

企业信用管理对策及软件实现

邓礼全(副教授)

(电子科技大学成都学院 成都 611731)

【摘要】信用管理的实质是对客户实施分类精益管理,有计划地防控客户信用风险,确保应收账款回收,同时可加快企业资金周转,以获取更大利润。本文分析了企业信用管理存在的问题,探讨了利用5C原则进行信用管理需要注意的事项,详细阐述了进行信用管理的策略和利用计算机软件实现策略的设计思想。

【关键词】信用管理 信用控制 5C原则 计算机软件

市场竞争的日趋激烈使得许多企业采用信用销售措施去招揽客户,扩大市场占有率。伴随着营业额的扩大,企业的应收账款也呈居高不下的态势,信用管理出现严重的漏洞。因此,采取有效措施进行信用管理,确保应收账款可预期、可控制、可跟踪已经成为企业进行市场营销活动的当务之急。

一、信用管理的原则

信用管理是企业获得长期收益和短期收益的动态平衡,并以不断增强企业核心竞争优势为根本目标。成功的企业都具有科学的信用管理标准和机制。

企业信用管理具有客户档案管理、客户授信以及信用控制三大基本功能。它是通过收集处理客户信用数据、对客户的信用质量做出评估、对客户进行信用授予、利用信用政策进行信用控制以及对信用销售后产生的应收账款进行跟踪管理的过程。

1. 信用授予的原则。在对客户信用授予前,可以利用5C原则从品质 character、能力 capacity、资本 capital、抵押 collateral 和条件 condition 这五个方面评价确定客户的信用等级。由于企业所处的环境和自身抗风险的能力不同且不断变化,因而这些原则的内容和标准也有所变化。

基本的要求是:①品质(Character)。指客户诚实守信、履行偿债义务的可能性,其主要表现在客户负责人的道德品质。②能力(Capacity)。指客户履约的物质基础和盈利能力。物质基础包括流动资产的质量数量、流动资产与流动负债的比例,需要注意客户财务数据的真实性;另一方面,要注意考察客户短期盈利和长期盈利能力的平衡性。③资本(Capital)。指客户的财务实力和财务状况,表明客户偿还债务的财务基础。特别要注意客户的负债比率、流动比率、有形资产净值等财务数据。④抵押(Collateral)。指当客户无力支付款项时被用作债务担保的资产。这

是评估客户信用标准时的底线,是企业迫不得已保护自身利益的基本条件。⑤条件(Condition)。泛指影响客户付款的经济环境,包括宏观环境、中观环境和微观环境。这些环境相互影响,相互作用,并具有一定的互补性和替代性。

明确了上述原则的具体内涵后,首先应当根据所处行业的具体特点和竞争态势,进行定量和定性分析,制定详细的评比标准。其次,对于5C,企业要考虑自己的实力、近期目标和长期目标,运用不同的权重加以衡量,以得到满意的综合结果。再次,注意评定标准的相对性。既要与本企业进行纵向比较,也要与竞争对手进行横向比较。只有比竞争对手做得更好,就可以获得企业的竞争优势。

2. 信用管理的方法。利用5C原则评估客户后,可将客户信用等级分为ABCDE或12345级,授予客户对应的信用额度和信用期限,比如:5万元30天、10万元60天等。目前企业是查阅客户信用信息,在开订单时根据客户的订货金额和企业对客户的授信情况进行控制。这个过程如果是手工操作,则控制的时效性和准确性很差,受到的人为干扰因素多。而采用计算机系统在线进行信用控制能够消除这些缺点,及时准确地进行信用管理。

二、信用管理软件的结构

信用管理在企业中介于财务部与销售部门之间,信用管理系统是两个部门都可能使用的软件。设计时,必须从管理上考虑几个问题:第一,如何表达客户信用额度和信用期限这两个信用要素;第二,信用控制的流程如何确定;第三,产生的应收账款如何计算、如何传递。为此,我们可以通过技术手段解决管理中的难题,将信用管理软件由表及里分为3个层次,分别为:表示层、业务逻辑层和数据访问层。

