# 魔方影音一体化影院

14061019&14061052

**目录**

[魔方影音一体化影院 1](#_Toc468214527)

[一、需求概述 1](#_Toc468214528)

[1、系统概述 1](#_Toc468214529)

[2、主要功能 2](#_Toc468214530)

[3．系统安全性及完整性约束： 2](#_Toc468214531)

[二、数据流图 3](#_Toc468214532)

[1、顶层数据流图 3](#_Toc468214533)

[2、 用户层数据流图 3](#_Toc468214534)

[3、 生成相应订单层数据流图 6](#_Toc468214535)

[4、 资金状态层数据流图 7](#_Toc468214536)

[5、 个人信息维护数据流图 9](#_Toc468214537)

[6、 管理员0层数据流图 10](#_Toc468214538)

[7、 员工0层数据流图 11](#_Toc468214539)

[三、数据字典 12](#_Toc468214540)

[四、数据库概念模式设计 18](#_Toc468214541)

[1、系统局部ER图 18](#_Toc468214542)

[2、系统初步ER图 22](#_Toc468214543)

[3、系统基本ER图 24](#_Toc468214544)

[五、数据库模式分析与建表 25](#_Toc468214545)

[六、初步小结 32](#_Toc468214546)

[七、最终版本修改说明 32](#_Toc468214547)

## 一、需求概述

### 1、系统概述

本系统名为“魔方影音一体化影院”，目标是为实际电影院提供一个在线选票及信息管理的平台，方便电影院的日常运营及维护。基于用户来提供：在线查询电影的相关信息的查询（上映日期，主演，影片类型和影片内容），在线选座，支付订单，更改订单、退票等功能，来实现一个基于后端数据库系统的网页应用。

### 2、主要功能

对于用户来说：

浏览基础电影信息，提供对电影的名称搜索，按类型分类展示，普通展示并进行在线的选座购票，对已有的订单进行支付，和对订单进行更改与取消。用户可以在此系统上登录注册，更改个人信息和创建个人账户，并可以对个人账户进行充值。

对于管理员来说：

进行影片的排场，票房的管理，和人员信息的更改和安全维护(比如：普通人不能更改自己的账户余额)

