

中华人民共和国国家标准

GB/T 32392.5—2018/ISO/IEC 19763-5:2015

信息技术 互操作性元模型框架(MFI) 第5部分:过程模型注册元模型

Information technology—Metamodel framework for interoperability (MFI)—
Part 5: Metamodel for process model registration

(ISO/IEC 19763-5:2015, IDT)

2018-03-15 发布

2018-10-01 实施

中华人民共和国国家质量监督检验检疫总局 发布
中国国家标准化管理委员会

目 次

前言	I
引言	II
1 范围	1
2 规范性引用文件	1
3 术语、定义和缩略语.....	2
3.1 术语和定义	2
3.2 缩略语	4
4 符合性	4
4.1 概要	4
4.2 符合程度	4
4.3 实现符合性声明	5
5 MFI 过程模型注册元模型的结构	5
5.1 MFI 过程模型注册元模型概论	5
5.2 本部分与 MFI 其他部分的关系	6
5.3 过程模型注册元模型中的元类	7
附录 A (资料性附录) 过程模型注册元模型的注册案例	15
附录 B (资料性附录) 过程建模语言列表	26
参考文献	27

前 言

GB/T 32392《信息技术 互操作性元模型框架(MFI)》分为以下几个部分：

- 第 1 部分：参考模型；
- 第 2 部分：核心模型；
- 第 3 部分：本体注册元模型；
- 第 4 部分：模型映射元模型；
- 第 5 部分：过程模型注册元模型；
- 第 7 部分：服务模型注册元模型；
- 第 8 部分：角色和目标模型注册元模型；
- 第 9 部分：按需模型选择。

本部分为 GB/T 32392 的第 5 部分。

本部分按照 GB/T 1.1—2009 给出的规则起草。

本部分使用翻译法等同采用 ISO/IEC 19763-5:2015《信息技术 互操作性元模型框架(MFI) 第 5 部分：过程模型注册元模型》。

与本部分中规范性引用的国际文件有一致性对应关系的我国文件如下：

- GB/T 32392.1—2015 信息技术 互操作性元模型框架(MFI) 第 1 部分：参考模型(ISO/IEC 19763-1:2007, IDT)
- GB/T 32392.7—2018 信息技术 互操作性元模型框架(MFI) 第 7 部分：服务模型注册元模型(ISO/IEC 19763-7:2015, IDT)
- GB/T 32392.8—2018 信息技术 互操作性元模型框架(MFI) 第 8 部分：角色和目标模型注册元模型(ISO/IEC 19763-8:2015, IDT)

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本部分由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本部分起草单位：武汉大学、中国电子技术标准化研究院、金蝶软件(中国)有限公司。

本部分主要起草人：何克清、王翀、王健、冯在文、何非、王静、胡博。

引 言

在全球经济一体化的环境下,企业在简化产品开发和交付以及降低运营成本方面所面临的经济压力推动了业务过程协作与集成的发展。企业可以通过合并和结成合作伙伴的方式来解决这些问题。因此,需要为过程模型提供标准的注册库,使其能够被发现、理解和比较以支持过程的重用和集成,进而促进企业内和企业之间的互操作。

业务过程建模语言和符号体系已经被广泛用于对过程进行表示。然而,这些过程模型在语法和语义上存在较大差异,阻碍了业务过程在企业间的共享和重用。因此,需要提供一种通用的注册机制,用于对过程模型的管理信息和核心元数据进行注册。

本部分定义了一个元模型,用于注册过程模型的部分元数据和语义信息,以支持业务过程的发现和重用。本部分为过程模型提供了公共语义信息,帮助人们了解过程的结构和过程之间关联关系,辅助完成过程的发现,而无需关注业务过程所采用的原始建模符号。本部分不考虑任何与过程建模语法以及过程执行平台相关的信息。特别是,尽管过程模型的注册信息可用于进一步支持基于过程和 Web 服务之间关联的服务发现,表示 Web 服务内部执行序列和服务组合编排信息的业务过程不属于本部分的研究范畴。

