

基于语法树的批价优惠研究与设计

王爱平, 国玮玮, 李仿华, 徐晓艳

(安徽大学 计算机科学与技术学院, 安徽 合肥 230039)

摘要: 以中国电信行业计费系统为研究对象, 从批价优惠业务需求层面进行分析, 提出了基于语法树的批价优惠方法, 并在理论和实践上给出了一个电信行业中应用的解决方案。

关键词: 计费系统; 语法树; 批价; 解决方案

中图分类号: TP311

文献标识码: A

文章编号: 1674-7720(2011)19-0087-04

Research and design of rating privilege based on syntax-tree

Wang Aiping, Guo Weiwei, Li Fanghua, Xu Xiaoyan

(School of Computer Science and Technology, Anhui University, Hefei 230039, China)

Abstract: Taking the China Telecom's charging system of as the object of study, this paper proposes a method of rating privilege based on syntax-tree from the business requirements levels of rating. It gives a feasible solution in the telecom industry theoretically and practically.

Key words: charging system; syntax-tree; rating; solution

计费系统^[1]是电信运营商业务支撑系统中最为核心的部分, 其灵活性、响应速度、支撑能力的高低也在很大程度上决定了前端业务模式的多样性和客户体验的满意程度。随着电信行业的发展, 电信运营商需要构建一个能够支持 3G 网络多业务发展的融合计费系统, 以满足复杂的资费套餐以及多业务融合的需求^[2]。

面对更加复杂的全业务运营模式, 运营商想要推出适应全业务运营环境下的计费系统, 必然需要对原有的计费系统进行整合和改造。为了提高计费系统性能、支持多业务组合、个性化资费套餐灵活配置的能力, 本文提出了基于语法树的批价优惠方法。

1 业务介绍

计费系统框架图如图 1 所示, 计费系统的核心是批价优惠处理, 批价优惠分为三种类型: 前向优惠、优惠和后向优惠。传统上的套餐是指优惠, 前向优惠和后向优惠是优惠的补充, 主要是针对所有用户的计费策略, 可以把其视为标准资费的补充。批价优惠处理的过程是, 首先与局数据 (公共数据中心) 建立连接进行全局零件库加载操作 (这里的局数据存储在内数据库 MDB^[3]中), 然后创建业务处理线程 (如网络服务线程、读文件处理线程、业务处理线程、写文件处理线程等), 根据业

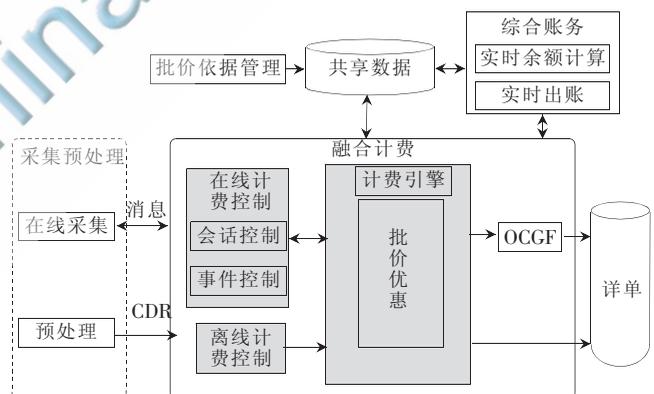


图1 计费系统框架图

务导航将 CDR 话单类 (如语音话单、数据业务话单、GPRS 话单、VPMN 话单等) 分配给不同业务处理线程处理, 即对用户话单中的费用进行优惠。

用户订购的每一个套餐资费即记账资费包括多个子项, 这些子项称之为子记账资费, 每一个子记账资费对应的有一个费率方案和一种场景, 每一个费率方案由不同的零件组合而成, 场景是用户打电话产生的信息, 包括用户品牌、日期、地区、呼叫类型、用户类型、费用类型等。优惠过程实质上是用记账计费, 子记账资费再到优惠零件一级一级分解任务并执行任务的过程, 因