对于员工来说：对用户的请求进行响应：比如用户订单的取消请求和对电影信息的基础数据输入。

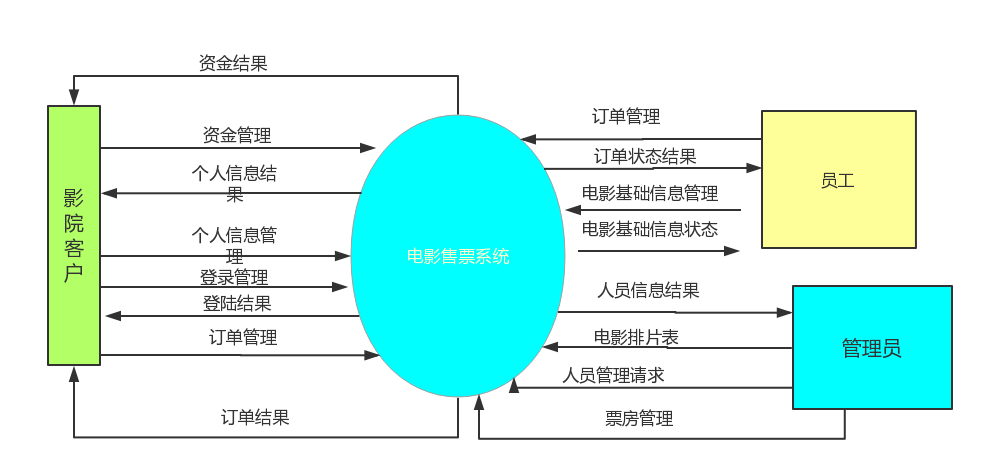
### 3．系统安全性及完整性约束：

用户不能更改超越自己权限的属性（比如：自己账户中的余额），这一点是通过：划分登录人群的管理权限来进行实现。

此外。系统在设计过程中按照实体完整性和参照完整性的约束逐步深入设计，满足完整性约束条件。

## 二、数据流图

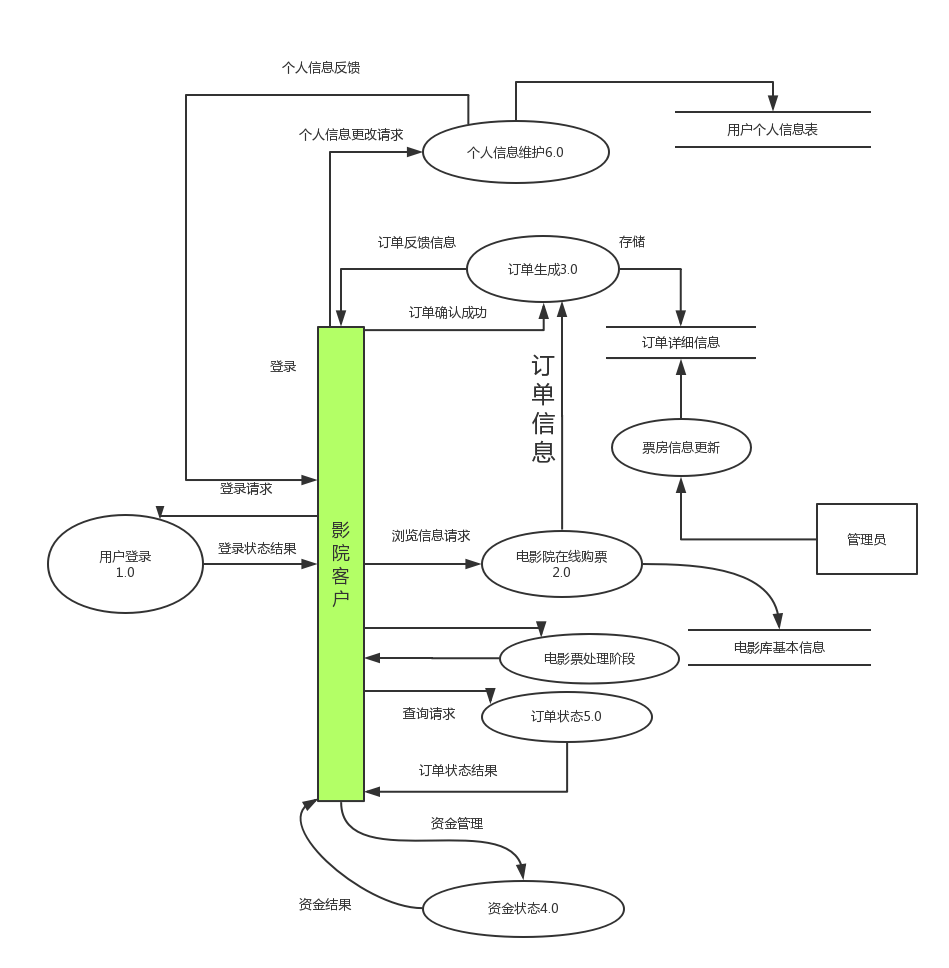
### 1、顶层数据流图



顶层数据流图（2.1）

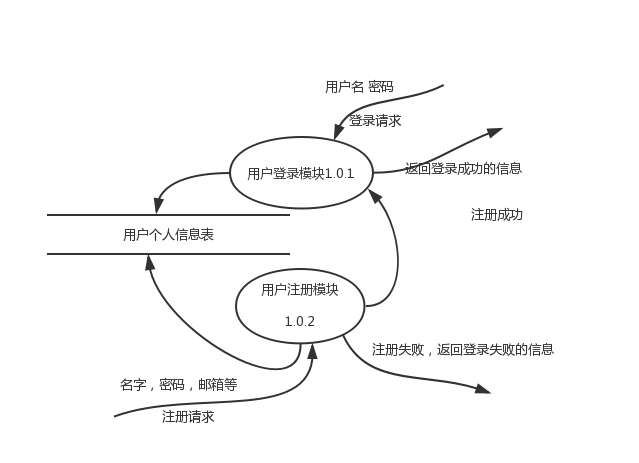
### 用户层数据流图

（1）用户0层数据流图



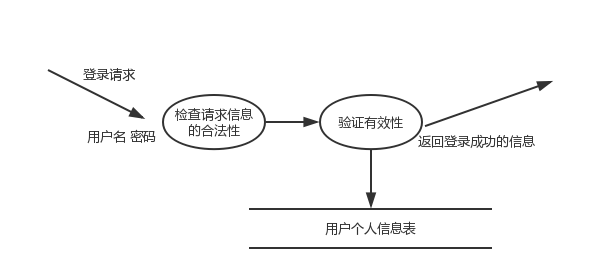
用户0层数据流图（2.2）

（2）用户登录1.0\_1层数据流图



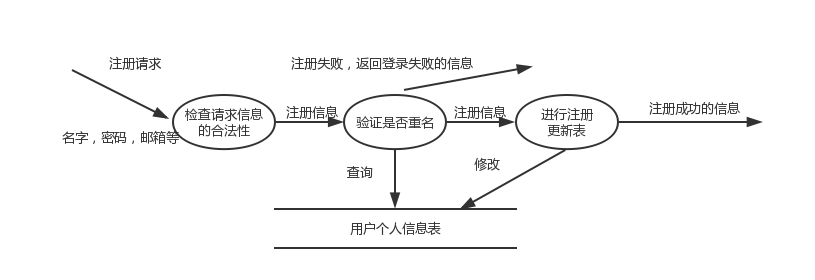
用户登录1.0\_1层数据流图（2.3）

（3）用户登录模块1.0.1\_2层数据流图



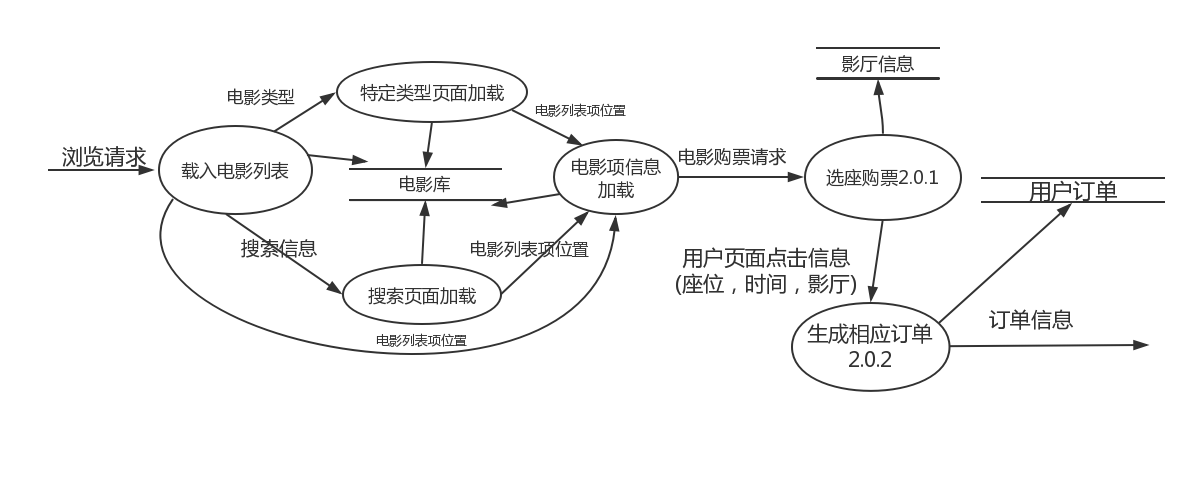
用户登录模块1.0.1\_2层数据流图（2.4）

（4）用户注册模块 1.0.2\_2层数据流图



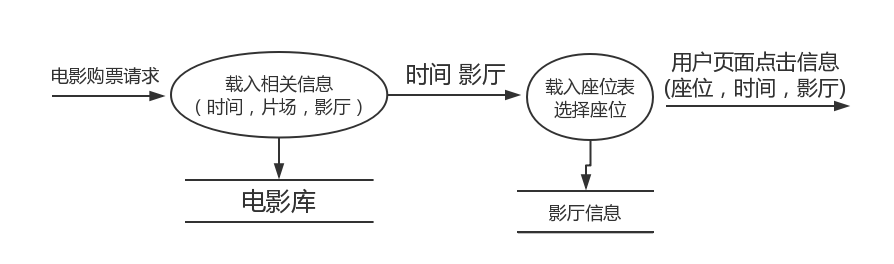
用户注册模块 1.0.2\_2层数据流图（2.5）

