

Design and Performance Analysis of the Coupon Accounting System

Qiong Liu

International School of Software, Wuhan University, Wuhan
Email: liuqiong60099@126.com

Received: Dec. 10th, 2012; revised: Dec. 12th, 2012; accepted: Dec. 15th, 2012

Abstract: With the development of computer technology and network, office automation technology has been rapidly popularized. A mobile phone dealer has set up an office automation system, however, when they started the coupon promotion inside the whole marketing network recently, it was found that the accounting for coupons cannot be accomplished by the original system. The work load of the accounting is quite large, and our software system is designed for this issue. We adopt effective optimization measures based on the analysis of the computation bottlenecks, and perform comparison test for the different computation schemes. The results show that the computation scheme in our software system is proper and effective, which consumes much less time than the other schemes and is completely acceptable for the users.

Keywords: Office Automation; C Language; Access Databases; Object-Oriented Technology

营销代理网点结算系统的设计及其性能分析

刘琼

武汉大学国际软件学院, 武汉
Email: liuqiong60099@126.com

收稿日期: 2012年12月10日; 修回日期: 2012年12月12日; 录用日期: 2012年12月15日

摘要: 计算机及网络技术的发展使办公自动化技术得到迅速的普及, 但销售商在其营销网络内使用的优惠券促销机制无法通过原有的办公自动化系统实现。此项结算的计算量庞大, 针对以上问题, 本文设计出营销代理网点结算系统。并在认真分析计算过程中的性能瓶颈的基础上, 对计算过程进行了有效的优化, 对不同的计算思路、方案开展了对比测试。测试结果证明本系统采用的计算方案从思路、优化策略上都是正确而富有成效的, 其计算耗时大大低于其它方案。

关键词: 办公自动化; C#语言; Access 数据库; 面向对象技术

1. 引言

过去, 处于管理的需要, 电信公司为该营销网络搭建了基本的办公自动化系统, 网点信息、各个网点的员工信息、销售业务信息都可以通过一个网站系统进行提交、汇总、查询。2009年初, 国家工信部发放了3G运营牌照。为了吸引更多用户使用电信的网络, 电信公司对移动电话销售的重视程度大大加强, 在整

个营销网络内实行了优惠券促销机制。客户购买移动电话或使用电信公司的某些服务, 可获得不同面额的优惠券, 凭优惠券可到任意营销网点优惠购买移动电话。每月底, 各营销网点需到总公司将本月销售所收优惠券结算兑现。

新的促销机制并不复杂, 然而由于优惠券的结算额并不等同于面额, 算法比较麻烦, 并且整个营销网

络的每月销售数量很大，所以每月底的优惠券结算问题就成为一个计算繁琐、工作量巨大却又不容半点查错的棘手问题。营销网络内的办公自动化系统不具备结算功能，但已经长期运行，牵涉方方面面，无论从技术上还是从工作流程上都难以进行更新升级。为此，电信公司决定采用折中方案，即单独为优惠券结算问题开发专用的统计、结算和报表生成软件^[1,2]。

本系统即为解决上述问题而研发。系统的运行将原本依靠人力需要半个多月的工作缩短到几分钟即可完成，极大释放了劳动力资源。更重要的是，原本在超负荷的压力下极容易出现统计错误，而且很多错误因工作量原因无人发现也无人验证而成为糊涂账，通过本系统的采用，可完全避免此类错误。另外，本系统采用了独立运行的模式，不追求与现有办公自动化系统的整合，避免了“为适应计算机系统而调整办公流程”的现象，软件系统的启用未对现有办公流程提出任何新要求，所有办公室职员仍按原有方式开展工作，不受任何干扰^[3,4]。

2. 系统设计

本系统的首要目的是辅助计算，用户在使用本系统时，各项操作都是围绕最终的结算工作进行的。为了得到正确的结算数据，用户须确保各种相关的数据都被正确地录入并维护。只要保证相关数据的正确性，最后的结算工作仅仅是鼠标的一次点击而已。用户使用本系统时的操作流程如下图 1 所示，其中灰色的模块都是需要本系统实现的功能。

本系统是对现有办公系统的扩展，为了获取结算需要的相关数据，用户需要首先从现有的办公系统中导出，包括营销业务数据、销售人员信息、机型调价信息。准备好数据后，用户即可启动本系统，并使用用户名和口令登录。登录后本系统进入主界面。

