

支持功能可重构的 ERP 开发平台中的构件技术

胡嘉贤,常会友,衣 杨

(中山大学 信息科学与技术学院,广东 广州 510275)

(hujiaxian@sina.com)

摘 要:研究了支持功能可重构的企业资源计划(ERP)开发平台中的构件的析取、管理、部署以及运行的关键技术,包括构件的定义、描述、分类以及组装方式等。该方法与典型的 J2EE 多层体系架构方法相比,简化并加速了 ERP 系统的开发和实施过程,实现了基于构件的可重构 ERP,提升了 ERP 系统的可靠性、可复用性和可移植性。

关键词:企业资源计划开发平台;ERP 构件;引擎服务;消息总线

中图分类号: TP391 **文献标识码:** A

Components technology of reconfiguration supported ERP developing platform

HU Jia-xian, CHANG Hui-you, YI Yang

(School of Information Science and Technology, Sun Yat-sen University, Guangzhou Guangdong 510275, China)

Abstract: Some important key techniques of the extraction, management, deployment and running of the components in the reconfiguration supported ERP(Enterprise Resource Planning) developing platform were studied, which including the definition of ERP components' specification, classification and integrating methods. Compared with the classical J2EE architecture, the proposed method predigested and accelerated the development and application process of ERP system, implemented the reconfiguration based on components, and made the system more reliable, reusable, and transplantable.

Key words: ERP developing platform; ERP components; engine service; message bus

0 引言

占世界软件销售额 85% 的是大型的专用软件,而其开发的失败率却高达 70%。同其他大型应用软件一样,ERP 的开发方式上仍然存在着开发量大、开发周期长、变化多等突出问题,生产方式更多地是基于代码级的重复、类的重复,严重影响着开发的效率和质量^[1]。近年来,为了从根本上提升软件的开发、发布和维护效率,真正实现企业信息化过程中的用户参与、快速开发、快速应用、灵活调整,大幅度提升管理系统实施和应用的成功率及投入产出比,“平台+构件”逐渐成为 ERP 的主要生产模式。

构件是软件系统中具有相对独立功能、可以明确辨识、接口由契约指定、语境有明显依赖关系、可独立部署、可组装的软件实体,并且可以重复使用^[2]。构件技术就是一种类似于“零部件组装”集成组装式的软件生产方式^[3]。它把零件、生产线和装配运行的概念运用在软件产业中。文献[4]提出了一种构件的分类体系以及构件实体属性的描述方法,并应用实体语义相似性系数和领域语义字典实现对构件的检索;文献[5]提出了支持接口连接式和插头插座式体系结构类型的构件模型以及基于体系结构的构件集成组装框架;文献[6]提出了一种开放式软件总线结构,并设计了在此总线上的构件集成与组装方法。上述研究为基于构件的 ERP 开发平台提供理论与方法的支持,但缺乏对 ERP 开发过程的针对性。本文结合 ERP 系统开发中的实际问题,研究设计了支持功能

可重构的 ERP 开发平台中的构件对象的实现、管理、部署以及运行的关键技术:定义了构件的范围及构件的规格说明;确定了构件的类型及其在分层架构中的位置;并设计了通过引擎和消息总线完成构件组装的方式,为 ERP 系统的快速开发提供了一种有效的方法。

1 ERP 开发平台中构件的析取和管理

1.1 ERP 开发平台中构件的结构

构件模型以学术界普遍认同的具有指导性作用的 3C 模型^[7]为基础、采用 XML 描述构件,如图 1 所示。

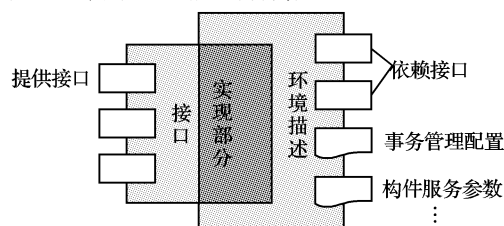


图 1 构件的结构

从总体上,构件分为 4 个组成部分:

构件基本信息 包括构件名称、构件类型、作者、版本等。其中构件名称是构件的唯一标识;系统在标识构件时,有一个相应的注册版本号,该编号用于构件的版本管理。

构件提供接口信息 包括接口名称、接口参数等,其操作规约如下:

Operation = Name + Parameter + Return +

收稿日期:2006-01-24;修订日期:2006-04-06

基金项目:国家自然科学基金资助项目(60573159);广东省自然科学基金重点项目(05200302)

作者简介:胡嘉贤(1982-),女,广东潮安人,硕士研究生,主要研究方向:可重构 ERP 系统研究、Web Service、J2EE;常会友(1962-),男,河北宣化人,教授,博士,主要研究方向:计算机智能、决策支持、体系结构、数据库、嵌入式系统;衣杨(1967-),女,辽宁沈阳人,副教授,博士,主要研究方向:智能控制理论和算法、复杂系统建模与优化、基于知识系统的非确定信息的管理和处理。

Exception + PortFacet

Name 说明操作的名称;Parameter 说明操作所需的参数及其类型;Return 说明操作返回值类型;Exception 说明操作的异常情况;PortFacet 是对操作的侧面描述,帮助用户理解操作。

构件实现 描述构件的具体实现,包括构件的内部关系和构件的实现体。前者定义构件是否为复合构件,以及组成复合构件的子构件的名称。后者描述构件的实现体的位置以及实现体的类型及其要素,包括文件大小、数目、保存地、容器类型、构件部署运行的支撑环境、事务处理、安全选项等。通过实现体描述,系统可以将同一构件描述与不同的实现体绑定起来。

构件环境描述 描述构件的依赖条件和运行环境,主要包括依赖接口名称、依赖接口参数、使用前置条件和后置条件、构件服务参数等。在构件抽取的时候,构件中加入了参数化属性,用户通过对构件的这些属性进行配置就可以使构件适应于不同的环境。参数化属性使得构件的通用性和灵活性得到了很大的提高,适应了环境的需求。

1.2 ERP 开发平台中构件的分类

ERP 开发平台中的构件按功能结构可以分为页面构件、展现构件、业务构件、运算构件和数据构件5类。其中,页面构件和展现构件位于J2EE多层体系架构中的表现层,业务构件和运算构件位于业务逻辑层,而数据构件位于数据层。

页面构件 用于完成用户交互界面生成及页面集成,可以是标准J2EE表现层的构件,如JSP、Tag等;它依赖于展现构件,并提供一组页面展现服务。

展现构件 用于控制业务流转与页面交互,通过ERP开发平台开发的图形化组装业务构件和页面构件,来表达一个完整的功能;它依赖于业务构件、页面构件和展现构件自身,并提供功能级服务接口,可以完成一组完整的功能操作。

业务构件 用于完成企业某一业务功能(如单据处理),是通过ERP开发平台开发的、由多个运算逻辑组成的图形化逻辑流程;它依赖于运算构件与业务构件,并提供对特定业务进行操作的业务服务,比如库存管理类,提供对仓库库存进行管理的多个业务服务接口。

运算构件 ERP开发平台中最底层的构件,用于完成特定的业务计算和程序集成,包含的运算逻辑是标准的Java Method;它不依赖于其他构件,提供各类型的底层运算服务。

数据构件 用于完成数据模型定义,管理数据实体映射关系,是底层构件,无依赖接口,并以XML数据格式提供接口。

1.3 ERP 开发平台中构件析取和封装

构件的抽取和封装是实现可重构的关键,具体的实现步骤如下:

1) 分解。基于系统的业务功能,自顶向下,按照如下顺序对ERP系统进行分解:系统→子系统→功能模块→具体功能→构件。

2) 分类。基于J2EE多层体系架构,将构件划分为:页面构件、展现构件、业务构件、运算构件和数据构件。

3) 描述。基于构件的定义,从如下方面对构件进行描述:构件的概念、接口、操作规约、实现体、构件体、构件复合、构件注释、构件语境等。

4) 封装。基于构件的分类,运用不同的方式(如编写源码、图形化建模等),完成构件实现体的生产和封装。

以可重构ERP销售子系统为例说明上述过程,见图2,用XML描述构件模型,表1详细列出了销售管理子系统所包含

的构件及其实现方法。

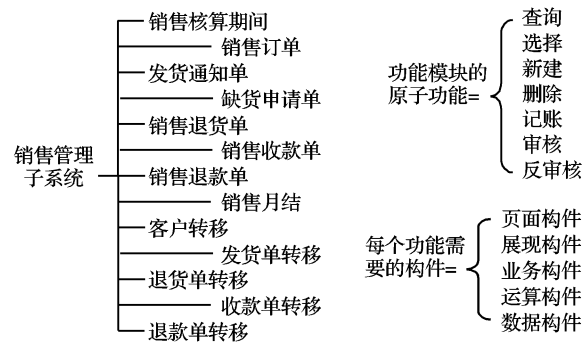


图2 销售管理子系统包含的功能模块

表1 销售管理子系统构件列表

| 构件 | 构件列表 |
|------|---|
| 页面构件 | 各个功能模块的查询列表页面、编辑页面、明细列表页面、明细编辑页面(如: 订单列表页面 SALOrderList.jsp, 订单编辑页面 SALOrderEdit.jsp) |
| 展现构件 | 业务流转与页面交互的图形化描述,如: 订单查询中调用查询业务构件完成查询功能,并通过页面显示结果 |
| 业务构件 | 完成一个具体的业务功能,由多个运算构件组成的图形化描述,如: 订单查询 |
| 运算构件 | 标准Java方法,完成新增、删除、修改、查询等底层操作,以及计算金额、汇总数量、审核、反审核、记账、反记账、核销等运算 |
| 数据构件 | 各种单据对应的数据库表 |

2 ERP 开发平台中构件的部署和运行

通过构件间的连接通信实现构件的组装,实现ERP系统的快速开发。

2.1 ERP 开发平台中构件的通信和互操作

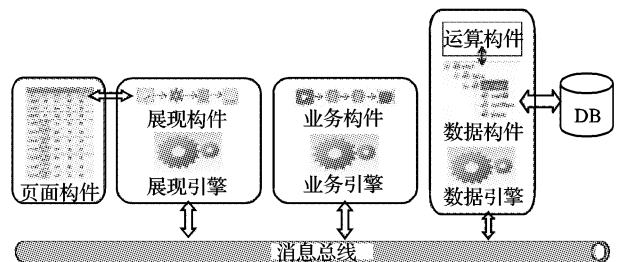


图3 引擎服务层运行原理

通信和互操作是可重构ERP系统的关键问题之一,本文设计了引擎服务层,分别在展现、业务和数据层加入了相应的引擎,由引擎完成大部分重复性工作,如数据提取和提取、构件描述的解析、生成消息等,然后通过消息总线实现构件之间的连接通信,进而形成整个ERP系统。当提供服务的构件接口发生变化或替换构件时,不需检查系统中每个构件的实

现细节、无需调用语句或消息参数做出修改,只要在引擎中进行统一的转换修改即可。引擎服务的原理如下:

- 1) 展现引擎接受用户请求,确定要调用的展现构件并解析执行,根据展现逻辑流程中定义的业务逻辑,传入业务引擎;
- 2) 业务引擎根据调用的业务逻辑中的流程调用相应的运算构件;
- 3) 对于需要访问数据库的操作通过数据引擎根据数据映射访问相关的数据库;
- 4) 业务引擎处理完业务逻辑后返回相关数据给展现层;
- 5) 展现引擎根据展现构件逻辑定义控制页面流转,把数据传送到页面上,在页面中可以使用丰富的标签库展现数据。

其中,在各个引擎之间通过消息总线来进行通信,消息是构件之间通信的唯一方式。

2.2 ERP 开发平台中构件的部署和运行

通过消息总线进行构件组装。消息总线也称为连接件 (Connector),负责消息的分派、传递和过滤以及处理结果的返回,如图 4 所示。其步骤如下:

- 1) 初始化。在系统开始运行之前,对 Connector 的运行环境进行初始化,并将构件注册信息写入 Connector 环境中,再将构件、描述文件及 Connector、Connector 环境部署在应用服务器上。
- 2) 构件-消息响应登记表。各个构件挂接在 Connector 上,向 Connector 登记感兴趣的消息类型,形成构件-消息响应登记表。
- 3) 解析构件。系统运行时,引擎解析构件,并需要根据发出消息,由 Connector 根据接收到的消息类型和构件-消息响应登记表的信息,定位并传递该消息给相应的响应者。
- 4) 响应。构件接收到消息后,根据自身状态对消息进行响应,并通过 Connector 返回处理结果。其中,请求消息要求符合统一接口规范,消息内封装了服务请求必需的信息,例如:响应构件名称、同步方式还是异步方式、方法名称、参数类型、参数值等。

