
毕业设计（论文）

课题名称：中小型超市管理系统规划

专业名称 计算机信息管理

班 级 XXX

指导教师 XXX

学生姓名 XXXX

完成时间：XXXX年 XX 月 XX 日至 XX 年 XXX 月 XX 日

初稿时间： XXXX 年 XX 月 XX 日

定稿时间： XXXX 年 XX 月 XXXXXX日

目 录

<u>一 需求分析</u>	
<u>(一) 行业概念的理解</u>	1
<u>(二) 业务流程说明</u>	2
<u>(三) 问题解决方案</u>	2
<u>(四) 数据流图</u>	3
<u>二 系统设计</u>	
<u>二 系统设计</u>	
<u>(一) 系统模块的划分</u>	4
<u>(二) 数据字典</u>	4
<u>三、程序的设计与编码</u>	
<u>(一) POS 系统的开发</u>	8
<u>1. 功能及实现方法</u>	8
<u>2. 问题讨论：交易流水号的确定</u>	9
<u>(二) 管理信息系统的设计</u>	9
<u>1. 界面设计</u>	9
<u>2. 各个功能模块实现的要点和技术</u>	10
<u>2.1 类继承与重载</u>	10
<u>2.2 windows 消息机制</u>	11
<u>2.3 关于权限设置</u>	12
<u>2.4 应用 Delphi 显式事务控制</u>	13
<u>2.5 存储过程的使用</u>	13
<u>参考文献</u>	

中小型超市管理系统规划

内容摘要：《中小型超市管理信息系统》是一个比较复杂和全面的系统，可以完成中小型超市的大部分业务。总的来说包括进货管理，物价管理，库存管理，销售管理，供货厂商管理，POS收款机子系统等。该系统界面朴素一致，但不失美观大方；操作方便，查询功能强大；数据全面、准确。系统采用了目前比较流行而技术非常成熟地客户机/服务器结构(C/S)，前台应用程序用 Delphi 开发,后台数据库是 Microsoft SQL Server 2000 。

关键字：面向对象的程序设计，继承，WINDOWS 消息，存储过程，事务

一 需求分析

(一) 行业概念的理解

POS Point Of Sells, 电子收款机系统, 也就是直接与顾客打交道的收银机系统。

定单状态: 用来描述商品进货方式, 一般分为定单、日配、自采等。

定单: 表示该商品进货必须先向供货商发出定单, 供货商按定单上的商品明细发货, 超市则按定单上的商品明细收货。

日配: 指那些每天都要进货的商品或者补货期较短的商品, 例如水果, 生鲜蔬菜等。如果按定单, 时间不允许而且工作重复烦琐。

自采: 顾名思义, 超市自己负责商品的进货。

销售状态: 用来描述某种商品当前的销售状态, 一般定义为正常、促销、折扣等, 正常表示商品处于正常的售价状态, 即按正常的售价销售。

促销: 表示超市把某种商品让利销售, 以此来吸引跟多的消费者。一般低于正常售价甚至成本价。

折扣: 表示某种商品销售情况不好, 或者商品有瑕疵, 或者保质期快到等情况发生时, 使尽快商品销售出去的一种手段。

库存调整: 当非因正常销售发生而实物数量变动时, 通过调整库存使其帐面数量与实物数量相符。

日处理: 当一天的销售结束后, 开始做日处理。把当天的销售数据汇总, 调整商品库存。

损益: 商品出现损坏, 变质, 被盗等情况发生时, 商品发生损益。损益通过调整库存得到体现。

商品变价: 商品变价与促销/折扣调整价格是两个不同的概念。商品变价是指修改商品的正常售价, 调整后在一定的时间内不会在发生改变。价格可能是升高也可能是降低, 一般都是由于进价变动引起的。

促销/折扣期: 每一种商品促销/折扣都有一定的期限, 即有一定的时间限制。一种商品不可能无期限的促销/折扣。当期限超过, 商品自动恢复正常售价。

毛利、毛利百分比: $\text{毛利} = \text{商品售价} - \text{商品进价}$ $\text{毛利百分比} = (\text{商品售价} - \text{商品进价}) \div \text{商品售价}$

商品进价及成本核算: 在成本核算中, 商品成本价的确定一般有最后进价, 移动加权平均, 按商品批次定价等几种方式, 由于后两种核算方式较复杂, 本系统采用简单的最后进价方式。