之后用户有四个方面的工作需要完成，这四项工作没有时间先后的要求。

- 1) 将先前获取的营销业务数据导入系统；
- 2) 将先前获取的销售人员信息与本系统中已经

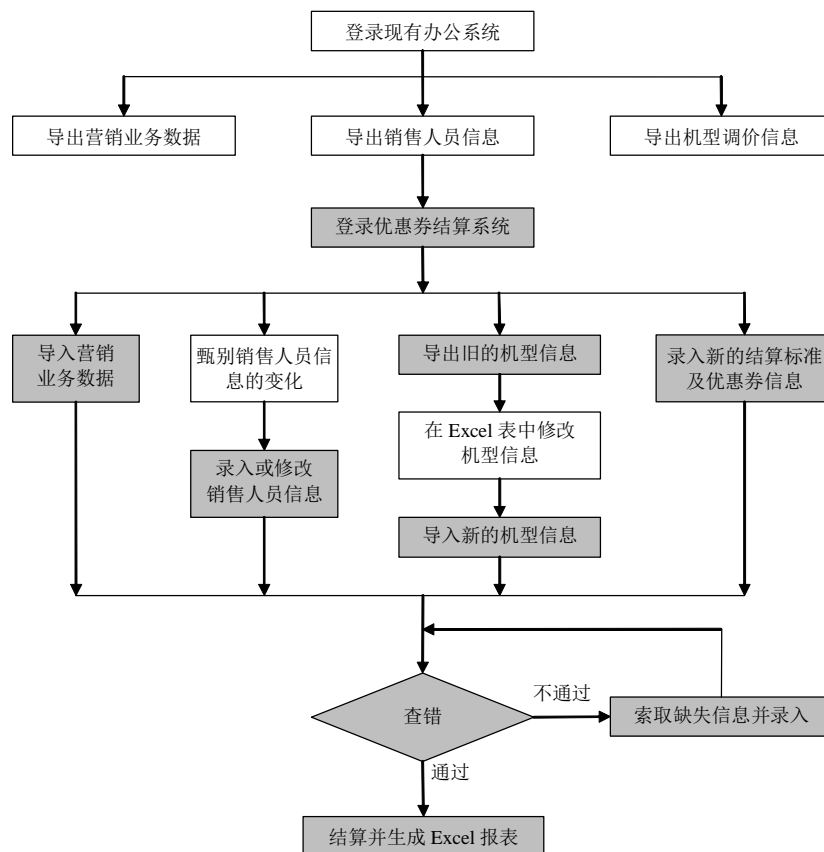


Figure 1. The operating procedures of coupon accounting system
图 1. 优惠券结算系统的操作流程

存在的工号信息进行对比,把新的员工、营销网点信息以手工的方式逐条录入系统;

3) 将现有的机型信息导出为 Excel 表格,并根据先前获取的机型调价信息修改该表格,然后将该 Excel 表格再导入系统;

4) 将新的结算标准和优惠券面额逐条录入系统(相关信息不能从已有的办公系统中获取,但用户会在第一时间从其部门领导处获得该信息)。

完成了上述四个步骤,也就初步完成了数据的准备。但由于已有的办公系统在某些技术环节或操作流程上不够完善,此时仍不能确保营销业务信息中涉及的工号、机型、优惠券都已正确录入本系统,所以必须进行查错。如果发现某些信息缺失,那么用户还需要设法从同事或兄弟部门处获取缺失的信息,录入本系统并再次查错,否则结算的结果将会有错误。如果查错通过,则可以进行结算并生成 Excel 报表。

结算的实现有两种思路:充分借助 SQL 语言,或者更多借助 C#程序判断每笔业务的实际结算额。前一种思路充分利用数据库的高效连接查询能力,效率更高;后一种思路实现难度小,程序更加直观。

由于系统涉及的数据量比较大,进行数据汇总时计算量较大,所以对计算效率比较敏感。我们选择前一种思路实现结算。其难度主要在于:

1) 在 SQL 语言层面实现业务日期与机型价格生效日期的匹配,即弄清楚每笔业务适用于哪一个日期生效的机型价格;