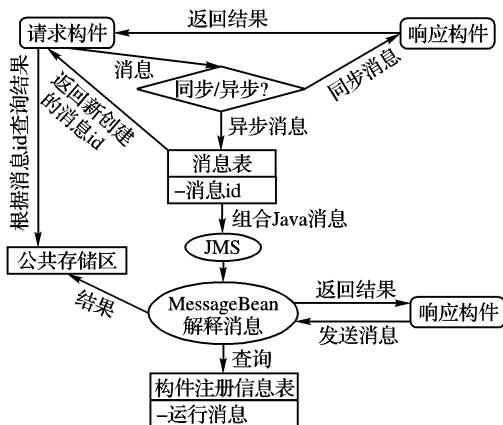


图4 消息总线运行原理

Connector 对消息的处理有两种方式:同步消息和异步消息。Connector 根据请求消息中的异步标识进行判断。若消息是同步的,Connector 直接将消息发送到响应构件,让请求和响应构件直接相联。响应构件处理完成后,结果直接返回请求构件;若是异步消息,Connector 接收到请求消息后,自动产生一个 Java 消息,并将 Java 消息交给 JMS (Java Message Service) 放进相应的消息队列里面。JMS 是访问企业消息系统的标准 API,它便于消息系统中的 Java 应用程序进行消息交换,并且通过提供标准的产生、发送、接收消息的接口简化企业应用的开发。

下面以查询销售订单功能为例,说明构件的运行过程:

- 1) 依照构件模型的定义,用 XML 编写构件描述文件,包括:名称、接口、是否符合构件、实现体描述、运行环境等。
- 2) 编写查询所有销售订单数据库表并返回查询结果的 Java 方法,构成查询运算构件的实现体。
- 3) 使用图形化编辑器开发业务构件的实现体,其流程连接顺序为“开始”→“查询”→“返回结果”→“结束”,每一个节点都定义了其输入输出参数。
- 4) 将构件注册在 Connector 上,并登记其感兴趣的消息。
- 5) 运行时,引擎对业务构件进行解析,发现需要调用查询运算构件,随即根据它的构件描述文件生成相应的请求消息,发送给 Connector;Connector 再将该消息传递给运算构件并处理返回结果。

6) 功能裁减。当 ERP 系统应用于不同的企业时,将根据企业的经营模型、生产方式以及操作方法等实际的需求做适应性的功能修改,使其适应该企业的实际需求:

在构件级别上 通过修改构件注册表里的信息来实现一个构件的新增、替换或者删除。

在操作级别上 通过对构件描述文件的提供接口部分进行裁剪来实现,对于那些没有在描述文件中描述的服务,用户就无法使用到了。通过这种方式,用户不仅可以选择构件,还可以选择构件中的功能。

3 结语

本文研究了可重构 ERP 系统开发平台构件的相关技术,提出了构件的析取、管理、通信和运行的技术方案,实现了基于构件的可重构 ERP 的组装。

本文的技术创新集中在如下两个方面:1) 通过引擎服务层,对构件进行解析,统一完成页面和业务的流转控制,使一般 ERP 系统中需要编写几百行代码的展现构件和业务构件得以用图形化的方式进行开发。一方面大大提高了代码重用率,减少开发工作量;另一方面,增强了页面和业务流转的灵活性,可以方便地进行修改。2) 引擎解析构件后,通过消息总线完成构件之间的连接通信,进而构成完整的 ERP 系统。消息是构件间通信的唯一途径,消息的组装基于构件的描述文件,从而可实现构件的功能裁减,适应 ERP 系统的功能变化。