1. 表示层。表示层是展现给用户的层面,比如:应收

账款的账龄分析表、被挂起订单的订单表、客户表等,可以用视图也可以用报表实现。在软件系统中,可以用功能页面、用户控件以及相关的样式表和图片文件加以表述。

2. 业务逻辑层。针对具体的信用控制问题,对客户数据和订单数据进行逻辑控制,其核心控制思想是当客户的赊销订单期限超过信用期限或者客户的赊销订单金额超过信用额度时,自动挂起新开的赊销订单,等待客户付款结算部分超期超限赊销订单,使客户信用状况满足赊销要求后,才能继续赊销订单流程,进而产生送货单据和发票,最终完成销售过程。

3. 数据访问层。在面向对象C#中,是访问数据库的各种操作,如连接和断开数据库、读取或写入数据库中的数据,比如:客户信用数据、订单数据等,这些访问在管理层面是不可见的,但技术上是必要且封装的,可以将图1数据表中的数据读入到图2对应类对象的数据成员中,对这些数据成员进行相关处理以实现在线信用控制。

三、信用管理系统的实现

1. 数据表及对应的类定义。我们以SQL Server数据库为例说明相关数据表的设计,并用C#定义这些数据表达的类。在这个系统,需要多个数据表,我们说明与信用管理密切相关的三个表:①客户财务信息表(customer);②订单头表(orderhead);③应收款账龄分析表(receivableaging)

这三个表用customer_id进行关联,在customer中customer_id是主关键字,而在orderhead和receivableaging中是外部关键字,它们的结构及关联情况如图1所示。

在C#中可以定义三个类来产生对象,读取和写入对应数据表的数据,用于处理信用控制问题,如图2所示。

2. 信用控制的策略。为了实现信用控制,客户资料表除了包含必要的描述客户号、客户名、客户地址等信息外,还必须包含能进行财务控制的信用信息,如图1的customer表所示。

credit_limit表示是否需要对该客户进行信用控制,需要进行信用控制的,该控制选项为真“True”。

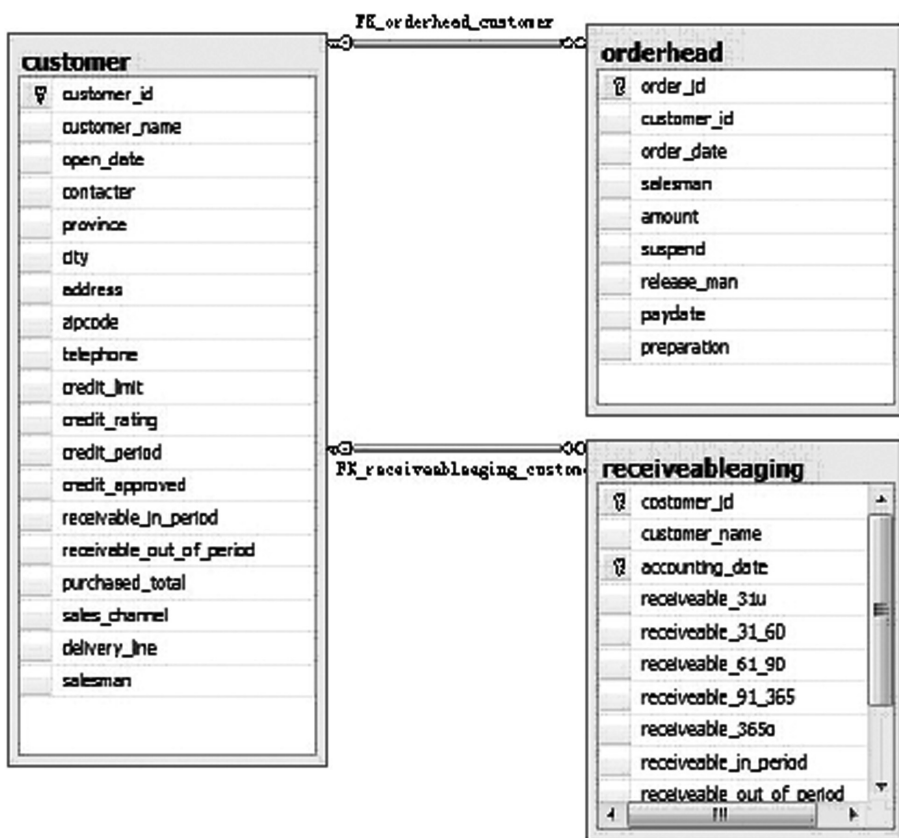


图1 关联的数据表

credit_period是对该客户授予的信用期限,比如:30、60等,表明该客户可赊购30天或60天;现金客户的credit_period为0天。

credit_approved是对该客户授予的信用额度,比如5万元,表明该客户可赊购5万元的货物;现金客户的credit_approved为0元。

receivable_in_period是该客户在信用期限内尚未付款的订单合计金额, receivable_out_of_period是该客户超出信用期限尚未付款的订单合计金额。这两个金额是通过orderhead中未付款订单的amount值,根据当前日期与order_date的时间差大小分别按照credit_period汇总得到的,可每天结算得到最新数据以便次日进行赊销信用控制。

