

工作流过程定义规范

数据类型

14

计算机科学2000Vol. 27No. 11

工作流元模型

工作流过程定义规范

The Workflow Specification of Process Definition

53-56

缪晓阳 石文俊 吴朝晖

TP399

(浙江大学计算机系统工程研究所 浙江大学计算中心 杭州310027)

Abstract This paper discusses the representation of a business process in a form. There are three basic aspects: the concept of workflow process definition, on which the idea of process definition interchange is raised; the meta-model of workflow, which is used to describe the entities and attributes of entities within the process definition; the workflow process definition language (WPDL), which is used to implement the process definition.

Keywords Workflow process definition, Process definition interchange, Workflow meta-model, Workflow process model, WPDL

1 引言

工作流过程定义规范定义了一个通用的交换格式,对业务流程进行形式化的说明,这种通用的交换格式,可支持不同产品间过程定义的传送。规范也内在地包括了开发环境和运行环境形式上的分离,这使得由建模工具生成的过程定义,能作为许多其它 WFMS 运行时产品的输入,符合了用户所要求的建模和产品运行的独立性。

为了提供通用的访问和描述过程定义的方法,建立了工作流过程定义的元模型,元模型确定了过程定义中通用的实体,通过一组属性描述了这个有限实体集的特征,正是基于这种模型,不同的厂商之间才能够以通用的格式完成过程定义的传送。

工作流过程定义语言(WPDL)的关键元素之一是对由不同厂商使用的信息的扩展性。WPDL 不可能支持每一个厂商要求的每一个附加信息。

2 过程定义概述

2.1 过程定义概念

过程定义(Process Definition):业务流程的形式化描述,用来支持系统建模和运行过程的自动化。过程定义由活动(activity)及这些活动之间的关系,过程开始条件、过程结束条件,以及一些关于个体行为的信息,如组织成员,与 IT 相关的应用和数据等组成(WFMC Glossary-WFMC-TC-1011)。

过程定义经工作流引擎解释,用作产生过程实例的模板,并控制过程实例的运行。一个过程可以分解成可独立定义的子过程。创建、交换和修改过程定义的能力对任何工作流系统都是至关重要的。

一个过程定义至少应包含完成初始化及运行过程

所需的实体和属性的最小集,创建一个过程实例将继续承其中的一部分实体和属性,在 WAPI(WFMC Workflow Application Programming Interface 参阅 WFMC-TC-1009)中,通过 API 来操作这些过程实体及其属性。

2.2 过程定义交换概念

为什么要引入过程定义交换(Process Definition Interchange)?主要有以下几个原因:

- 把定义过程和执行过程分开,使用户能够为系统的每个部分使用最优的产品。

- 通过一个公用设计池交换和检索过程定义,接受不同工具的大量访问,以及运行时系统,都要求只使用一种格式实现过程定义的控制,以便于过程定义的管理。

- 过程执行中,被授权用户要求修改过程定义。

- 过程执行中,为了增加不同工作流引擎间的互操作性,过程定义从一个工作流引擎传送到另一个工作流引擎是必要的。

为了满足上述要求,工作流过程定义规范定义了一个公用交换格式,支持不同产品间过程定义的交换。这样就使得开发环境和运行环境彼此独立——由开发时工具建立的过程定义,可以为不同的运行时 WFMC 产品所解释。

2.3 过程定义交换实现

目前,过程定义交换主要通过以下两种机制的组合实现:

(1)使用过程定义语言(WPDL)定义过程:WPDL 是一种公共交换的标准。工作引擎内部表示的过程定义经引出操作,映射成用 WPDL 描述的文件,这些文件可被其它工作引擎引入而映射成该工作引擎表示的过程定义。

(2) 对象关系模型方式用于支持 API 进行查询、访问和修改过程定义的单个元素;使用这种方式时,相关产品之间必须用到底层的通信接口,比如基于 RPC 和 ORB 机制的接口。

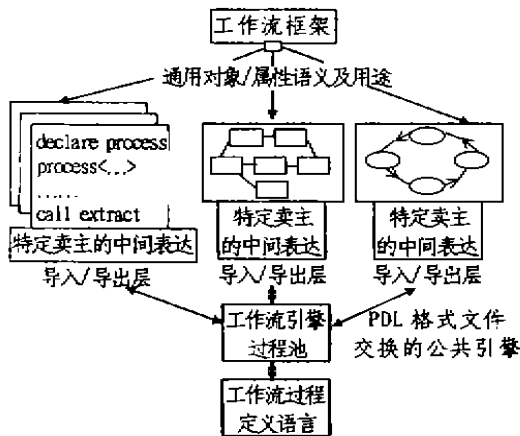


图1 过程定义交换

正如图1所示,在元模型框架内,不同的厂商提供过程定义的中间表达形式,通过使用过程定义语言(WPDL),开发时工具建立的过程定义,在不同的 workflow 运行时产品中得到翻译、执行。