（5）电影院在线购票2.0\_1层数据流图



电影院在线购票2.0\_1层数据流图（2.6）

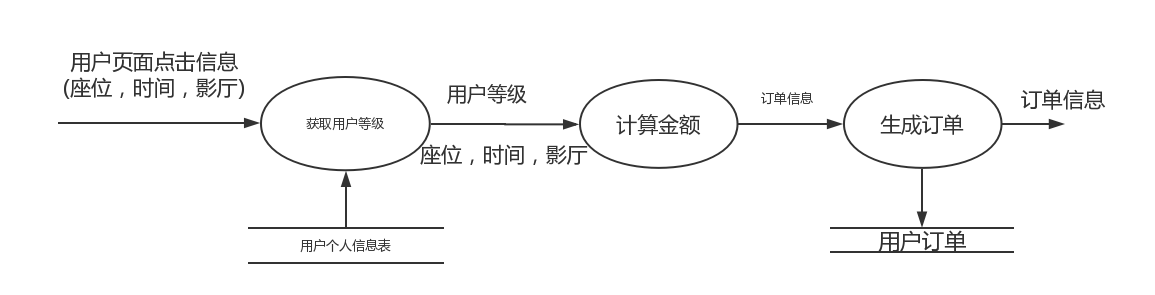
（6）选座购票2.0.1\_2层数据流图



选座购票2.0.1\_2层数据流图（2.7）

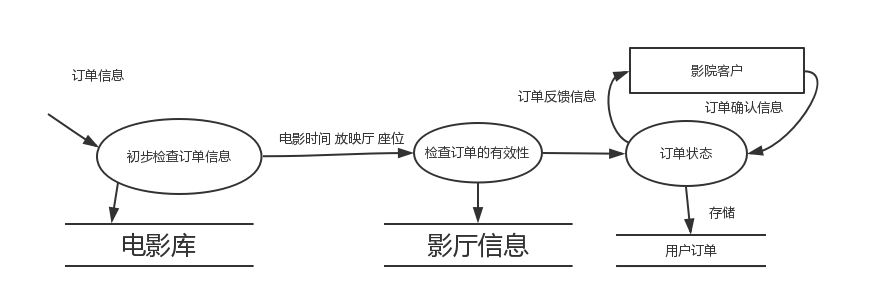
### 生成相应订单层数据流图

（1）生成相应订单 2.0.2\_2层数据流图



生成相应订单 2.0.2\_2层数据流图（2.8）

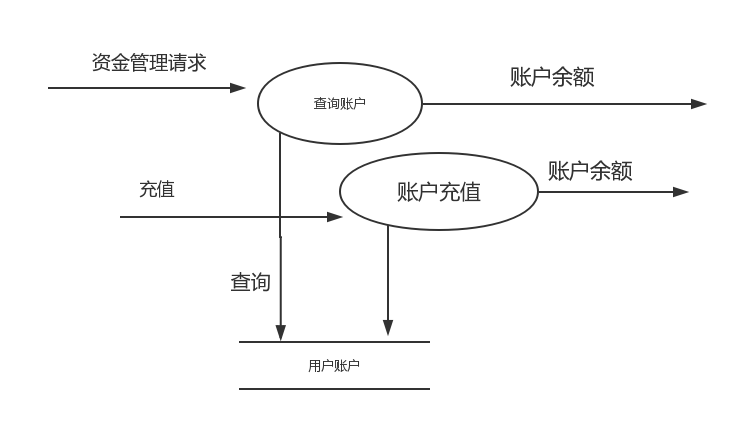
(2) 订单生成3.0\_1层数据流图



订单生成3.0\_1层数据流图（2.9）

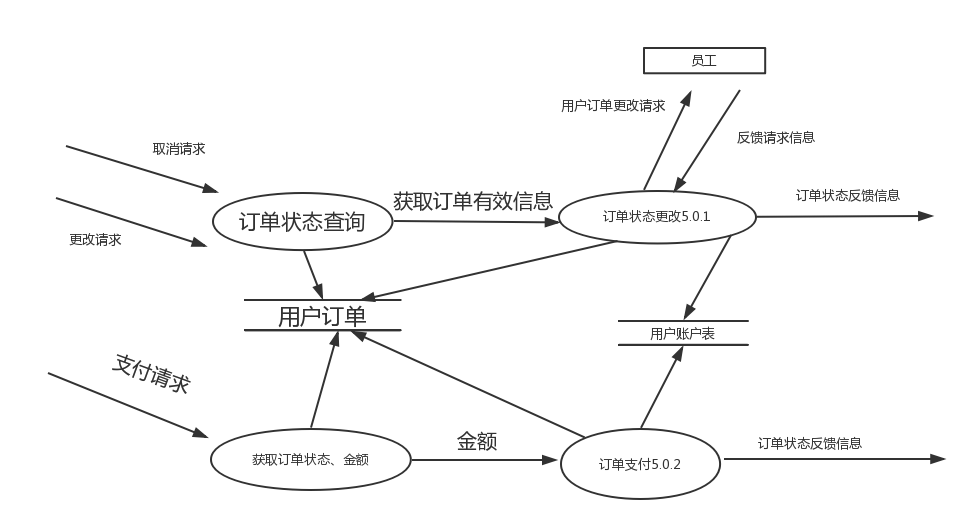
### 资金状态层数据流图

（1）资金状态4.0\_1层数据流图



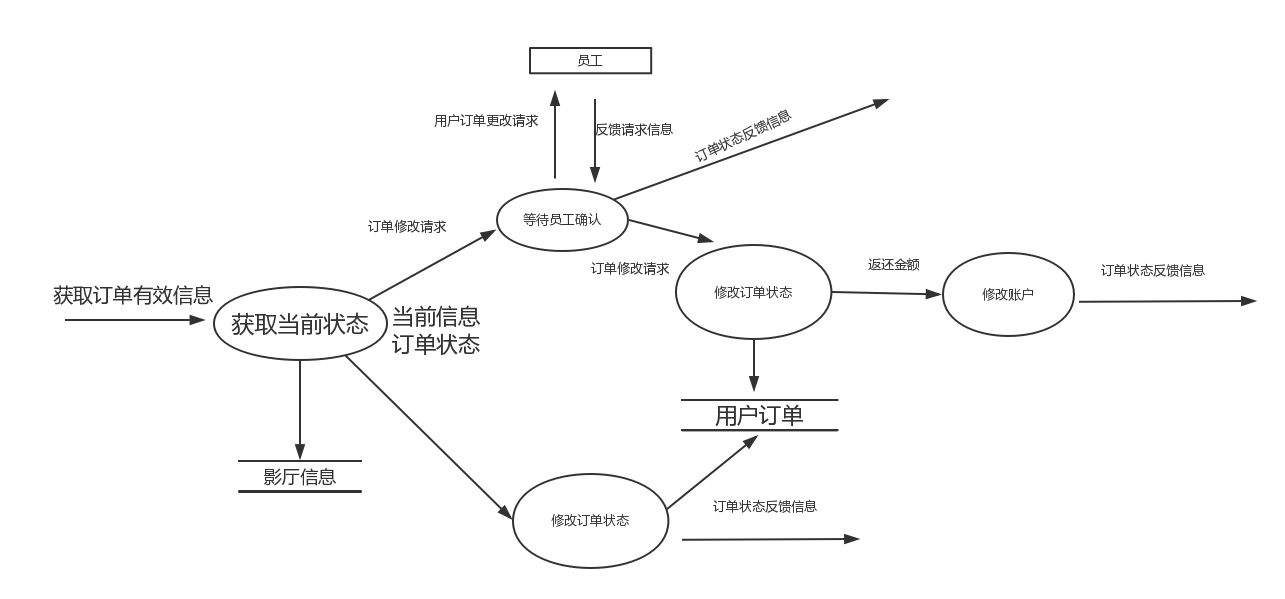
资金状态4.0\_1层数据流图（2.10）

（2）订单状态5.0\_1层数据流图



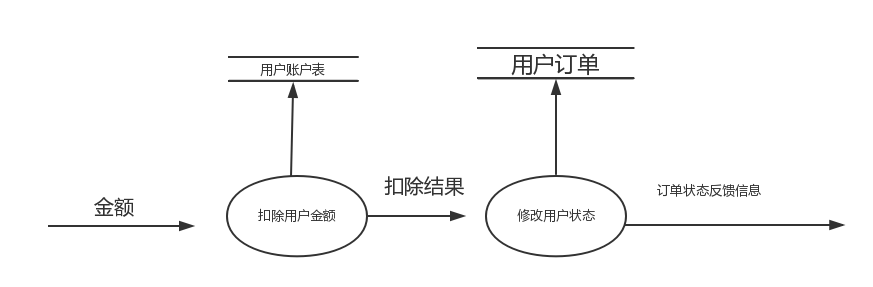
订单状态5.0\_1层数据流图（2.11）

（3）订单状态更改5.0.1\_2层数据流图



订单状态更改5.0.1\_2层数据流图（2.12）

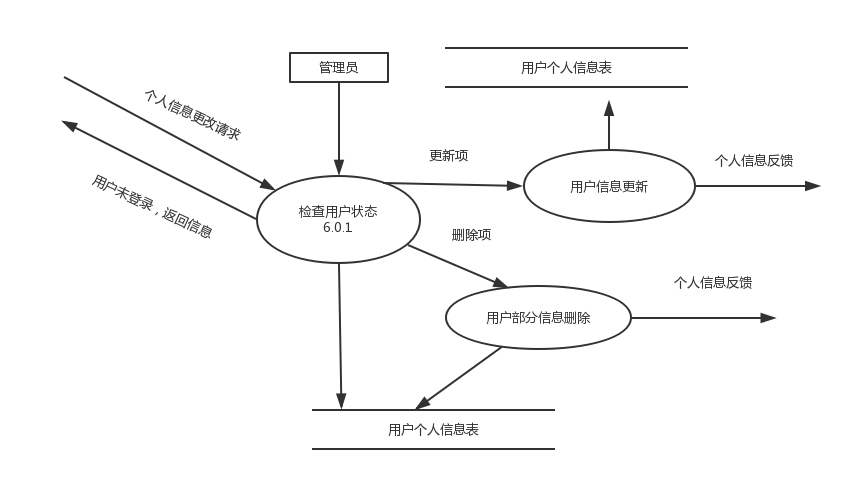
（4）订单状态更改5.0.2\_2层数据流图