2) 在 SQL 语言层面实现机型、优惠券面额与结算标准的匹配及实际结算额的计算,即弄清楚每笔业务适用于哪种结算标准并计算出最终的结算额。

3. 具体实现思路及方法

实际结算额的计算无法通过一个 SQL 语句完成,我们的思路是利用已有数据生成两份中间结果,两份中间结果分别用于解决上一节提出的两个难点,并有助于最终的统计。

```
SELECT p1.name, p1.commencement, t1.sum_p, max(t1.category) AS category
FROM phone AS p1, ticket AS t1
WHERE p1.category<>0 and (t1.category=p1.category or t1.category=0)
GROUP BY p1.name, p1.commencement, t1.sum_p
UNION
```

需要说明的是,为了计算每笔业务最终结算额,我们并不针对每笔业务进行计算,而是将所有类型的机型与优惠券的搭配进行排列组合,并一一计算每种组合的结算额。每一笔业务总会适用于某一种组合,届时只要将业务数据与此处的中间结果简单地连接、查询即可得到结果。采用这种查询是基于本系统现实情况的考虑。如前所述,本系统中机型的数据量不会超过 1000 条记录,而优惠券的类型不会超过 50 种,所以二者的排列组合不会超过 50,000 种。这样的数据规模对于 Access 数据库来说是可以承受的。如果机型或优惠券的种类大大超越现有的水平,那么数据规模很可能会超过 Access 数据库的承受能力,并进而造成统计运算的效率大幅降低^[5,6]。

1) 在 Access 数据库中新建名为“business_commencement”的查询,实现业务日期与机型价格生效日期的匹配,其 SQL 命令如下:

```
SELECT business.id, max(phone.commencement)
AS commencement
FROM business, phone
WHERE business.d>=phone.commencement and
business.phone=phone.name
GROUP BY business.id;
```

对于任意一笔业务,如果仅考虑其对应的机型和优惠券面额,还不足以计算出最终的实际结算额,因为同一款机型在不同时期的价格是不一样的。为此,必须根据业务日期找出该机型的最近的价格生效日期,以该日期生效的价格为标准计算结算额。本项查询的结果仅包含两个字段:业务序号及其对应的机型价格生效日期。找出最近的价格生效日期并不困难,只要在早于业务日期的所有价格生效日期中选取最大值即可。

2) 我们在 Access 数据库中新建名为“phone_ticket_category”的查询,针对每种机型、优惠券面额的排列组合计算适用的结算标准,其 SQL 命令如下:

```
SELECT phone.name, phone.commencement, ticket.sum_p, ticket.category
FROM phone, ticket
where phone.category=0 and ticket.category=0;
```

这个查询由两个部分合并起来。对于适用于普通结算标准的机型，无论采用哪种面额的优惠券，都将采用普通的结算标准进行结算；对于适用于特殊结算标准的机型，则面临两种情况：该笔业务的优惠券面额可能既适用于这种特殊结算标准也适用于普通结算标准(那么该笔业务应按照特殊结算标准核算)，也可能仅适用于普通结算标准(那么该笔业务应按照普通结算标准核算)，应取可能的结算标准取值中数值最

大者。

通过这个查询，可以生成一个全面的排列组合，记录每一款机型在每一个历史价格阶段对应每一种面额的优惠券时适用的结算标准。这个查询并不包含我们直接需要的数据，但却是后一项查询的基础。

3) 在 Access 数据库中新建名为 “phone_ticket_actual” 的查询，针对每种机型、优惠券面额的排列组合计算实际结算额，其 SQL 命令如下：

```
SELECT phone.name, phone.commencement, phone.category,
       phone.price, ticket.sum_p, ticket.sum_a,
       Iif(phone.price<=ticket.sum_a,phone.price,ticket.sum_a) AS actual
FROM (phone INNER JOIN phone_ticket_category ON
      (phone.commencement=phone_ticket_category.commencement) AND
      (phone.Name=phone_ticket_category.name))
INNER JOIN ticket ON
      (phone_ticket_category.sum_p=ticket.sum_p) AND
      (phone_ticket_category.category=ticket.category);
```

这个查询的目的是生成一个全面的排列组合，记录每一款机型在每一个历史价格时期对应每一种面额的优惠券时的实际结算价格。很显然，这项查询基于前一项查询，因为前一项查询已经记录了每一种排列组合适用的结算标准，所以此项查询只要在标准的结算价格与机型的实际价格间取较小值。

SQL 语句中的 Iif 函数会根据括号中的判别式决定返回哪一个值。如果不使用这个函数，那么实现上述功能所需的 SQL 语句会更复杂，需要将两种情况分别查询然后再合并起来。充分利用 Access 数据库提供的函数可以有效提升程序的效率和可读性。

