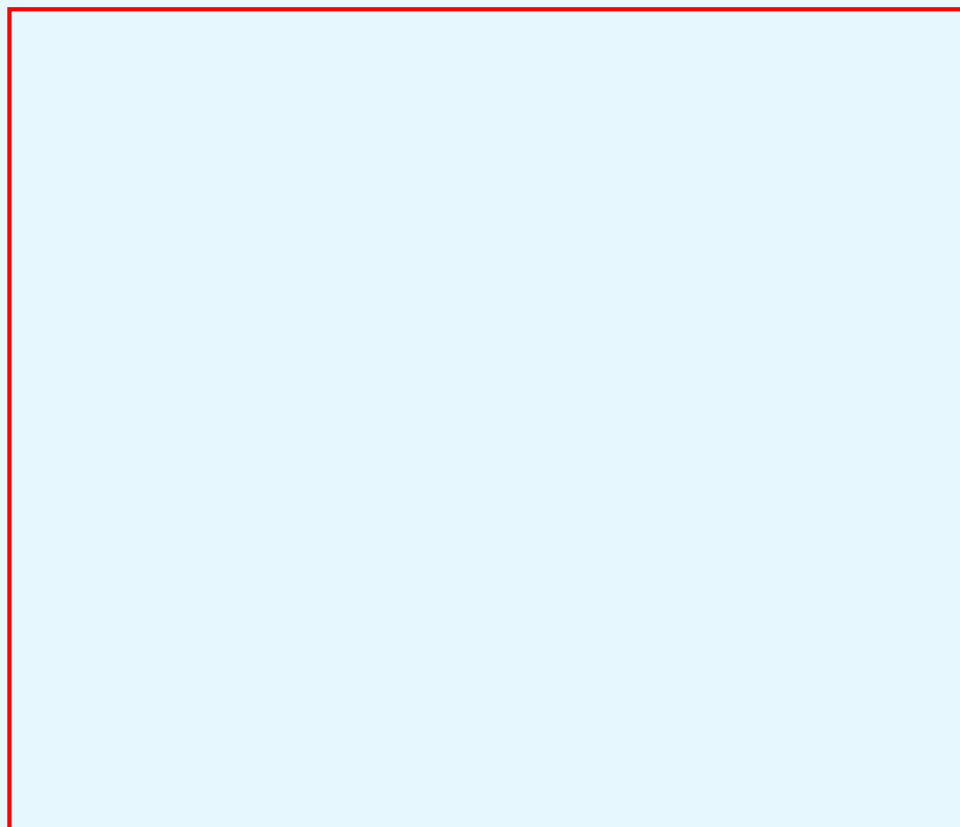


人工管理

数据管理特点：

- 数据面向应用
- 数据不保存

内存





人工管理

数据管理特点:

- 数据面向应用
- 数据不保存
- 数据不能共享

例：用两个C语言程序，求10个数据之和和最大值

```
/*程序 1：求 10 个数之和*/  
#include <stdio.h>  
main ()  
{  
    int i,s=0;  
    int a[10]={66,55,75,42,86,77,96,89,78,56};  
    for (i=0;i<10;i++)  
        s=s+a[i];  
    printf("%d",s);  
}
```