注:在本部分中,“过程”特指“业务过程”,“过程模型”特指“业务过程模型”。

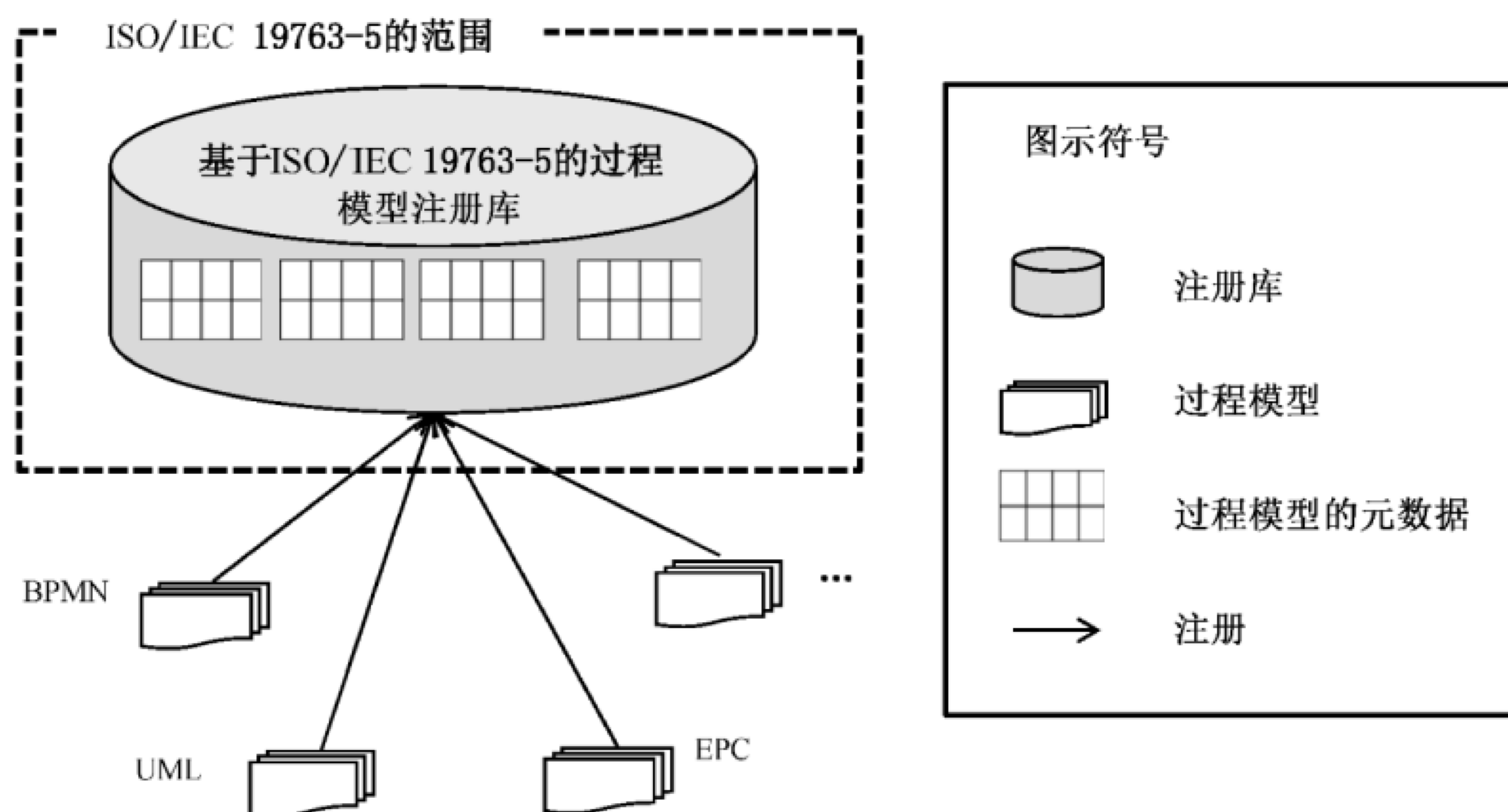
信息技术 互操作性元模型框架(MFI)