(二) 业务流程说明

在实际管理中, 有三种信息流动, 即资金流, 单据流和商品实物流。系统围绕实物流展开, 对商品从供应商那里进货到商品销售到顾客手中整个流程进行管理。简单来说,

就是商品的进销存管理。每个环节通过相应的单据来连接，同时资金发生流动。在商品管理方面，我们提出部门和商品分类的概念，这种分级管理方式简洁高效。所谓部门，就是按照商品的基本属性进行分类管理。例如，超市一般可分为食品部，生鲜部，日用品部等。部门下面再分类，所谓商品分类，指按商品的具体属性划分，例如，食品部下面可以有糖果饼干类，饮料类等，每个部门和每个商品类别都有相应的负责人负责。《中小型超市管理信息系统》完全体现了这种管理思想，商品的销售和商品库存都可以按部门级和分类级完成查询等其他操作。作为一个超市的管理人员，及时了解销售情况是非常必要的，以便及时调整进货或者安排营销策略，把握商机。而《中小型超市管理信息系统》正好满足了这一要求，可以实时查询前台的销售情况，并且数据准确可信。

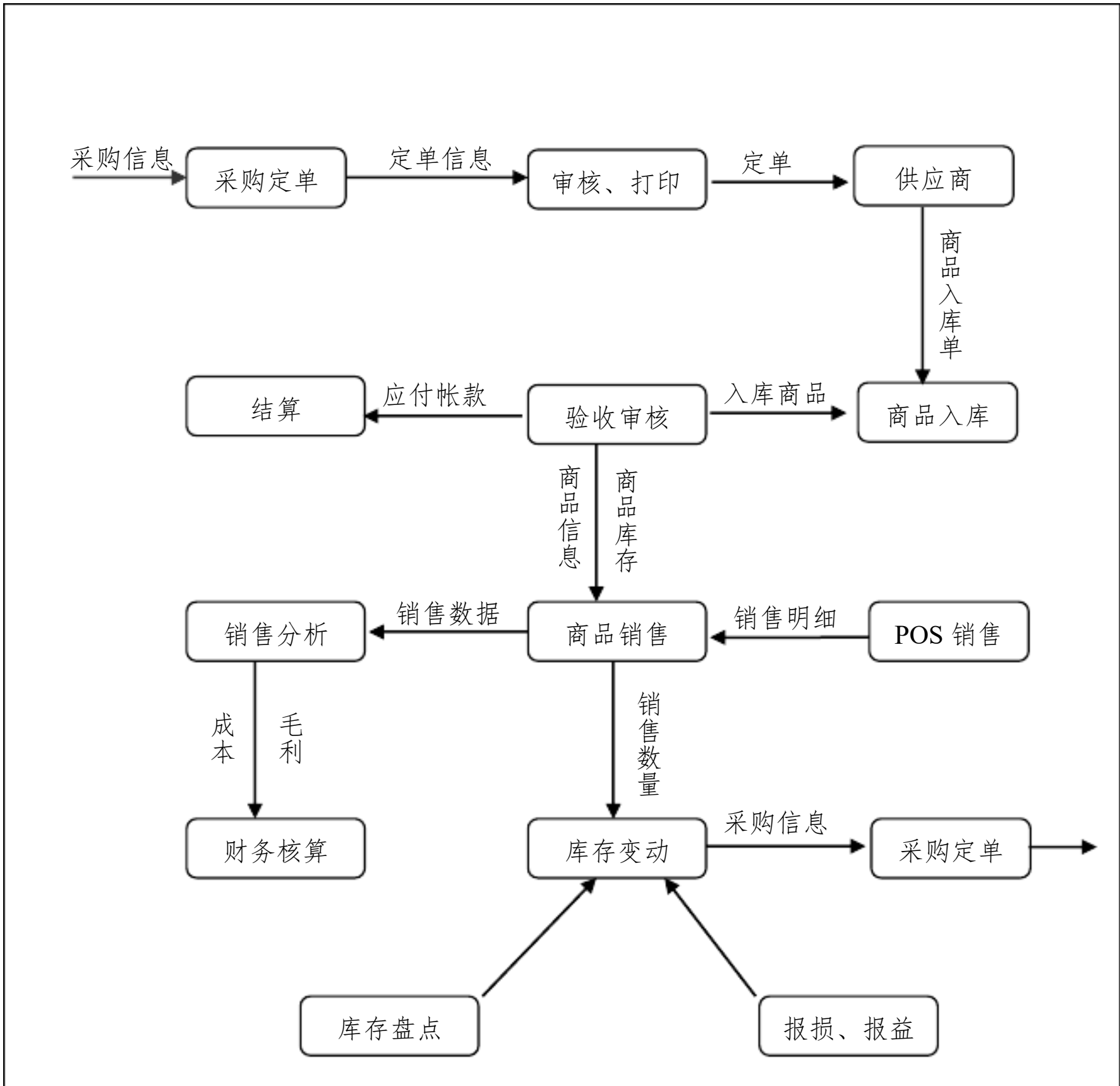
（三） 问题解决方案

超市系统在运行过程中数据流量是非常巨大的。有时候会因数据流量过大而使网络发生阻塞，使 POS 系统发生断网，直接影响销售活动的进行。这种情况是应该坚决避免的，应该尽量减少网络的数据流量。这一思想在本系统中的直接体现就是采用日处理和运用本地缓存技术。日处理的主要目的就是把当天的销售数据整理汇总，更新商品库存表，为第二天的销售做准备。与日处理不同的另一种处理方式就是每次商品销售完成以后就立即更新库存。但这样会使得服务器的负荷非常的重，经常会使 POS 系统发生断网。POS 系统销售商品时所取的商品信息是服务器上的数据，有效的保证了数据的一致性和集中控制。虽然商品信息来自服务器，但我们使用了本地缓存。POS 系统会把取得的数据暂时的存放在本地的内存，当要读取商品信息时，会先在本地缓存里查找，如果成功的话，就不必访问服务器。这样一来就有效的减少了网络的数据流量和服务器的负荷。此外，系统充分利用了客户机 / 服务器结构的特点，比较复杂的查询更新等操作都用存储过程来实现。存储过程在服务器端执行完毕仅返回执行结果，能有效的减少网络的通讯流量。