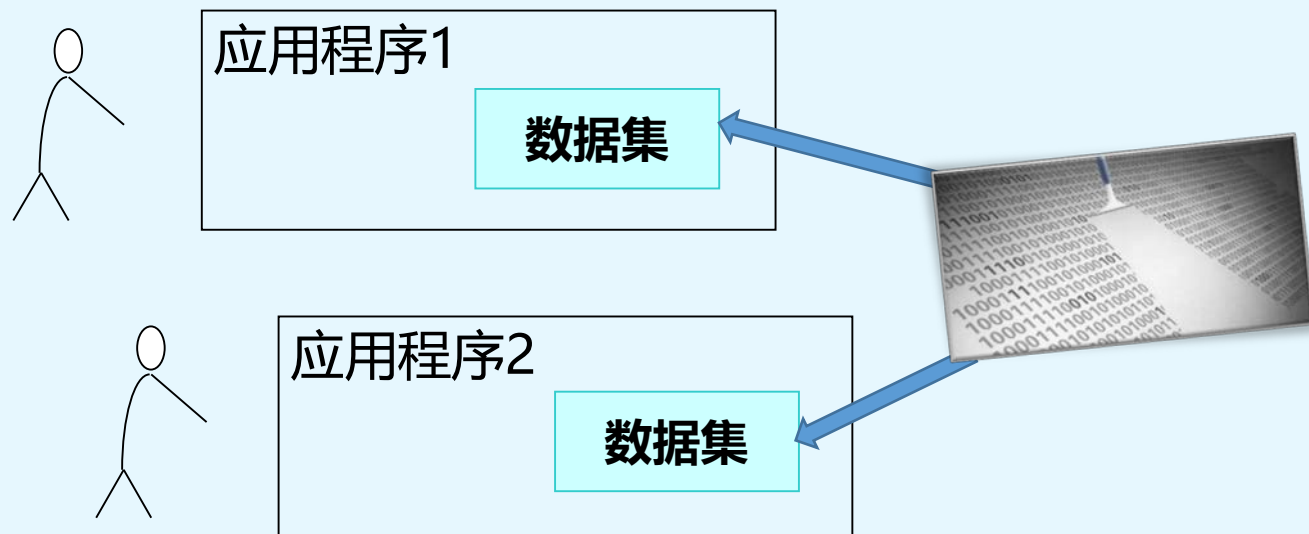
```
/*程序 2：求 10 个数中的最大值*/  
#include<stdio.h>  
main()  
{  
    int i,s;  
    int a [10]={66,55,75,42,86,77,96,89,78,56};  
    s=a [0];  
    for (i=1;i<10;i++)  
        if (s<a[i]) s=a[i];  
    printf("%d",s);  
}
```



人工管理

数据管理特点：

- 数据面向应用
- 数据不保存
- 数据不能共享





人工管理

数据管理特点：

- 数据面向应用
- 数据不保存
- 数据不能共享
- 不具有数据独立性



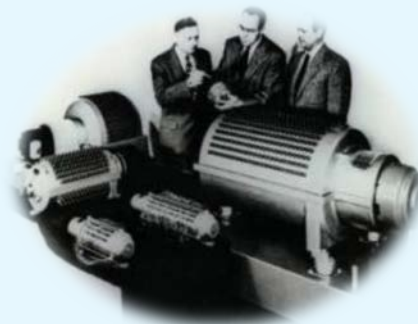
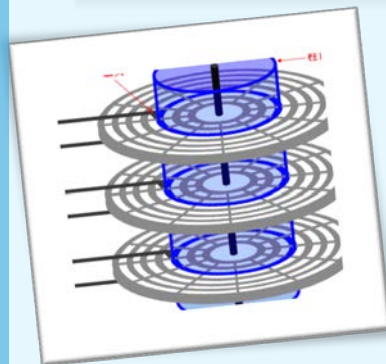
数据独立性是指用户的应用程序与数据的逻辑结构和物理结构是相互独立的。



文件系统管理

- 20世纪50年代末到60年代中期
- 计算机大量用于数据管理
- 硬件：磁盘、磁鼓等直接存取设备
- 软件：操作系统中文件系统专门管理数据
- 数据处理方式：不仅能进行批处理，还能进行联机实时处理

```
Starting MS-DOS 7.1...  
  
The following file is missing or corrupted: .\HIMEM.SYS  
  
Microsoft(R) MS-DOS 7.1  
(C)Copyright Microsoft Corp 1981-1999.  
C:\>_
```





文件系统管理

例：用两个C语言程序，求10个数据之和和最大值

/*程序 1：求 10 个数之和*/

```
#include <stdio.h>
```

```
main ()
```

```
{
```

```
    int i,s=0;
```

```
    int a[10]={66,55,75,42,86,77,96,89,78,56};
```

```
    for (i=0;i<10;i++)
```

```
        s=s+a[i];
```

```
    printf("%d",s);
```

```
}
```

/*程序 2：求 10 个数中的最大值*/

```
#include<stdio.h>
```

```
main()
```

```
{
```

```
    int i,s;
```

```
    int a [10]={66,55,75,42,86,77,96,89,78,56};
```

```
    s=a [0];
```

```
    for (i=1;i<10;i++)
```

```
        if (s<a[i]) s=a[i];
```

```
    printf("%d",s);
```

```
}
```