第5部分:过程模型注册元模型

1 范围

ISO/IEC 19763 的主要目标是构建支持互操作的元模型框架。

本部分提出了一个元模型,作为过程模型管理信息和部分核心元数据的注册机制。本部分定义的元模型主要用于促进过程模型库中以及过程模型库之间过程模型的语义发现和重用。本部分为使用特定过程建模语言[如 BPMN(业务过程建模符号体系)、UML(统一建模语言)活动图、EPC(事件驱动的业务过程链)等]创建的过程模型提供了核心元数据和公共语义,定义的元模型能够帮助发现过程的功能和组成,促进不同粒度的过程组件的重用。图 1 描述了本部分的研究范围。



注: 在注册之前,模型不需要存储于注册库中。

图 1 本部分的范围

下述内容超出了本部分的范围:

- 与过程模型建模符号或描述语言相关的细节;
- 过程执行的运行环境或实现平台。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件,仅注日期的版本适用于本文件。凡是不注日期的引用文件,其最新版本(包括所有的修改单)适用于本文件。

ISO/IEC 19763-1 信息技术 互操作性元模型框架 第1部分:框架(Information technology—Metamodel framework for interoperability (MFI)—Part 1:Framework)

ISO/IEC 19763-7 信息技术 互操作性元模型框架 第7部分:服务模型注册元模型(Information technology—Metamodel framework for interoperability (MFI)—Part 7:Metamodel for

service model registration)

ISO/IEC 19763-8 信息技术 互操作性元模型框架 第 8 部分:角色和目标模型注册元模型(Information technology—Metamodel framework for interoperability (MFI)—Part 8:Metamodel for role and goal model registration)

ISO/IEC 19763-10 信息技术 互操作性元模型框架 第 10 部分:核心模型和基本映射(Information technology—Metamodel framework for interoperability (MFI)—Part 10: Core model and basic mapping)