订单状态更改5.0.2\_2层数据流图（2.13）

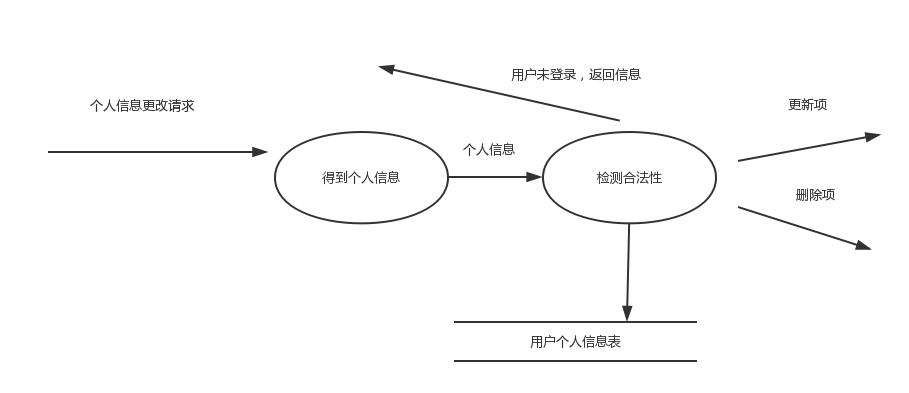
### 个人信息维护数据流图

（1）、个人信息维护6.0\_1层数据流图



个人信息维护6.0\_1层数据流图（2.14）

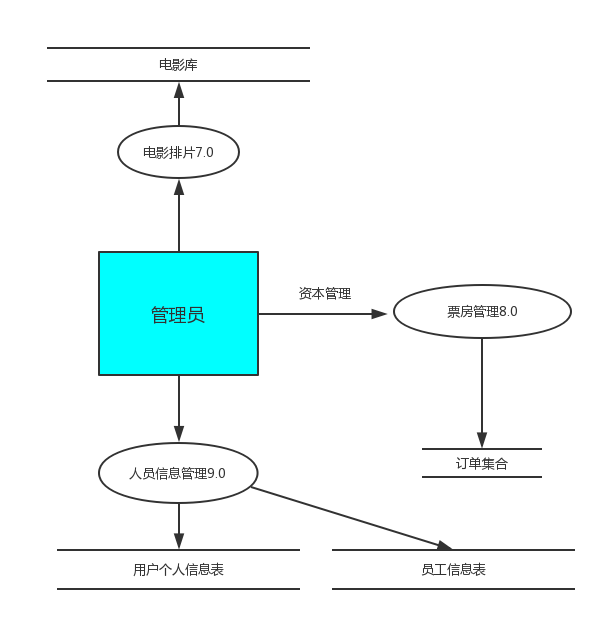
（2）、检查用户状态 6.0.1\_2层数据流图



检查用户状态 6.0.1\_2层数据流图（2.15）

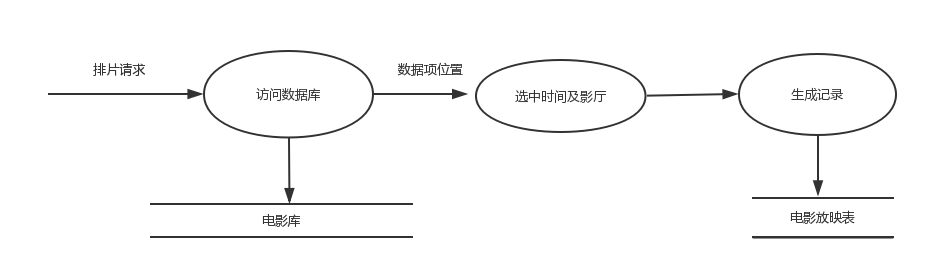
### 管理员0层数据流图

（1）、管理员0层数据流图



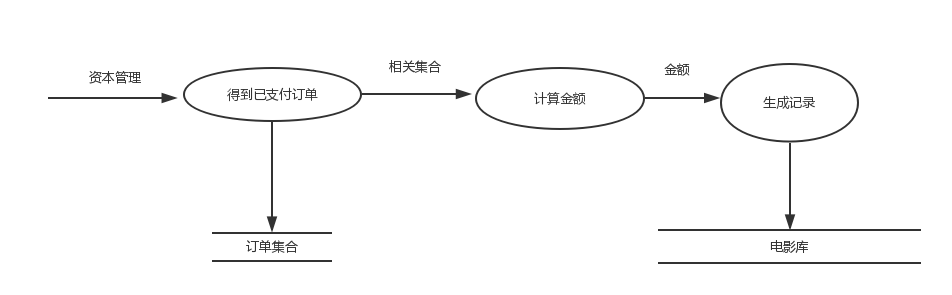
管理员0层数据流图（2.16）

（2）、电影排片7.0\_1层数据流图



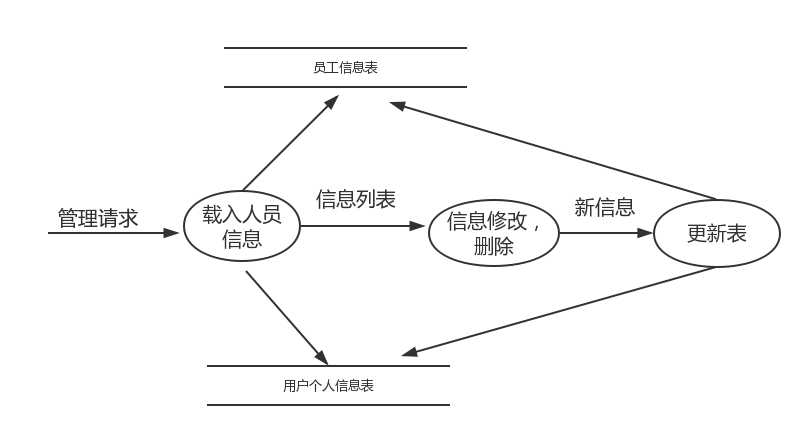
电影排片7.0\_1层数据流图（2.17）

（3）、票房管理8.0\_1层数据流图



票房管理8.0\_1层数据流图（2.18）

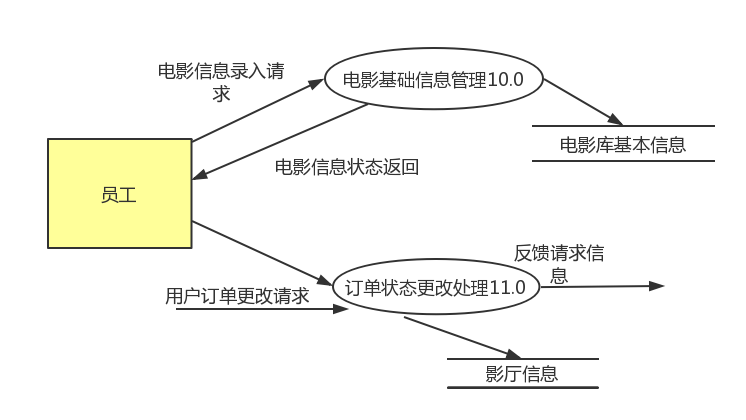
（4）、人员信息管理9.0\_1层数据流图



人员信息管理9.0\_1层数据流图（2.19）

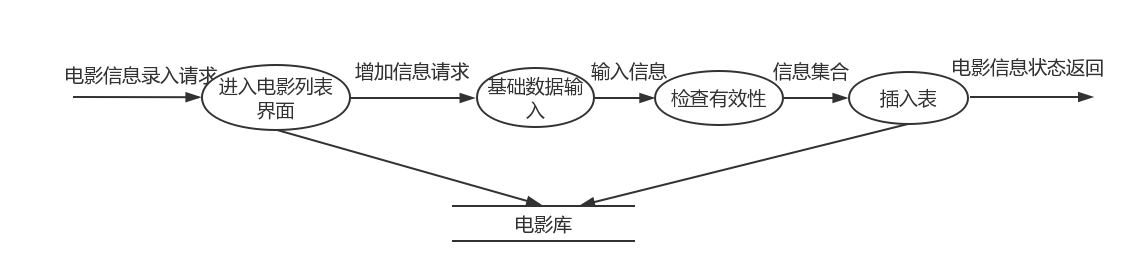
### 员工0层数据流图

（1）、员工0层数据流图