计算机信息系统在赊销开订单时通过查询客户财务信息表的相关资料决定是否对该客户的订单进行实时在线的挂起操作,控制的策略是:

A: credit_limit为“True”并且receivable_out_of_period>0,这是对客户欠款是否超出信用限期进行判定。

B: credit_limit为“True”并且(新开订单金额amount+receivable_in_period + amountreceivable_out_of_period) > credit_approved,这是对客户欠款是否超出信用限额进行判定。

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace Credit
{
    public class cCustomer          //客户类
    {
        private string customer_id;    //客户编号
        private string customer_name;  //客户名称
        private bool credit_limit;     //为 True 时需要进行信用控制
        private float credit_period;   //对该客户授予的信用期限
        private float credit_approved; //对该客户授予的信用额度
        public float credit_in_period;  //信用期内的应收款
        public float credit_out_of_period; //信用期外的应收款
        .....
    }
    public class cOrderHead        //订单头类
    {
        private string order_id;      //订单号码
        private string customer_id;    //客户编号
        public string order_date;      //订货日期
        public float amount;           //订货金额
        public bool suspend;           //为 True 时挂起
        public string release_man;     //订单释放批准人
        .....
    }
    public class cReceivableAging    //应收款账龄类
    {
        private string customer_id;    //客户编号
        private string accounting_date; //结算应收款的日期
        public float receivable_31u;    //1~30天的应收款
        public float receivable_31_60;  //31~60天的应收款
        public float receivable_61_90;  //61~90天的应收款
        public float receivable_91_365; //91~365天的应收款
        public float receivable_365o;   //365天以上的应收款
        public float receivable_in_period; //信用期内的应收款
        public float receivable_out_of_period; //超过信用期的应收款
        .....
    }
}

```

图 2 C#中的相关类

当 A、B 任一条件成立时,该客户的赊销订单将被挂起,图 1 订单头表 orderhead 中的 suspend 置为真“True”,直到该客户付款后 A、B 条件不成立或者经过企业信用控制主管同意释放该订单、suspend 置为“False”后,销售流程才能得以继续下去,产生相应的提货单和发票。此时该订单上的 release_man 将记录释放该订单的信用控制主管姓名,以便实现对信用控制的跟踪和反馈,做到信用管理有序进行。

3. 信用管理的跟踪。信用管理是一个事前监督、事中控制、事后跟踪的过程,事前监督主要是检查对客户的信用政策是否合理、信用期限和信用额度的数据录入是否正确;事中控制是按照信用政策对销售业务流程中的订单进行处理,挂起欠款超期超限的客户订单,按照信用主管的意见进行控制和释放;事后跟踪就是每天提供客户应收款账龄分析表,供销售部门催款和财务部门资金调配使用。该表结构见图 1 中的 receivableaging 和图 2 中的类定义给出的相应数据项解释。

应收款账龄分析表 receivableaging 中的分类数据是通过汇总订单头表 orderhead 中未付款订单的 amount 得到的。相应账龄可以利用 accounting_date 与 orderhead 中的 order_date 之间的时间差得出。只要设置适当的 accounting_date,随时可以计算出对应时间的客户应收款账龄,从而获得该表数据,跟踪财务部门的信用控制水平和销售部门的工作业绩,完善信用管理的层次,提高企业信用管理水平。

主要参考文献

1. 孙森.信用管理.北京:金融出版社,2012
2. 中国合作贸易企业协会.中国企业信用管理发展报告.北京:中国经济出版社,2012
3. 内格尔(Nagel).C#高级编程.北京:清华大学出版社,2013
4. 薛华成.管理信息系统.北京:清华大学出版社,2012
5. 赖茂生.信息资源管理教程.北京:清华大学出版社,2012
6. 谢邦昌.SQL Server R2 数据挖掘.北京:水利水电出版社,2011