3 工作流元模型

工作流元模型(Workflow Meta-Model)描述了工作流过程定义包含的实体,实体的属性,实体与实体之间的关系,以及怎样把这些过程定义归并成过程模型和共用数据。(见图2)

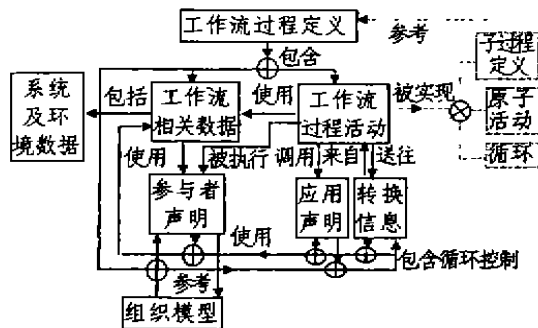


图2 工作流元模型

3.1 实体概述

顶层实体主要包括:

- 过程定义:描述过程本身的重要信息,如过程的名字和 ID 号。
- 过程活动:描述组成过程的一系列逻辑步骤——

任务,过程可根据粒度的大小不同划分为子过程和原子行为两类,后者一般是不能再次分解,可提交给用户手工执行或直接调用应用程序自动完成。

·转换信息:描述不同过程行为之间的跃迁,包含顺序和并行操作;不同的行为之间通过转换信息相互联系。

·参与者声明:参照企业组织模型,定义行为的执行者。执行者不一定单指一个人,有可能是机器设备或其他组织实体。

·组织模型:在复杂的情况下,参与者声明可能会参考过程定义之外的组织模型,以计算一些复杂表达式,如表达式中引用了业务功能,组织实体及关系等。

·应用声明:在过程描述中,一个行为通常会对应到一个或者多个应用,这些应用将会在运行期间被 workflow 核心系统所调用。

·过程相关数据:行为决策、应用调用、任务转换中所涉及的数据信息,有可能会影响行为的下一步执行。

·系统和环境数据:该类数据由 WFMS 或本地系统环境维护,为 workflow 活动访问或者由 WFMS 使用(按处理过程相关数据同样的方式,算出表达式的值,可认为是过程相关数据的扩充)。

3.2 属性概述

首先,我们来看一下 WPDL 中实体和属性概貌(图3)。值得注意的是,黑体字表示必需的属性,非黑体字表示可选的属性,斜体字表示与其他实体的关系,“...”表示待用的扩展属性。

大致上,属性可以分为七组来介绍。首先,每个实体都应有一个独一无二的 Id,为了便于理解和使用也使用 Name 和 Description 属性。第二组是刻画个别实体的特定属性,如创建日期,版本,作者,字符集等。第三组是过程/模型相关数据中使用的参数或值,这组属性也包含转换信息中使用的判断条件。第四组用于引用其它实体,第五组是执行引擎要用的文档和图标信息,第六组用于模拟和处理优化。最后是待用的扩展属性。

3.3 过程定义,过程模型与过程池

在单个过程定义中,可能并不需要用到元模型包含实体的所有属性,而过程相关数据、应用、参与者的定义又可能为大量的过程定义所引用,因而有必要提供一种机制,能够方便和高效地进行过程定义。元模型假设使用一个与工作流管理系统(WFMS)相连的过程定义池,容纳构成过程定义的各种实体类型。出于池本身的需要,以及能与池进行过程定义的高效传送,引入了工作流过程模型的概念,作为从大量不同的过程定义中归并公用数据实体的容器,避免了单个过程定义中的重复定义。

工作流 模型定义	工作流 过程定义	工程流 过程活动	转换信息	工作流 应用声明	工作流 相关数据	工作流参 与者声明
__ Id __ Name __ Description	__ Id __ Name __ Description	__ Id __ Name __ Description	__ Id __ Name __ Description	__ Id __ Name __ Description	__ Id __ Name __ Description	__ Id __ Name __ Description
__ WPDl 版本 __ 卖主 Id __ 创建日期 __ 版本 __ 作者 __ 字符集 __ 代码页 __ 国家代码 __ 扩展库 __ 一致性类	__ 创建日期 __ 版本 __ 作者 __ 字符集 __ 代码页 __ 国家代码 __ 合法日期 __ 扩展库	__ 自动化模式 __ 分裂(split) __ 联结(join) __ 循环(loop) __ 内嵌块		__ 工具名称	__ 类型	__ 类型 __ 类型相关信息
	__ 参数		__ 条件 __ 来源 __ 目的	__ 参数	__ 值	
__ 可靠性 __ 外部模式引 用 __ 访问权限	__ 可靠性 __ 访问权限	__ 参与者分派 __ 实现 __ 应用分派 __ 访问权限				
__ 文档 __ 图标 __ 优先级单位	__ 文档 __ 图标 __ 优先级	__ 文档 __ 图标 __ 优先级 __ 实例化 __ 持续时间 __ 代价 __ 等待时间 __ 工作时间				__ 策略 __ 容量 __ 代价 __ 准备时间