图中白色的模块表示表，灰色的模块表示查询。可以看到，为了得出最终的统计结果，我们对数据库进行了多轮查询，后一轮查询建立在前一轮查询结果的基础上。

我们知道，查询只是数据库中的一份数据视图，从数据结构、存储技术上都没有为进一步的查询提供充分的技术准备，其中最典型的问题就是不能对查询建立索引。所以，简单的多轮查询在性能上势必较差，尤其是数据量比较大的时候，这种性能的缺陷会明显得表现出来并有可能让用户无法接受。因此，有必要在多轮查询问题上进行充分优化。

4. 性能优化

4.1. 瓶颈分析

本系统最大的计算压力来自最终的结算统计。此处的计算可能耗费较多的时间，是系统中最重要瓶颈。结算统计方法流程图见图 2。

4.2. 优化方法

首先将上图中作为中间数据的查询结果输出到表中，使其以表的形式存放在数据库中，然后对该表建立索引。后续的查询不再直接以前面的查询为基础，而是基于建立了索引的表。以 “phone_ticket_actual” 查询为例，相关的优化代码如下：

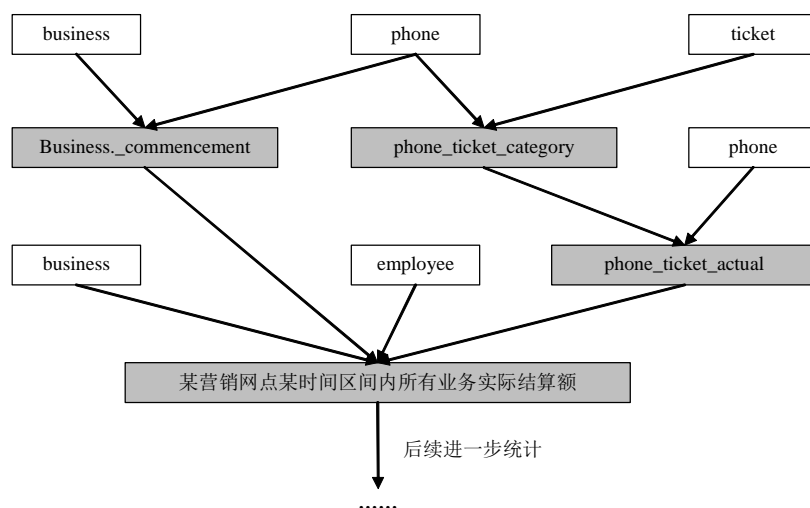


Figure 2. Accounting statistical flowchart
图 2. 结算统计流程图

```
temp_cmd.CommandText = "select * into temp_table_1
                        from phone_ticket_actual";
temp_cmd.ExecuteNonQuery();
temp_cmd.CommandText = "create index temp_table_1_name
                        on temp_table_1 (name)";
temp_cmd.ExecuteNonQuery();
temp_cmd.CommandText = "create index temp_table_1_sum_p
                        on temp_table_1 (sum_p)";
temp_cmd.ExecuteNonQuery();
```

这样一来，“phone_ticket_actual”的查询结果就被输出到了“temp_table_1”表中，并且该表在“name”字段和“sum_p”字段上分别建立了索引。对“business_commencement”和其它查询结果的处理方法与此方法类似。

这样的优化方法虽然在两次查询之间增加了新的步骤，看似更加繁琐，但是磨刀不误砍柴工，建立表和索引可以将数据进行必要的整理，使后续的查询工作更加顺利地进行。如果不做这样的优化，后续的查询速度将一轮比一轮慢，直到最后完全让人无法接受^[7-9]。

4.3. 相关性能测试

为了测试优化的效果，我们开展了性能测试。测试比较三种方案：

1) 较少使用 SQL 命令，借助 C#程序判断每笔业

务的实际结算额；

2) 使用 SQL 命令实现结算额的计算，但不进行任何优化；

3) 使用 SQL 命令实现结算额的计算，并进行优化。

测试将在同样的硬、软件平台下开展，分别测试业务量为 5000 条记录、10,000 条记录、15,000 条记录、20,000 条、25,000 条记录时统计计算花费的时间。得到的结果如图 3。

通过上图可以看出，当业务数据量比较小(少于 5000 条记录)时，借助程序逐条分析计算的速度甚至快于借助 SQL 但是不采取优化的方案。但随着数据量增大，借助程序分析的方案耗时大幅上升，远远超过了另外两个方案，并且当业务数据量超过 20,000 条时，该方案的耗时量已达到或超过 100 秒，对于用户来说已经难以接受^[10,11]。