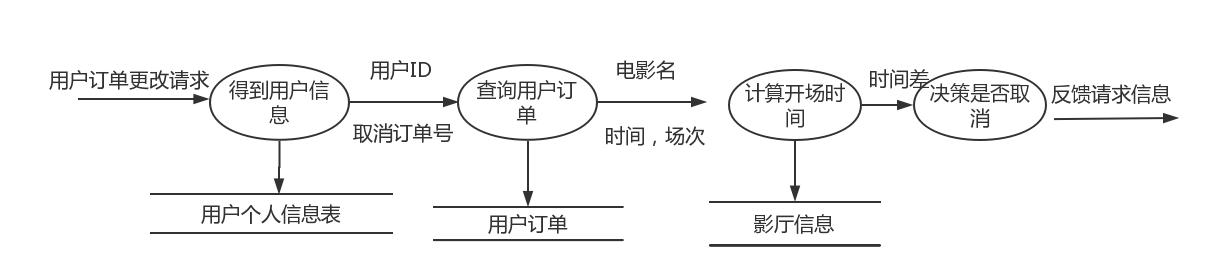
员工0层数据流图（2.20）

（2、人员信息管理10.0\_1层数据流图



人员信息管理10.0\_1层数据流图（2.21）

（3、）人员信息管理11.0\_1层数据流图



人员信息管理11.0\_1层数据流图（2.22）

## 三、数据字典

表（3.1.1）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：user | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| user\_id | 01 | int | 4B | 0 |  | 不相同 | 不允许 | 用户id |
| user\_name | 02 | varchar | 20B | 0 |  |  | 不允许 | 用户名 |
| register\_time | 03 | date  time | 8B | 0 | 秒 |  | 不允许 | 注册时间 |
| password | 04 | varchar | 30B | 0 |  |  | 不允许 | 密码 |
| email | 05 | varchar | 30B | 0 |  | 不相同  符合格式 | 不允许 | 电子邮箱 |
| grade | 06 | char | 2B | 0 |  | A，B，C，D其中一个 | 不允许 | 等级 |
| cash | 07 | float | 4B | 2 | 元 | 大于等于0 | 不允许 | 零钱 |