3 术语、定义和缩略语

3.1 术语和定义

ISO/IEC 19763-7、ISO/IEC 19763-8 和 ISO/IEC 19763-10 界定的以及下列术语和定义适用于本文件。

3.1.1

活动 activity

一组衔接的任务(3.1.21)。

3.1.2

控制约束 control constraint

对给定过程(3.1.12)集合执行顺序的约束。

3.1.3

依赖 dependency

过程模型元素(3.1.14)之间的关联关系,指定了控制约束(3.1.2)。

3.1.4

事件 event

特定环境下的发生。

3.1.5

退出条件 exit condition

如果为真,将导致操作非正常结束的约束。

注:操作可以是过程或服务操作。

3.1.6

目标 goal

用户与过程(3.1.12)或服务(3.1.18)交互时预期得到的结果。

3.1.7

门户条件 guard condition

在相关联的过程(3.1.12)执行前应满足的条件。

3.1.8

参与类型 involvement type

一种声明,指明角色在参与过程(3.1.12)或服务(3.1.18)时的类型。

示例:执行者、受益者、客户。

3.1.9

合并依赖 join dependency

一种依赖,当指定的前驱过程模型元素(3.1.14)完成后,后继的过程模型元素(3.1.14)将开始执行。

3.1.10

后置条件 postcondition

操作完成时应为真的约束。

注：操作可以是过程或服务操作。

3.1.11

前置条件 precondition

操作被调用前应为真的约束。

注：操作可以是过程或服务操作。

3.1.12

过程 process

一组结构化和相互关联的**活动**(3.1.1)或**任务**(3.1.21),用于实现一个特定的**目标**(3.1.6)。

注：本部分中,活动和任务均由过程元类表示。

3.1.13

过程模型 process model

使用一种特定**过程建模语言**(3.1.15)得到的**过程**(3.1.12)的表示。

3.1.14

过程模型元素 process model element

组成一个**过程**(3.1.12)的建模结构的抽象表示,包括**过程**(3.1.12)和**过程之间的依赖**(3.1.3)。

3.1.15

过程建模语言 process modelling language

用于表示**过程**(3.1.12)的特定语言。

注 1：PSL、BPMN、UML 活动图等都属于过程建模语言。

注 2：特定语言(ISO 1087-1:2000, 3.1.3)。

3.1.16

资源 resource

过程模型元素(3.1.14)使用、创建和消耗的资产。

3.1.17

角色 role

参与到某一情境中的实体所承担的特定职责。

3.1.18

服务 service

一种封装了一个或多个计算模块的应用,可以通过特定的接口进行访问。

3.1.19

顺序依赖 sequence dependency

过程(3.1.12)之间的一种控制结构,用来指定**过程**(3.1.12)将被依次执行。

3.1.20

分支依赖 split dependency

过程模型元素(3.1.14)之间的一种控制结构。当前置**过程模型元素**(3.1.14)完成后,一个或多个后续**过程模型元素**(3.1.14)将并发执行。

3.1.21

任务 task

需要完成的工作件。

3.1.22

元类 **metaclass**

实例是类的类。

3.1.23

抽象元类 **abstract metaclass**

不能直接被实例化的元类(3.1.22)。

3.2 缩略语

下列缩略语适用于本文件。

BPMN:业务过程模型和记法(Business Process Model and Notation)

EPC:事件驱动过程链(Event-driven Process Chain)

MFI:互操作性元模型框架(Metamodel Framework for Interoperability)

OWL-S:面向服务的网络本体描述语言(Web Ontology Language for Service)

PSL:过程规约语言(Process Specification Language)

UML:统一建模语言(Unified Modeling Language)

4 符合性

4.1 概要

声称与本部分相符合的实现应支持第5章中规定的元模型,该元模型依赖下列描述的符合程度。

4.2 符合程度

4.2.1 概论

为了同时满足互操作性和可扩展性的需要,应区别“严格符合”和“符合”实现。本部分描述了促进互操作性的规范。扩展是根据用户、开发商、组织和行业的需要而展开的,本部分没有对其进行规定。

严格符合实现的实用性有限,但是能够最大程度地实现与本部分相关的互操作。符合实现的实用性更广,但实现与本部分相关的互操作的能力较弱。

4.2.2 严格符合实现

严格符合实现:

- a) 应支持第5章中规定的元模型;
- b) 不应支持对第5章中规定的元模型实施的任何扩展,包括使用、测试、访问或调查。

4.2.3 符合实现

符合实现:

- a) 应支持第5章中规定的元模型;
- b) 可以支持与5.3中规定的元模型相一致的扩展,包括使用、测试、访问或调查。

注1:所有的严格符合实现都是符合实现。

注2:对元模型进行扩展可能导致未定义的行为。

4.3 实现符合性声明

声称本部分的实现应该包括陈述如下的实现符合声明：

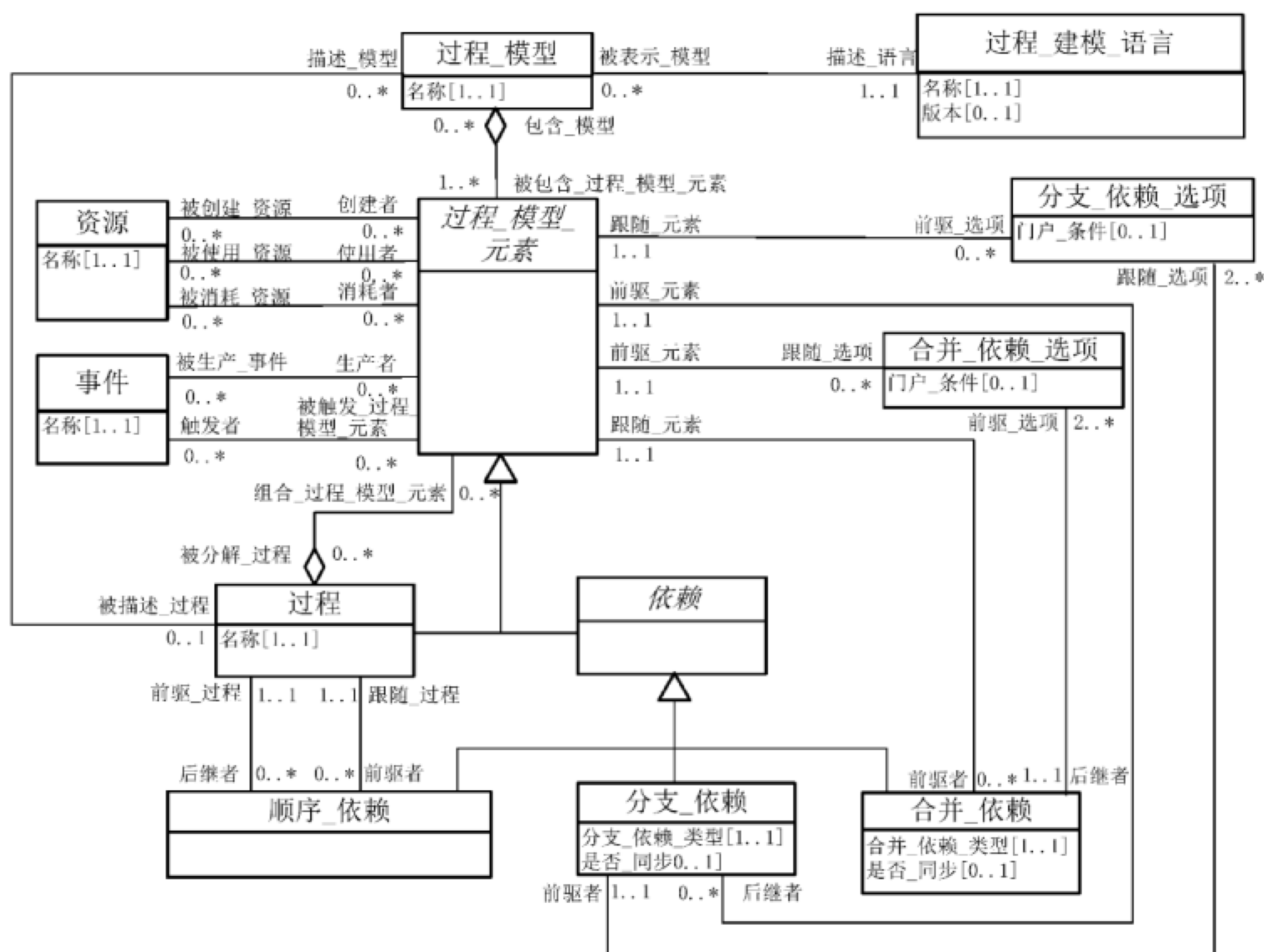
- a) 是严格符合实现(4.2.2)还是符合实现(4.2.3)；
- b) 如果是符合实现,则支持和使用了哪些扩展。

5 MFI 过程模型注册元模型的结构

5.1 MFI 过程模型注册元模型概论

MFI 过程模型注册元模型定义了一个通用的元模型,用于注册使用特定建模语言描述的过程模型的部分元数据。图 2 展示了过程模型注册的元模型。

在本部分中,过程模型是过程的表示,它使用一种特定的过程建模语言描述了所包含的过程模型元素。过程模型元素包括过程以及过程或其他过程模型元素之间的依赖。对每一个过程模型元素而言,存在用于触发一个过程模型元素的事件或者由一个过程模型元素产生的事件。过程模型元素需要使用、创建或者消耗资源以实现既定的目标。