文件 C:\Data.dat

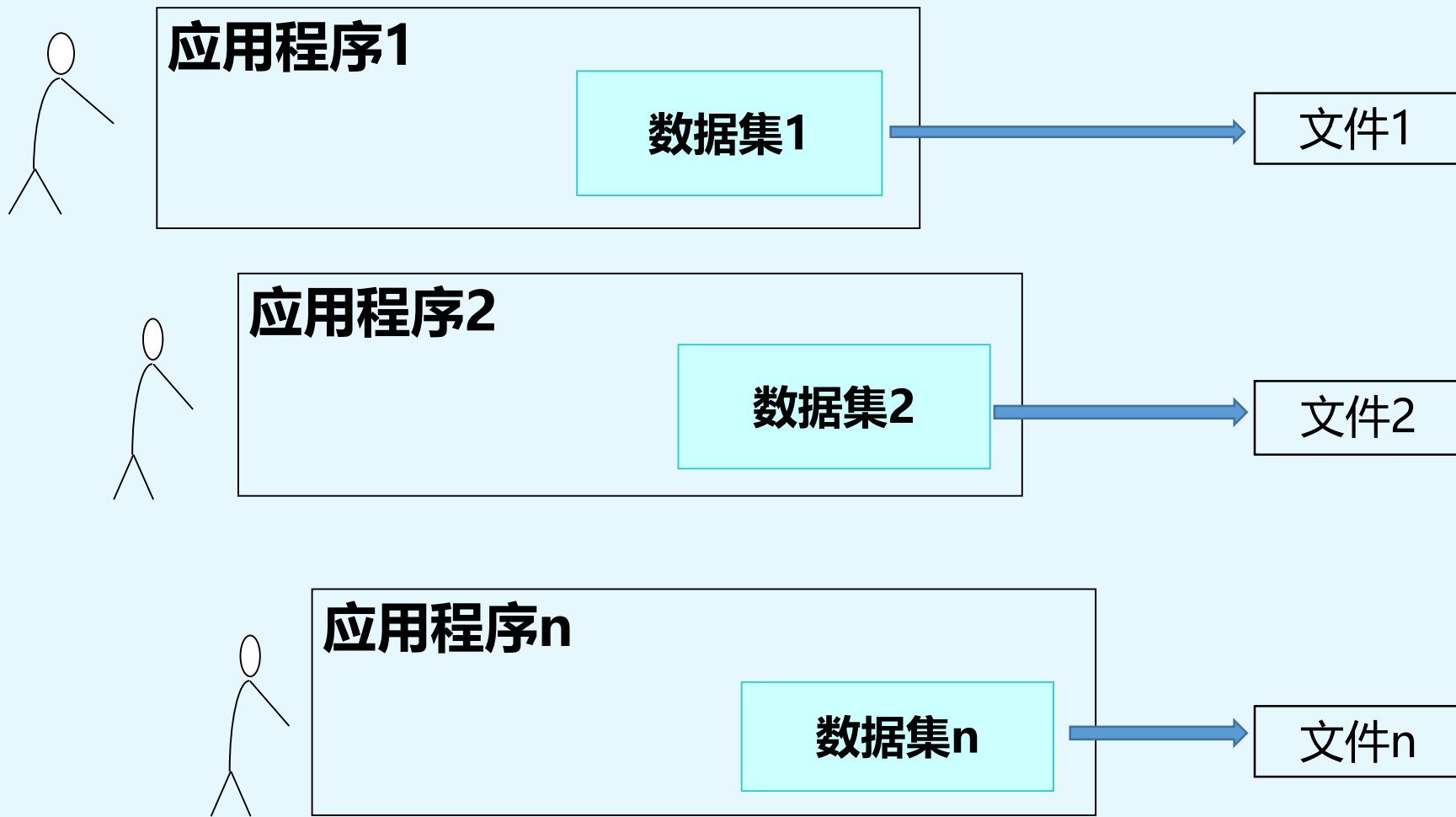


文件(F) 编辑(E)
格式(O) 查看(V)
帮助(H)