《微型机与应用》2011年第30卷第19期

此,优惠过程是一个子记帐资费作用效果的描述串。这里记帐资费与子记帐资费的关系通过语法树实现,每一个子记帐资费唯一地使用一个优惠零件,即优惠规则,且唯一地匹配一套参数,理论上,由于采用纵表结构的优惠零件具有扩展性,因此所有的优惠计费策略都可以通过优惠零件来实现。所以在最关键的优惠处理过程中,用户订购的每一种记帐资费都可以通过优惠语法树进行描述、解析、计算。

2 语法树与零件设计

2.1 优惠语法树

2.1.1 语法树定义

语法树是一棵二叉树,其中非叶子节点的值为子节点或子树之间的关系,叶子节点为记帐资费或者子记帐资费。定义了以下五种业务关系:

(1)互斥关系(mut):两节点之间只能取一个节点执行(子记帐资费)。先执行左节点,如果符合优惠条件则执行优惠,否则执行右节点。

(2)叠加关系(add):在左节点优惠的基础上,再执行右节点。

(3)最优关系(max):取两个节点最大优惠值。分别执行两个子节点,然后进行比较,取优惠值最大节点。

(4)最劣关系(min):取两个子节点最小优惠值,max的逆操作。

(5)特殊叠加关系(mad):针对分段的费用对象取最优后进行组合,颗粒度更细,其他关系针对整段费用对象做总费用判断。一般针对打折跟忙闲时类优惠做 mad 的分时段最优处理。即在 add 的基础上,针对费用对象分段考虑。

2.1.2 构建语法树

假如用户订购了 A、B、C 三个记帐资费,A 和 B 两个资费为互斥关系,A 的子资费表达式为 add(F01, F02),B 的子资费表达式为 add(F03, F04),C 的子资费表达式为 add(F05, F06)。记帐资费的优先顺序为:A、B、C,所以资费表达式为: add(mut(add(a, b), add(c, b)), add(e, f)),优惠语法树构造如图 2 所示。

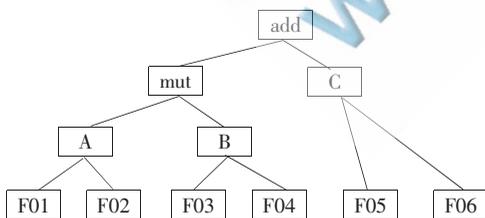


图 2 语法树

用户订购的所有套餐需要先按照优先级进行排序,以保证语法树的唯一性,然后按照套餐间的关系以及业务类型来生成记帐资费表达式。其实就是将所有的套餐构造一颗二叉树的过程。

《微型机与应用》2011 年第 30 卷第 19 期

2.2 优惠零件

(1)优惠零件定义

基本零件:也称简单零件,打折、置费率、减值优惠零件,实现最基础的操作。

派生零件:也称复杂零件,是在简单零件的基础上增加条件功能或者辅助逻辑功能,组合成功能更健壮优惠零件,如图 3 所示。

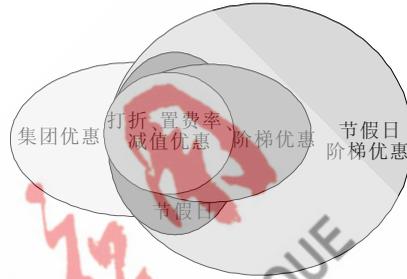


图 3 优惠零件

(2)优惠零件模板定义

在计费系统里,具体的计费优惠规则是体现在优惠零件里,根据优惠的业务特点把相近的一批优惠规则提炼出来,用一段简单的代码来实现这些规则,每一个优惠零件都代表了一种优惠规则。先使用 C++ 模板^[4]定义优惠零件,再用零件的有限组合构成各种优惠类型。