表2 支持可重构 ERP 开发平台与典型 J2EE 架构开发的 ERP 系统

| 比较项目 | 典型 J2EE 多层体系架构 | 支持功能可重构的 ERP 开发平台 |
|-------------|----------------|-------------------|
| 子系统数 | 16 | 16 |
| 模块数 | 172 | 172 |
| 文件数 | 1 624 | 1 968 |
| 系统开发时间/(人天) | 433 | 352 |
| 可复用模块数 | 50 | 116 |
| 再开发支持率(%) | 29 | 67 |
| 系统实施时间/(人天) | 227 | 110 |
| 系统维护时间/(人天) | 50 | 30 |

从 2002 年至今我们曾经分别采用典型的基于 J2EE 多层体系架构及支持功能可重构的 ERP 开发平台,开发了 ERP 系统。表 2 是以某卷烟厂 ERP 为背景的对比情况。可见,采用基于可重构 ERP 开发平台,开发效率提高了 20%,实施和维护的工作量减少约 50%。更加重要的是,此 ERP 系统支持功能裁减和流程重组;为二次、三次……的再开发提供了大量可复用的模块和代码。

(下转第 1743 页)

这两部分可以同时配置,也可以分开来配置。

动态可重构协处理器的通用数据寄存器组主要由8个8位寄存器队列和相关输出网络组成,每个寄存器队列的长度为4,它们两两为一组分别与4条全局的存储器总线相连。在控制器控制下,作为输入队列时,根据定时器设置,寄存器队列每若干周期向下移一位;作为输出队列时,队列每若干周期向上移一位,直到队列满。此外每次主处理器从队列中取数后或向队列中存数前都将清空该队列。

5 实验与结果

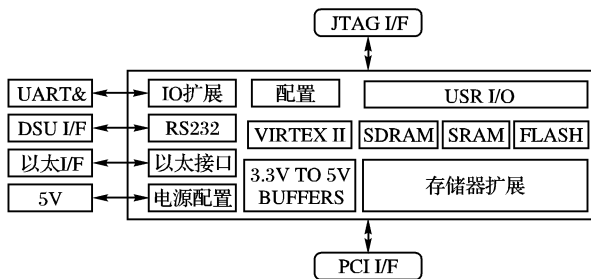


图5 GR-PCI-XC2V 结构

我们使用Gaisler公司的GR-PCI-XC2V实验开发板作为原型样机的实验板^[6,7],结构如图5,考查动态可重构协处理器在FFT中的应用。按频率抽取的基2FFT算法原理如下:

$$A' = A + C$$

$$A' = (A - C) \cdot W_N^k$$

对于8位的实数运算而言,一个以基2FFT的蝶需处理单元40个,其中乘法器32个,加法器4个,减法器4个,配置时间40周期,处理时序如图6。

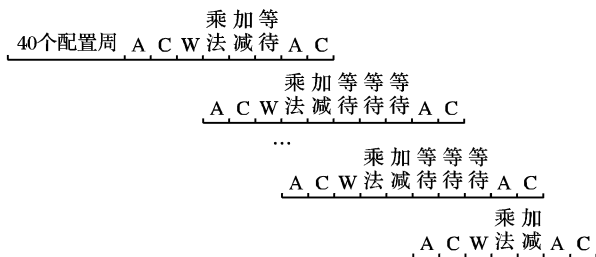


图6 FFT时序

图7绘出了25M时钟频率下动态可重构协处理器、LEON2、Intel 80386 SX嵌入式处理器和33.3MHz下ADSP 21020四者进行8位无符号乘法运算的时间曲线图。与通用CPU和DSP的乘法运算时间曲线从零点开始随运算次数呈线性增长不同,动态可重构协处理器的运算时间曲线不从零点开始,且呈折线增长。这是因为协处理器中每32个处理单元可配置为1个8位阵列式乘法器,每个乘法器的配置时间约为 $32 \times 40\text{ns}$,每次乘法运算时间都在一周期(40ns)内,且下一个乘法器的配置可以与前面乘法器的运算并行处理,因此,动态可重构协处理器乘法运算时间曲线

在最初阶段有一个1280ns的乘法器配置阶段,此后每过大约32周期,可重构阵列可增加一乘法器,实现并行计算,直到资源用完。图中分别显示了 8×8 可重构阵列大小的原型样机进行计算的**实际性能曲线(只能配置为两个乘法器)和扩大至 8×16 阵列大小的理论计算曲线,可以看出资源对系统性能的影响。