66
55
75
42
86
77
96
89
78
56



文件系统管理



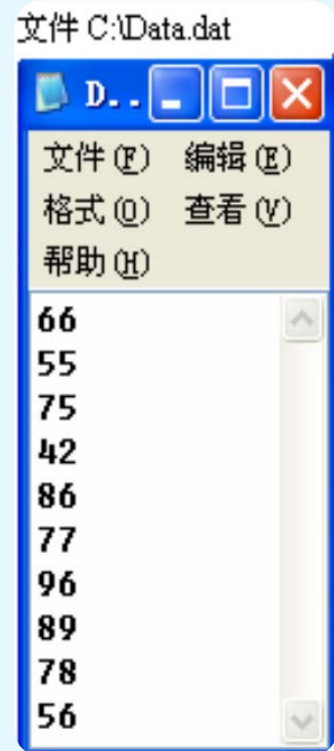


文件系统管理

例：用两个C语言程序，求10个数据之和和最大值

```
/*程序 3：求文件中 10 个数之和*/
#include <stdio.h>
main()
{
    int i,x,s=0;
    FILE*fp;
    /*打开文件*/
    fp=fopen( "c:\\data.dat" , "r" );
    for(i=0;i<10;i++)
        { /*文件中读数据*/
            fscanf(fp, "%d" ,&x);
            s=s+x;
        }
    printf ( "%d" ,s);
    fclose (fp); /*关闭文件*/
}
```

```
/* 程序 4：求文件中 10 个数中的最大值*/
#include <stdio.h>
main()
{
    int i,x,s=-32767;
    FILE*fp;
    /*打开文件*/
    fp=fopen( "c:\\data.dat" , "r" );
    for(i=0;i<10;i++)
        { /*文件中读数据*/
            fscanf(fp, "%d" ,&x);
            if ( s<x ) s=x;
        }
    printf( "%d" ,s);
    fclose(fp); /*关闭文件*/
}
```

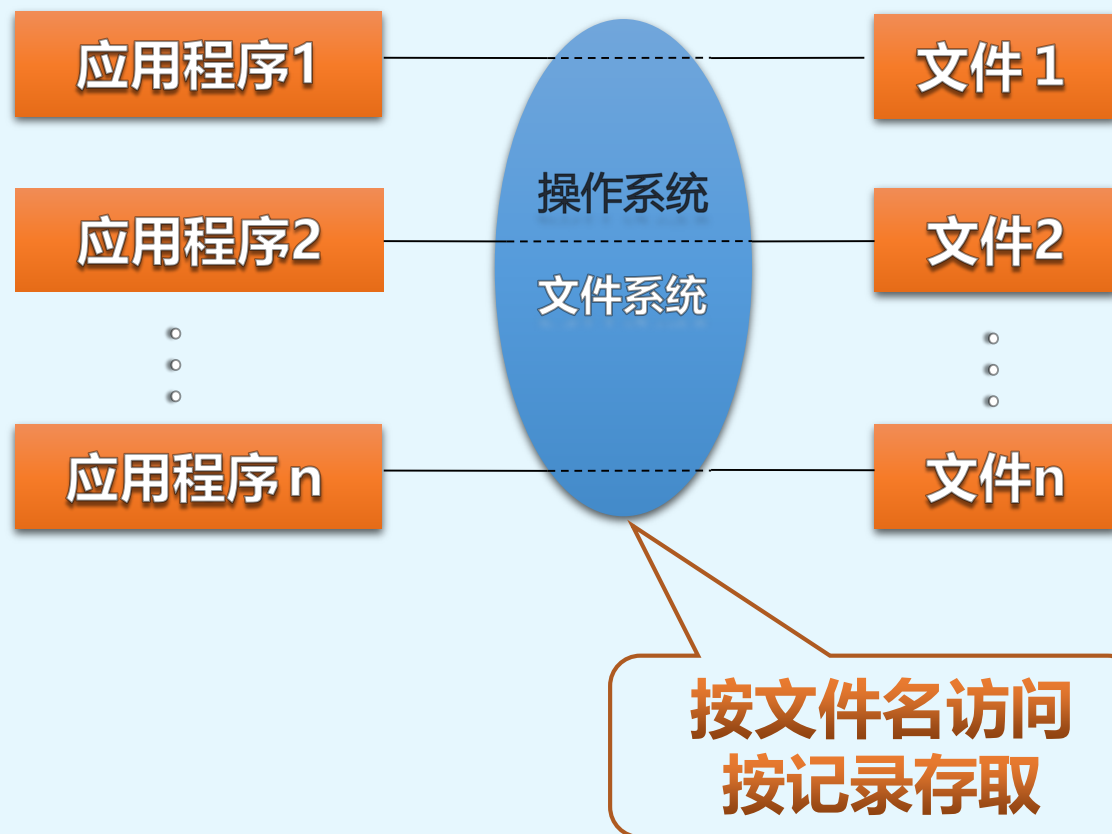




文件系统管理

数据管理特点:

- 由文件系统管理数据
- 数据可以长期保存
- 数据是面向应用的





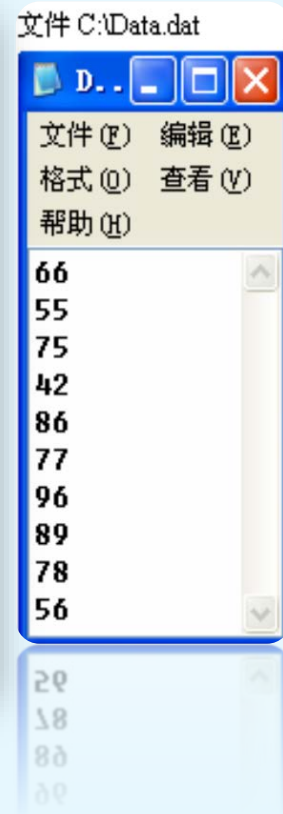
文件系统管理

数据管理特点:

- 数据共享性
具有了一定的共享性

```
/*程序 3: 求文件中 10 个数之和*/
#include <stdio.h>
main()
{
    int i,x,s=0;
    FILE *fp;
    /*打开文件*/
    fp=fopen( "c:\\data.dat", "r" );
    for(i=0;i<10;i++)
        { /*文件中读数据*/
            fscanf(fp, "%d", &x);
            s=s+x;
        }
    printf( "%d", s);
    fclose (fp); /*关闭文件*/
}
```

```
/* 程序 4: 求文件中 10 个数中的最大值*/
#include <stdio.h>
main()
{
    int i,x,s=-32767;
    FILE *fp;
    /*打开文件*/
    fp=fopen( "c:\\data.dat", "r" );
    for(i=0;i<10;i++)
        { /*文件中读数据*/
            fscanf(fp, "%d", &x);
            if ( s<x ) s=x;
        }
    printf( "%d", s);
    fclose(fp); /*关闭文件*/
}
```





文件系统管理

数据管理特点：

- **数据共享性**

具有了一定的共享性
存在冗余存储

学生文件S的记录结构

学号	姓名	性别	出生日期	所在系
----	----	----	------	-----

课程文件C的记录结构

课程编号	课程名	先修课程号	主讲教师
------	-----	-------	------

学生选课文件SC的记录结构

学号	姓名	课程编号	成绩
----	----	------	----

数据不一致



文件系统管理

数据管理特点:

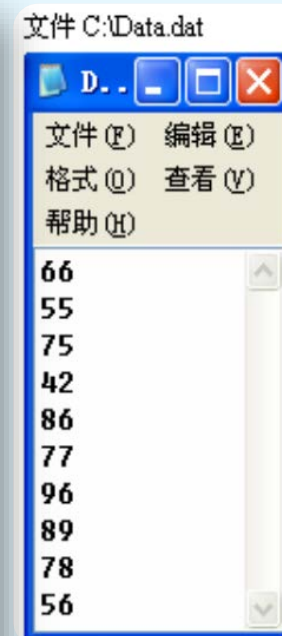
● 数据独立性

具有设备独立性

不具有数据独立性

```
/*程序 3: 求文件中 10 个数之和*/
#include <stdio.h>
main()
{
    int i,x,s=0;
    FILE *fp;
    /*打开文件*/
    fp=fopen( "c:\\data.dat" , "r" );
    for(i=0;i<10;i++)
        { /*文件中读数据*/
            fscanf(fp, "%d" ,&x);
            s=s+x;
        }
    printf( "%d" ,s);
    fclose (fp); /*关闭文件*/
}
```

```
/* 程序 4: 求文件中 10 个数中的最大值*/
#include <stdio.h>
main()
{
    int i,x,s=-32767;
    FILE *fp;
    /*打开文件*/
    fp=fopen( "c:\\data.dat" , "r" );
    for(i=0;i<10;i++)
        { /*文件中读数据*/
            fscanf(fp, "%d" ,&x);
            if ( s<x ) s=x;
        }
    printf( "%d" ,s);
    fclose(fp); /*关闭文件*/
}
```





数据库系统管理

- 20世纪60年代后期
- 管理的数据对象、应用范围、数据量、处理速度和共享性需求变化
- 硬件：大容量磁盘、光盘，价格下降
- 软件：价格上升，编制和维护成本增加
- 数据处理：要求更高的联机实时处理，出现分布处理





数据库系统管理

- 1963年，美国Honeywell公司的IDS (Integrated Data Store) 系统投入运行，揭开了数据库技术的序幕。
- 1965年，美国一家火箭公司利用该系统帮助设计了阿波罗登月宇航器，推动了数据库技术的产生。
- 1968年，美国IBM公司研发了基于层次模型的数据库系统IMS (Information Management System) 。
- 1969年，美国数据库系统语言研究会 (conference on data system language) 下属的数据库任务组 (DataBase Task Group) 提出基于网状数据模型的一个系统方案。
- 1970年，美国IBM的E.F.Codd发表论文提出了关系数据模型。