模板定义如下:

cFavBase: 每个具体的优惠零件都需要继承优惠零件基类;

cFav000: 对费用对象赋值的最基本的优惠零件,所有优惠零件最后都是归结为对次优惠零件的调用;

cFav001: 基本打折类优惠零件;

cFav002: 简单分段优惠零件;

cFav003: 忙闲时类优惠零件;

cFav004: 节假日简单无阶梯话单优惠类零件;

cFav005: 亲情号码优惠零件;

cFav006: 小区优惠零件。

因此可以组成多种优惠类型,举例如下:

基本打折类优惠零件: cFav001 <cFav000>;

忙闲时类优惠零件: cFav001 <cFav003 <cFav000, cFav000>>;

节假日简单无阶梯话单优惠类: cFav001 <cFav004 <cFav000>>;

亲情号码优惠: cFav001 <cFav009 <cFav000>>;

所以节假日,亲情号码与忙闲时:

cFav001 <cFav004 <cFav009 <cFav003 <cFav000, cFav000>>>>

3 语法树处理的 C++ 实现

图 4 是批价优惠处理的实现类图,为了简明只给出基本的属性和方法,图中 cDataBase 是 CDR、cFeeObj、cUserInfoNode、cStatisObj、cFavProc 的基类。话单基类 CDR

欢迎网上投稿 www.pcachina.com 97

存放话单的基本信息和中间变量结构体等信息,费用对象类 cFeeObj 创建费用对象,cUserInfoNode 用于生成用户资料节点地址对象,引用用户信息类 cUserInfo 类来创建用户资料节点上的记帐资费语法树,cStatObj 类负责创建累计对象,cFavProc 类负责创建优惠过程对象。

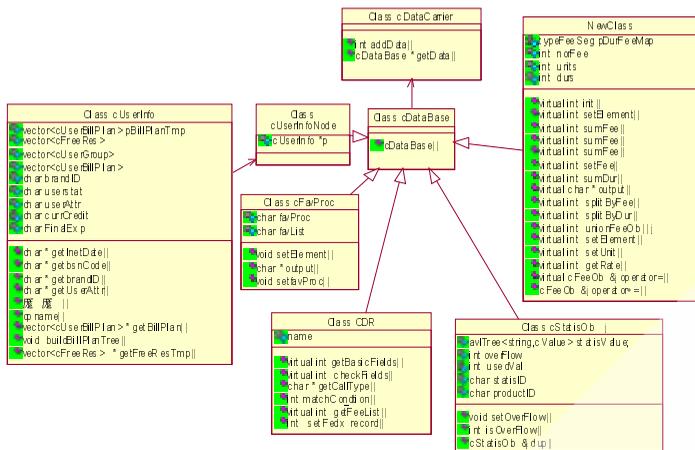


图4 业务处理类

3.1 记帐资费表达式的解析和调用

首先按照优先级对 cUserInfo 中的所有套餐进行排序,然后按照套餐间的关系以及业务类型生成记帐资费表达式,其实就是将所有的套餐构造成一颗二叉树的过程。下面以语音业务的记帐资费表达式为例说明表达式的解析和调用的过程。语音业务的优惠语法树如图5所示,生成的记帐资费表达式为 2: add (2: add (1: name: index; effdate; expdate; type; id, 1: name: index; effdate; expdate; type; id), 0: subname: index)。

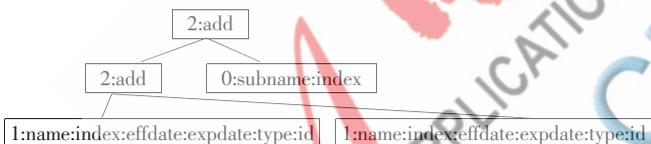


图5 语音业务的优惠语法树

从图5可以看出记帐资费表达式的结构实际上是一颗二叉树,解析记帐资费表达式的过程是先将表达式中信息放到一个结构体 p, 属性有节点类型 type、函数名称 name、左子树 pl、右子树 pr、生效日期 effDate、失效日期 expDate、索引 index 等。下面对记帐资费表达式构成的树中的节点进行说明:

(1)以“2:”开头,称之为函数节点,表明此节点有子节点(记帐资费或子记帐资费);“2: add”中的 2 代表节点类型,add 代表函数,表示子节点之间的关系,其中“add”表示子节点之间为叠加关系,执行左节点后又执行右节点,经过解析之后,p.type=2、p.name=函数名、p.pl=左子树、p.pr=右子树。

(2)以“1:”开头,称之为记帐资费节点,此节点为叶子节点,比如:1: name: index: effdate: expdate: type: id 分别代表的含义为节点类型、记帐资费名称、索引、开始时

间、失效时间、业务类型、产品 id;表达式经过解析后 p.type=1、p.name=记帐资费名称、p.index=索引、p.effData=开始时间、p.expData=失效时间。

(3)以“0:”开头,称之为子记帐资费节点,此节点为叶子节点,比如:0: c1: 18258 分别代表的含义为节点类型、子记帐资费名称、索引,表达式经过解析后,p.type=0、p.name=子记帐资费名称、p.index=索引。

根据记帐资费表达式进行批价处理函数为 favProcess(... char*expression, int sFlag..),先声明一个结构体变量 p,对表达式 expression 进行解析并将相关的信息存放放到变量 p 中,然后根据 p.type 和 p.name 做不同的递归调用。

(1)当 p.type=2、p.name=mut: 若 favProcess(... p.pl, int sFlag..) 的返回值不等于 0 时,整个函数返回整个返回值;否则返回 favProcess(... p.pr, int sFlag..);当 p.name=add、min、max 或者 mad 时进行不同方式的递归调用。

(2)当 p.type=1 时,根据 p.index 从局数据中获取名为 p.name 记帐资费的子记帐资费的组合表达式 billpalnexp, 然后递归执行 favProcess (... billpalnexp, int sFlag..)。

(3)当 p.type=0 时,根据 p.index 从局数据中获取名为 p.name 的子记帐资费的详细信息,生成子记帐资费费率计划对象 T_PP_SubProcPLAN*pSubBillPlan 并调用函数 execSubBillPlan (... expnod.name ..., pSubBillPlan, ...),再将其返回值作为 favProcess 的返回值;执行execSubBillPlan 的过程实际上就是调用优惠零件的过程。

3.2 优惠零件调用

因为优惠模板种类太多,挑出其中最基本的 cFavBase、cFav000、cFav001 和 cFav002 作为示例,类关系如图6所示,图中,类 cFavBase 是所有优惠零件的基类,每个具体的优惠零件都由它派生出来,并实现其中参数初始化和优惠过程的接口;cFav000 是 cFavBase 的子类,它是对费用对象赋值的最基本的优惠零件,所有优惠零件最后都归结为对此优惠零件的调用;cFav001 是基本打折类优惠零件;cFav002 是简单优惠分段优惠零件。