图3 工作流实体的属性

过程模型框架内包含了大量来自过程定义实体中的通用属性,模型中的每一个过程定义都将自动继承获得这些通用属性,除非过程定义中又进行重新说明。

在工作流过程模型中,应注意以下两点:

(1)参与者说明,应用声明,相关数据是全局实体,能为模型中的所有过程定义所引用。

(2)模型引用了两个外部对象类型——逻辑模型和物理组织模型,逻辑模型使得过程模型和它控制的对象能够引用外部模型中的顶层实体。

4 过程定义语言(WPDl)

WPDl 语言使用扩充巴科斯范式(EBNF)的文法描述,过程的定义由一组产生式完成,每个产生式由以下部分构成:表示实体名字和属性的符号,如:〈Exam-

ple_Model〉,〈Example_Workflow〉,〈Example_Activity〉等;产生式符号(::=);引入对象,属性和关系的关键字如 MODEL, WORKFLOW, ACTIVITY, TRANSITION 等;有特殊含义的字符如“|”, “[”, “]”, “/”, “/*”等。

4.1 WPDl 数据类型

WPDl 的数据类型可以分为三个层次:基本数据类型,简单数据类型,复杂数据类型。基本数据类型包括字符串(string),浮点(float),整型(integer),引用(reference),时间日期(date),以字母或下划线开头的字母数字组合(identifier),即:

```

<basic_data> ::= (string) | (float) |
                (integer) | (reference) |
                (date) | (identifier);
    
```

在基本数据类型的基础上形成了简单的数据类型,其形式如下:

```
<plain_data> ::= {basic_data}
                {boolean} {perform};
```

其中,boolean 是布尔类型,perform 是已声明参与者类型。

复杂的数据类型则允许在扩展属性或过程定义相关数据内有数组,纪录,枚举,列表的定义,如:

```
<complex_data> ::=
    <plain_data>

    {<RECORD> (<Member_list>) <END>}
        //a record type
    {<ARRAY> (<OpenArrayB>) <cardinal>}
    {<UPTO> (<cardinal>) <CloseArrayB>}
    {<OF> (<complex_data_type>)
        //an array type
    {<ENUM> (<element list>) <END>}
        //an enumeration type;
    {<LIST> (<OF>) (<element type>)
        //a list data type
```

4.2 WPD 通用表示格式

在 WPD 中,工作流模型本身作为一个元模型实体,是元模型内其它实体描述的序列。实体描述的某些部分被归并到一个 WPD 包中,如过程定义头文件(model definition header),达到编制文档结构化的要求。每个实体的描述都是由实体关键字、标志、属性列表、实体结束关键字的序列构成,模型实体在实体结束关键字前还可能附加元模型实体描述列表,

在属性中,除了预先定义的属性列表外,通过扩展属性,允许工作流过程定义产品的生产厂商为元模型内的实体定义他们自己的属性集。工作流过程定义语言(WPD)的关键元素之一是对由不同厂商使用的信息的扩展性,WPD 不可能支持每一个厂商要求的每一个附加信息,基于描述过程定义的工作流模型,WPD 能支持大量不同的解决方案。

```
<extended attribute list>
 ::= EXTENDED_ATTRIBUTE
    {attribute id}
    {attribute type}
    {attribute value}
    [{description}] //optional
```

限于篇幅,最后以通用的格式给出的工作流工程定义

的 WPD 实现只是一个概念上的实现,但读者应该能有个大致的理解了。

```
<Workflow Model> ::=
MODEL (<model id>)
    {Workflow Model Definition Header}
    [{Workflow Participant Specification}]
    [{Workflow Application List}]
    [{Workflow Relevant Data List}]
    {Workflow Process Definition},
END_MODEL
<Workflow Process Definition> ::=
WORKFLOW (<process id>)
    {Workflow Process Definition Header}
    [{Access restriction part}]
    {Activity List}
    {Transition Information List}
    [{Workflow Participant Specification}]
    [{Workflow Application List}]
<Activity List> ::=
ACTIVITY (<activity id>)
    [NAME (<name>)]
    [DESCRIPTION (<description>)]
    {Activity Kind Information}
    [{Access Restriction part}]
    [{Transition Restriction part}]
END_ACTIVITY
[Activity List]
<Transition Information List> ::=
TRANSITION (<transition id>)
    [NAME (<name>)]
    {transition kind description}
    [DESCRIPTION (<description>)]
END_TRANSITION
[Transition Information List]
:
```

图4 WPD 文法

小结 本文对工作流过程定义规范的介绍,还是相当简要的,一些重要的方面,如对过程定义引入/引出接口的整个结构及与相连的工作流服务的关系,我们并没有作介绍。随着进一步的研究,还会有其它的实体、属性、关系加入到元模型中来。

参考文献

- 1 The Workflow Management Coalition Specification. Interface 1: Process Definition Interchange & Process Model. Document Number WFMCTC-1016-P
- 2 OMG. OMG CORBA Services, Common Object Services Specification. OMG T Document Number 95-3-31, Object Management Group, Framingham, MA, 1995