表（3.1.2）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：orders | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| oder\_time | 01 | date  time | 8B | 0 | 秒 |  | 不允许 | 下单时间 |
| get\_ticket\_key | 02 | char | 12B | 0 |  | 不相同 | 不允许 | 取票码 |
| tiket\_state | 03 | tinyint | 1B | 0 |  |  | 不允许 | 取票状态 |

表（3.1.3）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：discount | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| discount\_id | 01 | int | 4B | 0 |  | 不相同 | 不允许 | 折扣号 |
| discount\_name | 02 | varchar | 30B | 0 |  |  | 不允许 | 折扣名 |
| discount\_rate | 03 | float | 4B | 1 |  | 大于等于0小于等于1 | 不允许 | 折扣率 |

表（3.1.4）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：ticket | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| ticket\_id | 01 | int | 4B | 0 |  | 不相同 | 不允许 | 票号 |
| movie\_id | 02 | int | 4B | 0 |  |  | 不允许 | 电影号 |
| seat\_id | 03 | int | 4B | 0 |  | 不相同 | 不允许 | 座位号 |
| price | 04 | float | 4B | 0 | 元 | 大于等于0 | 不允许 | 票价 |
| sold\_state | 05 | tinyint | 1B | 0 |  | 不相同  符合格式 | 不允许 | 销售状态 |

表（3.1.5）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：movie | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| m\_id | 01 | int | 4B | 0 |  | 不相同 | 不允许 | 电影号 |
| m\_name | 02 | varchar | 30B | 0 |  |  | 不允许 | 电影名 |
| director | 03 | varchar | 30B | 0 |  |  | 允许 | 导演 |
| origin | 04 | varchar | 20B | 0 |  |  | 允许 | 产地 |
| actors | 05 | varchar | 40B | 0 |  |  | 允许 | 主角 |
| type | 06 | varchar | 20B | 0 |  |  | 允许 | 类型 |
| language | 07 | varchar | 10B | 0 |  |  | 允许 | 语言 |
| D\_  type | 08 | char | 1B | 0 |  | ‘2’或’3’ | 允许 | 2D或3D |
| release\_time | 09 | datetime | 8B | 0 | 秒 |  | 允许 | 上映时间 |
| score | 10 | float | 4B | 1 |  | >=0 &&  <=10 | 允许 | 豆瓣评分 |
| time | 11 | smallint | 2B | 0 | 分 | >=0 | 允许 | 片长 |
| introduction | 12 | text | 6000B | 0 |  |  | 允许 | 简介 |
| img\_  path | 13 | varchar | 20B | 0 |  |  | 允许 | 图片路径 |
| box\_  office | 14 | double | 4B | 2 | 元 | >=0 | 允许 | 票房 |

表（3.1.6）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：play | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| start\_  time | 01 | date  time | 8B | 0 | 秒 | 不相同 | 不允许 | 电影开始放映时间 |

表（3.1.7）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：admin | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| adm\_id | 01 | int | 4B | 0 |  | 不相同 | 不允许 | 管理员id |
| adm\_password | 02 | varchar | 20B | 0 |  |  | 不允许 | 管理员密码 |

表（3.1.8）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：seat | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| seat\_id | 01 | int | 4B | 0 |  | 不相同 | 不允许 | 座位号 |
| row | 02 | smallint | 2B | 0 |  |  | 不允许 | 座位排号 |
| col | 03 | smallint | 2B | 0 |  |  | 不允许 | 座位列号 |

表（3.1.9）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：seat\_mhall | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| seat\_id | 01 | int | 4B | 0 |  | 不相同 | 不允许 | 座位号 |
| mhall\_  id | 02 | int | 4B | 0 |  |  | 不允许 | 影厅号 |

表（3.1.10）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：mhall | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| mhall\_  id | 01 | int | 4B | 0 |  |  | 不允许 | 影厅号 |
| location | 02 | varchar | 20B | 0 |  |  | 不允许 | 所在地 |
| mh\_  type | 03 | varchar | 10B | 0 |  |  | 允许 | 影厅类型 |

表（3.1.11）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：work | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| start\_  time | 01 | date  time | 8B | 0 | 秒 |  | 不允许 | 工作开始时间 |
| end\_  time | 02 | date  time | 8B | 0 | 秒 |  | 不允许 | 工作结束时间 |

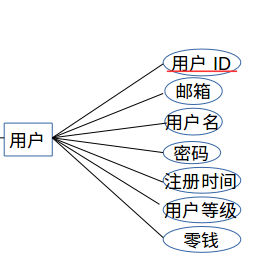
表（3.1.12）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据组名：worker | | | | | | | | |
| 数据  项名 | 编号 | 数据类型 | 数据宽度 | 小数位数 | 单位 | 值约束 | 允许空值 | 描述 |
| worker\_id | 01 | int | 4B | 0 |  |  | 不允许 | 工号 |
| position | 02 | varchar | 20B | 0 |  |  | 不允许 | 职位 |