（四） 数据流图

对整个业务活动进行分析归纳后，我得出如下的数据流图：

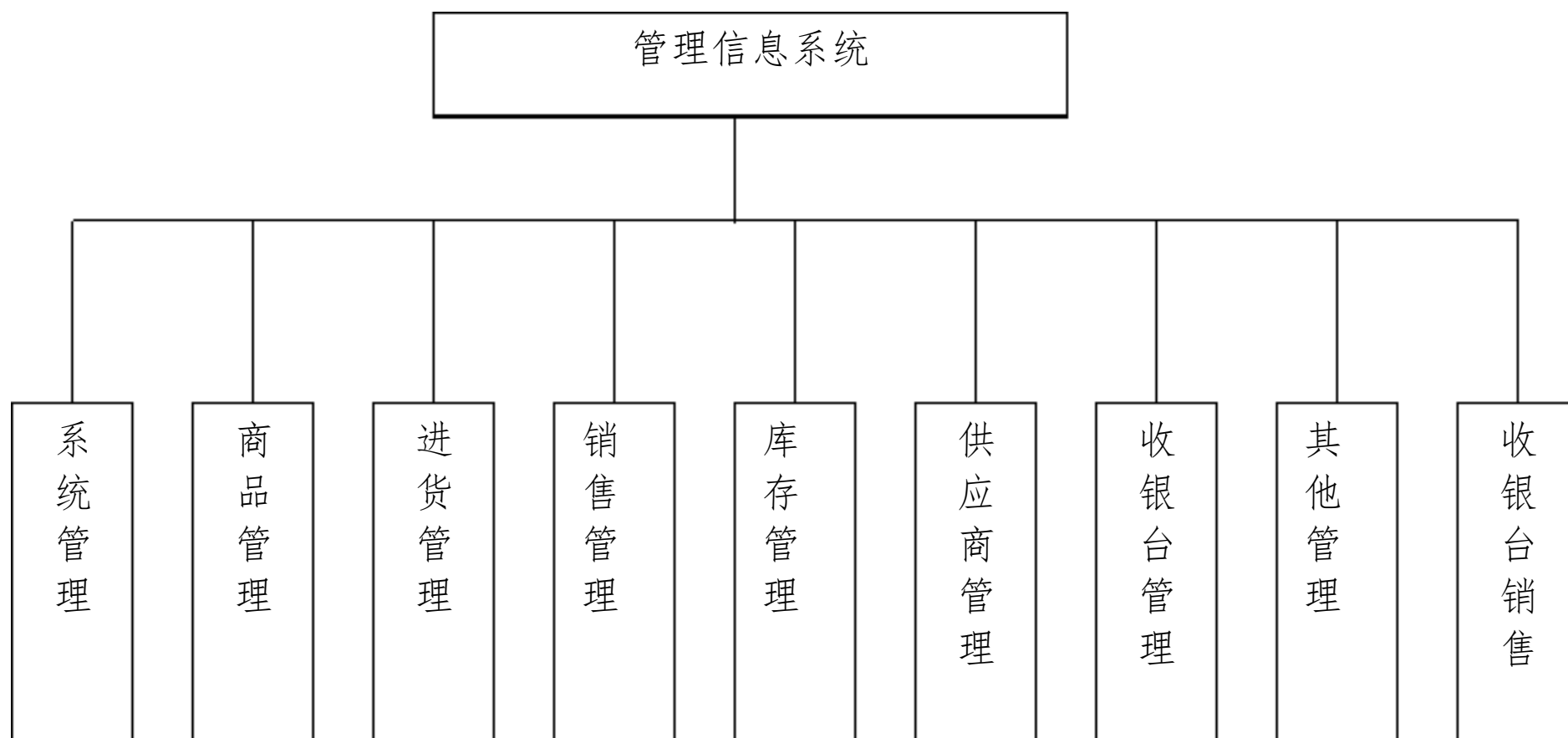
数据流程图



二 系统设计

(一) 系统模块的划分

根据前面所作的需求分析，我把整个系统分为如下几个模块：



(二) 数据字典

商品入库单明细 ENTER_GOODS_DETAIL

字段名称	数据类型	描述
ENTER_DAN_ID	int	入库单 ID
GOODS_ID	int	商品 ID
QUANTITY	numeric(18, 2)	入库数量
AMOUNT	numeric(18, 2)	入库金额
STATUS	char(4)	审核状态
WORK_PERSON	smallint	操作人员 ID
WORK_DATE	datetime	记帐日期

商品信息 GOODS_INFO

字段名称	数据类型	描述
GOODS_ID	int	商品 ID
BAR_ID	int	商品条形码
GUIGE	char(4)	规格

COST_PRICE	money	进价
BUY_PRICE1	money	正常售价
BUY_PRICE2	money	促销/折扣销售价格
LARGE_TYPE	smallint	所属部门号
SMALL_TYPE	smallint	所属类别
MAX_STORE	numeric(18, 2)	最大库存
MIN_STORE	numeric(18, 2)	最小库存
GYS_CODE	int	供应商代码
STATUS	char(4)	商品进货状态
FLAG	char(4)	促销/折扣标记
BUYER	int	采购员 ID