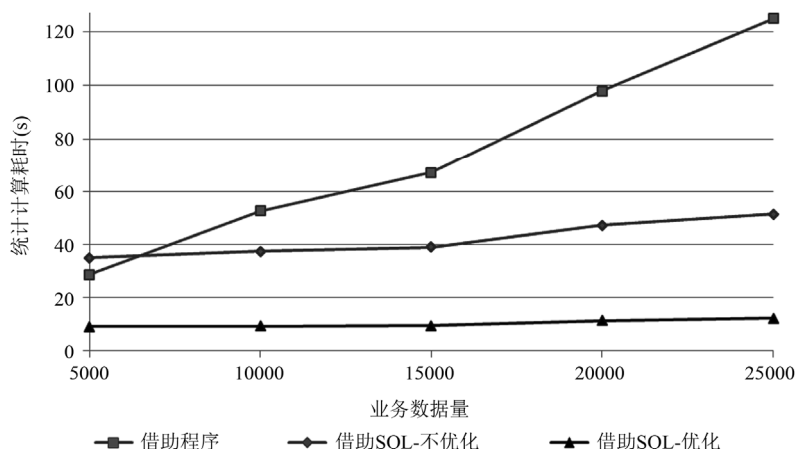


Figure 3. The performance comparison of different schemes
图 3. 不同方案进行统计计算的性能比较

借助 SQL 命令进行计算的两个方案的计算耗时同样随着业务量增大而增加，但是其增加幅度非常缓慢，即便业务数据量达到 25,000 条，其耗时也不超过 5000 条记录运算耗时的 1.5 倍，这就让用户在使用本系统时对于统计计算的耗时可以获得相对准确的期望而不必过多关注业务数据量。相比较而言，本系统采用的经过优化的方案表现出了极为明显的性能优势，不仅在所有业务数据量的计算耗时上都大大低于另两种方案，而且其计算耗时随业务数据量的增长也最为缓慢。当业务数据量达到 25,000 条时，该方案的运算耗时也不过 12 秒，比 5000 条业务数据量时仅增长了不到 3.5 秒，因此对于用户而言完全可以接受。由此可见，本系统采用的优化方案完全必要，并且效果明显。

5. 结论

在细致的需求分析和对相关编程技术熟练掌握的基础上，本系统的程序实现得以顺利完成。在此过程中，我们主要关注了以下技术重点或难点：数据库的连接与访问、对话框的设计、口令的计算与存储、数据导入与导出的实现、第三方应用程序的启动以及结算的实现。通过多方查阅技术资料以及审慎的思考，上述技术重点和难点全部被顺利解决，系统的各项预期功能全部实现。

结算功能是本系统最受关注的功能，也是各项功能中计算负荷最大的，其计算效率对整个系统的用户体验至关重要。本系统认真分析了计算过程中的性能瓶颈，对计算过程进行了有效的优化。通过对比测试，证明本系统采用的计算方案从思路、优化策略上都是正确而富有成效的。

参考文献 (References)

- [1] 张海滨. 办公自动化理性浅析[J]. 办公自动化(综合版), 2010, 20: 10.
- [2] 张卓其, 史明坤. 网上支付与网上金融服务[M]. 大连: 东北财经大学出版社, 2002.
- [3] 王珊. 数据库系统概论第四版[M]. 北京: 高等教育出版社, 2008.
- [4] G. Booch. 面向对象分析与设计第三版[M]. 北京: 人民邮电出版社, 2009.
- [5] R. Patton. 软件测试第二版[M]. 北京: 机械工业出版社, 2006.
- [6] 张宽海. 网上支付与结算[M]. 北京: 机械工业出版社, 2008.
- [7] 周虹. 电子支付与网络银行[M]. 北京: 中国人民大学出版社, 2006.
- [8] 中国电子商务协会编著. 第三方电子支付探索与实践[M]. 北京: 中国标准出版社, 2008.
- [9] 张经松. 网上电子支付与结算[M]. 北京: 人民邮电出版社, 2011.
- [10] 特班. 电子商务管理视角第 5 版[M]. 北京: 机械工业出版社, 2010.
- [11] 吕廷杰. 网络经济与电子商务[M]. 北京: 北京邮电大学出版社, 2000.