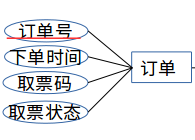
## 四、数据库概念模式设计

### 1、系统局部ER图

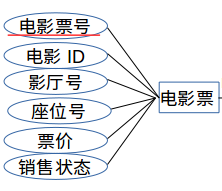
#### （1）各个实体ER图



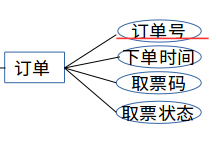
用户属性图（4.1.1）



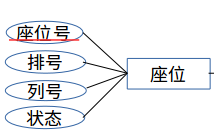
订单属性图（4.1.2）



电影票属性图（4.1.3）



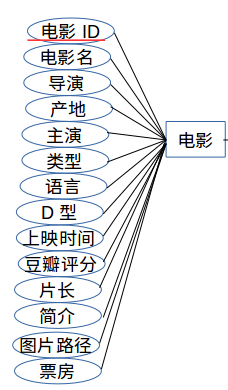
订单属性图（4.1.4）



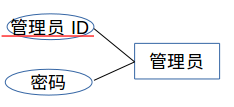
座位属性图（4.1.5）



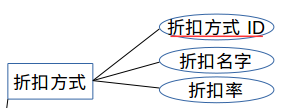
影厅属性图（4.1.6）



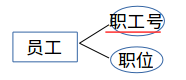
电影属性图（4.1.7）



管理员属性图（4.1.8）

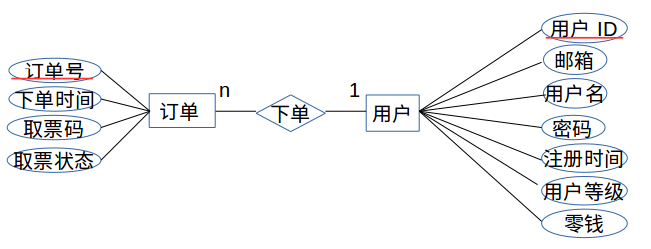


折扣方式属性图（4.1.9）

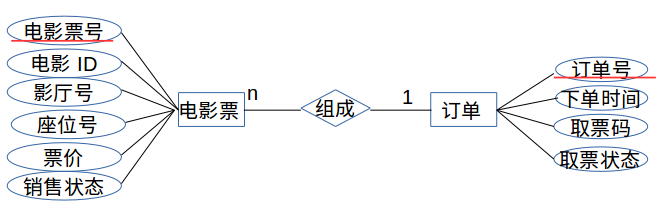


职工属性图（4.1.10）

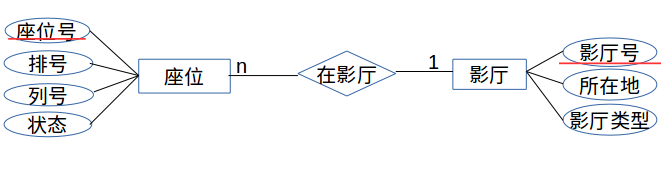
#### （2）局部ER图



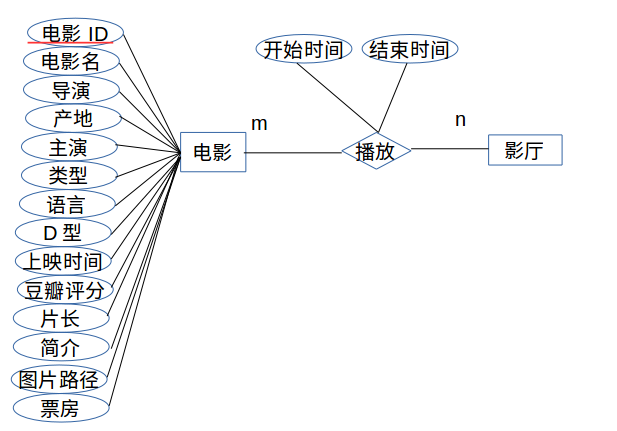
订单生成局部ER图（4.1.11）



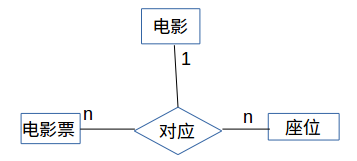
用户下单局部ER图（4.1.12）



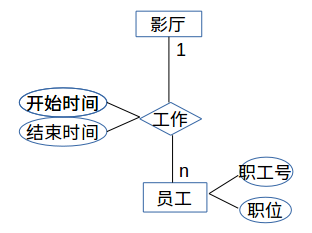
座位-影厅局部ER图（4.1.13）



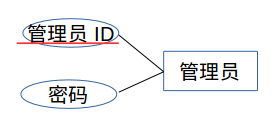
电影播放局部ER图（4.1.14）



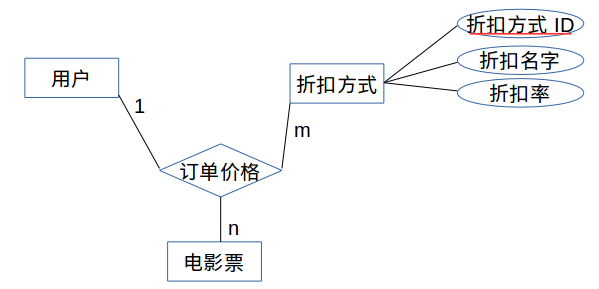
电影票-电影-座位局部ER图（4.1.15）



员工影厅局部ER图（4.1.16）



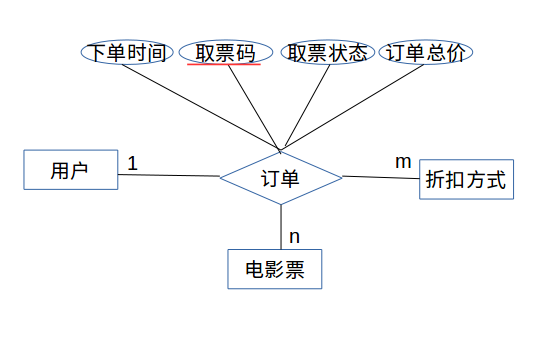
管理员局部ER图（4.1.17）



订单价格ER图（4.1.18）

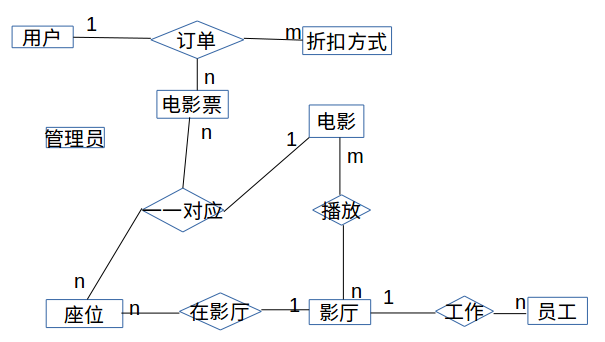
### 2、系统初步ER图

首先考虑订单部分，由图(4.1.18)可以看到订单的价格是由用户、折扣方式以及电影票共同产生的。而图(4.1.11)和图(4.1.12)中订单又与用户和电影票相关，所以为解决冲突，这里把订单改为关系，原来作为订单实体的属性改为关系属性。合并成下面的ER图（4.2.1）。