VAR_DATE	datetime	登记日期

供应商信息 GYS_INFO

字段名称	数据类型	描述
GYS_CODE	int	厂商代码
NAME	varchar(20)	企业全称
ADDRESS	varchar(20)	地址
TEL	char(20)	电话号码
RELATION	char(10)	联系人
TYPE	char(4)	类型
BANK	char(4)	开户行
ACCOUNT	varchar(20)	帐号
MENO	varchar(20)	备注

部门信息 LARGE_TYPE

字段名称	数据类型	描述
LARGE_TYPE_ID	int	部门代码
NAME	varchar(10)	部门名称

商品分类信息 SMALL_TYPE

字段名称	数据类型	描述
SMALL_TYPE_ID	int	商品分类代码
NAME	varChar(10)	分类名称

人员信息 PERSON_INFO

字段名称	数据类型	描述
PERSON_ID	int	人员代码
NAME	varchar(10)	姓名
SEX	char (2)	性别
AGE	smallint	年龄
DUTY	char(8)	职务
DEGREE	char(8)	学历
LOGIN	varbinary(20)	口令
MENO	varchar(20)	备注

前台销售表 POS_SALE

字段名称	数据类型	描述
GOODS_ID	int	商品 ID
BAR_ID	varchar(10)	条形码
QUANTITY	numeric(18, 2)	销售数量
AMOUNT	money	销售金额
CASHER	smallint	收银员 ID
P_NO	int	交易流水号