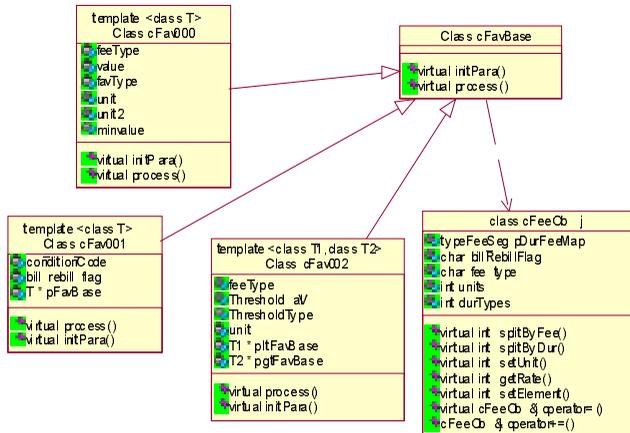


图6 优惠零件类图

调用 `exeSubBillPlan (cFeeObj*pFeeObj ... T_PP_SubPrPLAN *pSubBillPlan...)` 函数时, 实际上是根据子记账资费调用优惠零件进行批价处理。不同的子记账资费的优惠零件模板可能不一样, 但是它们的核心处理过程 `process` 最终都会调用最简单的优惠零件 (`cFav000`) 的处理函数。下面以简单分段优惠零件组合为例, 阐述优惠零件的调用过程, 图 7 为一个简单分段优惠的子记账资费的参数列表。

conditioncode	vec01
thresholdtype	2
itfavorotype	8
gtfavorotype	8
billrebillflag	0
gtunits	60
thresholdval	3
ltunits	180
ltfavoroca	200
feetypes	10000000
units	60
gtfavoroap	100

图 7 简单分段优惠的子记账资费的参数

(1) 创建优惠零件 `cFavBase*p=new cFav001 <cFav002 <cFav000, cFav000>>`: 创建零件时, `cFav002 <cFav000, cFav000>` 将被实例化作为 `cFav001` 的成员 `T*pFavBase`; 同时 `cFav000` 和 `cFav000` 分别实例化为 `T1*pltFavBase` (小于等于阈值 `ThresholdVal` 用到的具体的优惠零件) 和 `T2*pgtFavBase` (大于阈值 `ThresholdVal` 用到的具体的优惠零件)。

(2) 参数初始化 `p->initPara()`: 从子记账资费参数中读出 `conditionCode` 和 `bill_rebill_flag`, 然后调用 `cFav002 <cFav000, cFav000>(pFavBase)` 的初始化操作, 读出其中的 `ThresholdVal`、`Thresholdtype`、`feetype` 和 `unit`, 同时调用两个 `cFav000(pltFavBase` 和 `pgtFavBase)` 的初始化操作, 将前缀 `lt` 和 `gt` 的参数项分别赋值给 `pltFavBase` 和 `pgtFavBase`。

(3) 优惠过程 `p->process()`: 首先获取话单的相关条件信息 (如通话类型, 漫游类型, 对端类型, 费用类型等) 和标批二批标识, 然后与子记账中 `conditionCode` 限制的

相关条件信息进行匹配, 若匹配失败则返回 0; 若成功, 继续调用 `cFav002 <cFav000, cFav000>(pFavBase)` 的优惠过程, 根据话单中的计费量与阈值的关系, 最终调用 `cFav000` 的优惠过程, 即将 `feetype`、`value`、`unit`、`favortype` 等值传递给费用对象 `feeObj`, 让其负责最后费用的计算。

本文通过对移动计费系统批价优惠业务的分析, 提出了一种基于语法树的优惠方法并使用面向对象语言 C++ 的模板类设计了语法树的解析和调用过程, 构建优惠语法树实现多套餐的叠加优惠, 调用配置好的优惠零件使得程序具有更高的灵活性。该方法实现了各种复杂的优惠逻辑组合, 可以支撑灵活的市场营销策略, 提高了计费系统的支撑能力, 在理论和实践上给出了一个电信行业计费系统应用中的解决方案。

参考文献

(下转第 94 页)

- [1] 陈龙, 张春红, 云亮, 等. 电信运营支撑系统[M]. 北京: 人民邮电出版社, 2007.
- [2] 倪然. 融合计费: 全业务运营的必由之路[J]. 通讯世界, 2008(11): 88.
- [3] 刘全. 内存数据库在帐务后台中的应用[D]. 南京: 南京理工大学, 2004.
- [4] LAFORE R. C++面向对象程序设计 [M]. 邓子梁译. 北京: 中国电力出版社, 2004.

(收稿日期: 2011-06-12)

作者简介:

王爱平, 女, 1956 年生, 教授, 主要研究方向: 数据库与软件技术。