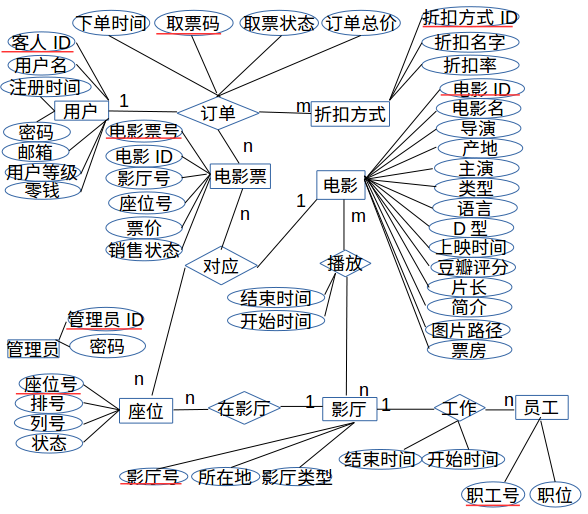
（4.2.1）

抛开属性，可以把以上局部ER图合并成如下图4.2.2



（4.2.2）

把属性都添上，得到初步ER图如下（图4.2.3）。



初步ER图（4.2.3）

### 3、系统基本ER图

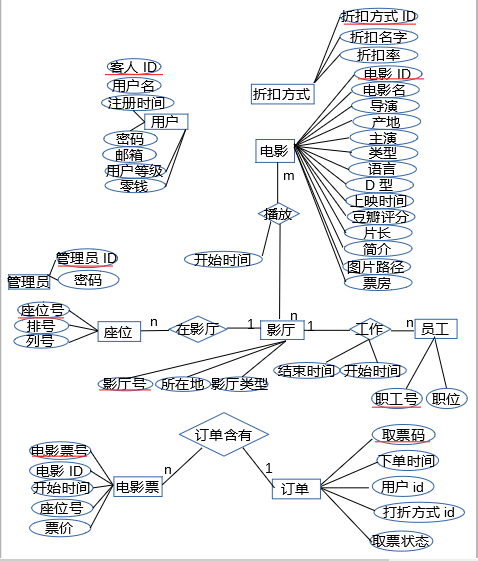
由以上初步ER图可以发现，单凭电影票的属性就可以确定电影和座位，所以电影票、电影和座位之间的对应关系是多除的，可以删除。

开始电影和影厅之间的播放关系中，结束时间可以由开始时间加电影片长确定，所以结束时间可以删除。

座位与电影票是一一对应的，所以座位的状态可以用电影票的销售状态代替，这个属性可以删除。

电影票的影厅号可以由座位与影厅的关系推出，所以这个属性可以去掉。

订单总价可以通过票价用户和折扣方式推出，可以去掉该属性。从初步ER图中去掉以上所说，可以得到基本ER图如下图4.2.4。



（4.2.4）

## 五、数据库模式分析与建表

我们图2.13的实体及其之间的关系进行分析。

1、用户表：(含义与数据字典保持一致)

user(user\_id，user\_name, register\_time, password, email, level, cash)

主属性：user\_id

非主属性：user\_name, register\_time, password, email, level, cash

函数依赖：

user\_id user\_name

user\_id register\_time

user\_id password

user\_id email

user\_id level

user\_id cash

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF

2、订单关系(含义与数据字典保持一致)

order(get\_ticket\_key, discoun\_id, ticket\_id, oder\_time, user\_id, tiket\_state)

主属性:get\_ticket\_key

非主属性：discoun\_id, ticket\_id, oder\_time, user\_id, tiket\_state

函数依赖：

get\_ticket\_key discoun\_id

get\_ticket\_key ticket\_id

get\_ticket\_key oder\_time

get\_ticket\_key user\_id

get\_ticket\_key tiket\_state

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

3、折扣 (含义与数据字典保持一致)

discount（discount\_id，discount\_name，discount\_rate）

主属性：discount\_id

非主属性：discount\_name，discount\_rate

函数依赖：

discount\_id discount\_name

discount\_id discount\_rate

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

4、电影票ticket (含义与数据字典保持一致)

ticket（ticket\_id，movie\_id，seat\_id，price，sold\_state）

ticket\_id是在用户购买电影票时随机生成的。

主属性：ticket\_id ,用来唯一标识一张电影票

非主属性：movie\_id，seat\_id，price，sold\_state

函数依赖：

ticket\_id movie\_id

ticket\_id seat\_id

ticket\_id price

ticket\_id sold\_state

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

5、movie(含义与数据字典保持一致)

movie(m\_id,m\_name,director,origin,actors,language,D\_type,release\_t ime, score, time, introduction, img\_path, box\_office)

主属性：m\_id

非主属性：m\_name, director, origin, actors, language, D\_type,release\_time, score, time, introduction, img\_path, box\_office

函数依赖：

m\_id m\_name

m\_id director

m\_id origin

m\_id actors

m\_id language

m\_id D\_type

m\_id release\_time

m\_id score

m\_idtime

m\_id introduction

m\_id img\_path

m\_idbox\_office

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

6、播放排片play(含义与数据字典保持一致)

主属性：m\_id， mhall\_id ，start\_time

非主属性：无

函数依赖：

m\_id，mhall\_id start\_time

同一个电影在不同的影厅有不同的播放时间，所以：电影ID 和影厅ID和播放时间构成一个元组。

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

7、座位 seat (含义与数据字典保持一致)

seat（seat\_id, row, col）

主属性：seat\_id

非主属性：row, col

函数依赖：

seat\_id row

seat\_id col

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

8、座位-影厅关系seat\_mhall (含义与数据字典保持一致)

seat\_mhall(seat\_id，mhall\_id)

主属性：seat\_id，mhall\_id

非主属性：无

函数依赖：

seat\_id，mhall\_id 共同来确定一个函数关系

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

9、影厅mhall (含义与数据字典保持一致)

mhall（mhall\_id, location, mh\_type）

主属性：mhall\_id

非主属性：location, mh\_type

函数依赖：

mhall\_id location

mhall\_id mh\_type

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

10、员工worker (含义与数据字典保持一致)

worker（worker\_id, position）

主属性：worker\_id

非主属性：position

函数依赖：

worker\_id position

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

11、工作关系work\_relation(含义与数据字典保持一致)

work\_relation(mhall\_id, worker\_id, start\_time, end\_time)

主属性：mhall\_id, worker\_id

非主属性：start\_time, end\_time

函数依赖：

由于一个员工在同一个影厅的不同时间段需要上班，所以上述关系不满足：码唯一确定一个元组，在插入时会产生异常，改写为：

work\_relation(mhall\_id, worker\_id, start\_time, end\_time)

此时：

主属性：mhall\_id, worker\_id，start\_time, end\_time

非主属性：无

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

12、管理员(含义与数据字典保持一致)

主属性：adm\_id

非主属性：adm\_password

函数依赖：

adm\_id adm\_password

结论:该关系在满足属性原子性的基础上，非主属性不存在对码的传递函数依赖和部分函数依赖，属于3NF。

## 六、初步小结

在设计的过程中：按照相关规范来进行约束，部分优化在实现过程中会处理，可能还有不尽完善的地方，请老师指出。

## 七、最终版本修改说明

一开始订单和电影票的关系没弄清楚，后来在结过思考发现，订单表里是不能直接放电影票信息的，因为我们不知道用户到底买了几张票，所以后来单开一个订单和电影票的关系表来记录一个订单中到底包含了哪些电影票。

再者是电影票的属性，一开始没注意到如果不加开场时间的话，是不能通过电影票的信息求出这个票的电影是什么时候开始的。所以把开始时间加入了电影票的属性中。最后是电影票中的销售状态，这个是不必要有的，我们在网站开发时如果要显示哪个变位被占了，只需要直接查找看看有没有电影票本身就行了。