收银台设置 POS_SET

字段名称	数据类型	描述
POS_ID	smallint	收银台号
WORK_PERSON	smallint	收银员 ID

商品变价记录表 PRICE_ADJUST

字段名称	数据类型	描述
GOODS_ID	smallint	商品 ID
COST_PRICE	smallint	商品进价
LAST_PRICE1	money	当前售价一
LAST_PRICE2	money	当前售价二
NOW_PRICE1	money	所调售价一
NOW_PRICE2	money	所调售价二

	varchar(20)	变价原因
WORK_PERSON	int	操作员 ID
WORK_DATE	datetime	变价日期

库存调整记录表 STORE_ADJUST

字段名称	数据类型	描述
WORK_DATE	datetime	记帐日期
GOODS_ID	int	商品 ID
QUANTITY	numeric(18,2)	调整前数量
AMOUNT	money	调整前金额
ADJUST_QUANTITY	numeric(18,2)	调整数量
DJUST_AMOUNT	Money	调整金额
ADJUST_REASON	int	调整原因 ID
WORK_PERSON	int	操作员 ID

日处理历史状态 DIALY_DAN

库存调整原因表 STORE_ADJUST_REASON

字段名称	数据类型	描述
ADJUST_ID	int	代号
NAME	smallint	摘要

商品库存表 STORE_DETAIL

字段名称	数据类型	描述
GOODS_ID	int	商品 ID
QUANTITY	numeric(18,2)	数量
AMOUNT	money	金额
MENO	varchar(10)	备注

商品促销/折扣记录 GOODS_PROMOTE_LOG

字段名称	数据类型	描述
WORK_DATE	datetime	记帐日期
GOODS_ID	int	商品 ID
CURRENT_PRICE	Money	当前售价
PROMOTE_PRICE	money	促销价

	datetime	开始日期
END_DATE	datetime	结束日期
WORK_PERSON	int	操作员

字段名称	数据类型	描述
WORK_DATE	smalldatetime	日处理时间
STATUS	char (4)	状态
WORK_PERSON	int	操作员 ID

权
限
表
MEN

U_FLAG

字段名称	数据类型	描述
ID	smallint	代号
N1...42	smallint	标志

有了前面的工作，我们选择就进入编码阶段了。

(一) POS系统的开发

1. 功能及实现方法

POS系统是作为一个单独的程序开发的，主要在POS机上运行，完成商品的交易任务。销售的商品信息存储在POS_SAL表中。我们把一次交易放在一个事务提交，保证了数据的完整性和数据的正确性。界面以灰色调为主，使得收银员长时间工作眼睛不感到疲劳。收银的操作也很简单方便。交易流水号自动增加。

当某种商品处于促销或折扣销售状态时，POS系统自动取商品的促销或折扣售价，无需人工干预，并且促销或折扣期结束时，系统能自动识别恢复正常售价。这一个功能的实现是通过商品视图来完成的。请看商品视图源码。

前台POS销售时取商品信息时用到的视图：

```
CREATE VIEW V_SPXX
AS
SELECT NAME , A.GOODS_ID,A.BAR_ID,A.GUIGE,
BUY_PRICE= CASE
    WHEN      (GETDATE())>B.END_DATEAND GETDATE()<B.BEGIN_DATE)OR
              A.FLAG='正常' THEN A.BUY_PRICE1
    WHEN      (GETDATE()<B.END_DATEAND GETDATE()>B.BEGIN_DATE)AND
              A.FLAG='促销' THEN A.BUY_PRICE2
    WHEN      (GETDATE()<B.END_DATEAND GETDATE()>B.BEGIN_DATE)AND
              A.FLAG='折扣' THEN A.BUY_PRICE2
    ELSE A.BUY_PRICE1
    END
FROM GOODS_INFO A ,GOODS_PROMOTE_LOG B
WHERE A.GOODS_ID*=B.GOODS_ID
```

交易是在一个事务里完成的，请看交易完成数据提交的代码：

```
if form_pos.Database1.InTransaction then
```

```
    try
```

```
        form_pos.query1.ApplyUpdates;
```

把本地缓存修改的操作提交服务器

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/235121143201012002>