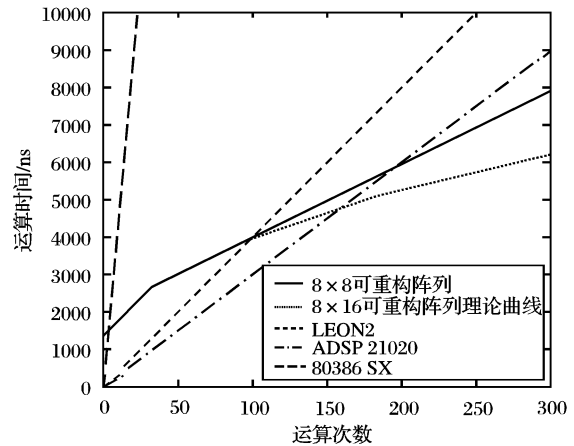


图7 乘法性能曲线

从上面的实验结果我们可以看出,动态可重构协处理器原型样机具有以下特点:一是灵活性,通过设置配置寄存器,同一可重构阵列可以动态配置为加、减、乘、乘加、FFT等不同运算逻辑;二是利于实现循环计算,可重构阵列配置完成后其运算过程和普通ASIC相似,但由于运算逻辑的配置时间往往超过运算时间,因此在单次运算中使用可重构计算并不能带来性能的提高,可重构计算应当用于处理有规律的运算,此时系统配置的时间开销被多次重用所分担,从而大大提高系统性能。

参考文献:

- [1] BONDALAPATI K, PRASANNA VK. Reconfigurable computing systems[J]. Proceedings of the IEEE, 2002, 90(7): 1201-1217.
- [2] GAISLER J. The LEON-2 processor user's manual[DB/OL]. <http://www.gaisler.com/doc/leon2-1.0.23-xst.pdf>, 2004-09-08.
- [3] BREBNER G, LEVI D. Networking on chip with platform FPGAs[A]. IEEE International Conference on Field-Programmable Technology[C]. Tokyo, Japanese, 2003. 13-20.
- [4] SLADE AL, NELSON B, HUTCHINGS B. Reconfigurable computing application frameworks[A]. Proceedings of the IEEE Symposium on Field-Programmable Custom Computing Machines[C]. California, 2003. 251-262.
- [5] HAUSER JR. Augmenting a microprocessor with reconfigurable hardware[D]. UC Berkeley, 2000.
- [6] Xilinx. Xilinx ISE 6 software manuals and help[DB/OL]. <http://toolbox.xilinx.com/docsan/xilinx6/books/manuals.pdf>, 2004-08-31.
- [7] Xilinx. Virtex-II Platform FPGA User Guide[DB/OL]. <http://direct.xilinx.com/bvdocs/userguides/ug002.pdf>, 2004-06-03.

(上接第1740页)

参考文献:

- [1] 周昌盛, 王小军, 李金. 基于组件技术的ERP系统的设计与实现[J]. 微计算机应用, 2005, 26(2): 204-206.
- [2] 杨美清. 构件技术引领软件开发新潮流[J]. 中国计算机用户, 2005, (6): 13.
- [3] CAPRETZ LF, CAPRETZ MAM, DAHAI L. Component-based software development[A]. IECON'01: The 27th Annual Conference of the IEEE[C]. Industrial Electronics Society, 2001, 3: 1834-1837.

- [4] 吉卫喜, 梁素勤. 软件构件的管理、标识与检索技术[J]. 计算机应用, 2003, 23(9): 67-75.
- [5] 张世琨, 张文娟, 常欣, 等. 基于软件体系结构的可复用构件制作和组装[J]. 软件学报, 2001, 12(9): 1351-1359.
- [6] 袁占亭, 张秋余, 张冬冬, 等. 基于软件总线技术的软件开发[J]. 计算机工程, 2005, 31(1): 105-107.
- [7] 姚万军, 李永刚. 可复用构件模型探讨[J]. 情报指挥控制系统与仿真技术, 2004, 26(2): 65-71.
- [8] 罗鸿. ERP原理, 设计, 实施[M]. 北京: 电子工业出版社, 2002.