注：名称为斜体的元类为抽象元类。

图 2 过程模型注册的元模型

依赖描述了过程模型所描述的过程之间的控制约束。在本部分中,依赖可以是顺序依赖、分支依赖或合并依赖。顺序依赖指定过程为依次执行。分支依赖规定,当前驱过程模型元素完成后,一个或多个

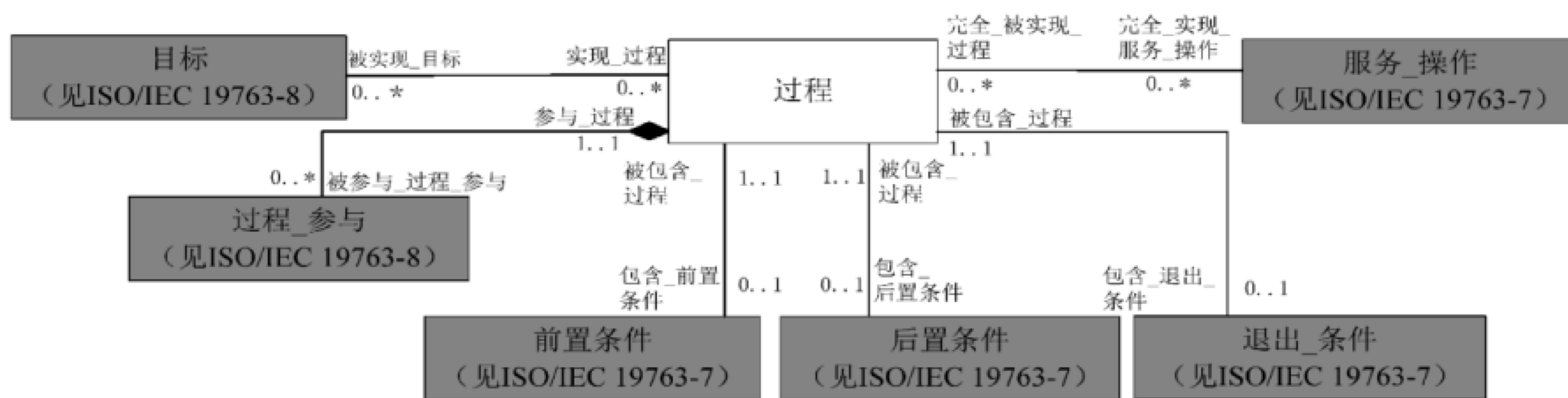
后继过程模型元素将并发执行。合并依赖规定,当选定的前驱过程模型元素完成后,后继过程模型元素将开始执行。在分支依赖中,分支依赖的类型可为后续过程指定一种逻辑门。类似地,在合并依赖中,合并依赖的类型可为前驱过程指定逻辑门。本部分中,分支依赖类型和合并依赖类型均可以是“异或”、“或”和“与”。对分支依赖类型而言,“异或”表示有且仅有一个后继过程模型元素可以被执行,“或”表示一个或多个后继过程模型元素可以被执行,“与”表示所有的后继过程模型元素都应被执行。对合并依赖类型而言,“异或”表示有且仅有一个前驱过程模型元素成功完成后,后继过程模型元素才可以执行;“或”表示当一个或多个前驱过程模型元素成功完成后,后继过程模型元素才可以执行;“与”表示当且仅当所有的前驱过程模型元素成功完成后,后继过程模型元素才可以执行。此外,分支依赖选项表示分支依赖类型确定后,将被执行的后续过程模型元素的门户条件。类似地,合并依赖选项表示合并依赖类型确定后,将被执行的前驱过程模型元素的门户条件。

过程模型注册元模型的注册示例参见附录 A,建议的过程模型建模语言参见附录 B。

5.2 本部分与 MFI 其他部分的关系

在本部分中,MFI 的其他部分是指 ISO/IEC 19763-7、ISO/IEC 19763-8 和 ISO/IEC 19763-10。

图 3 描述了 ISO/IEC 19763-5、ISO/IEC 19763-7 和 ISO/IEC 19763-8 之间的关系。



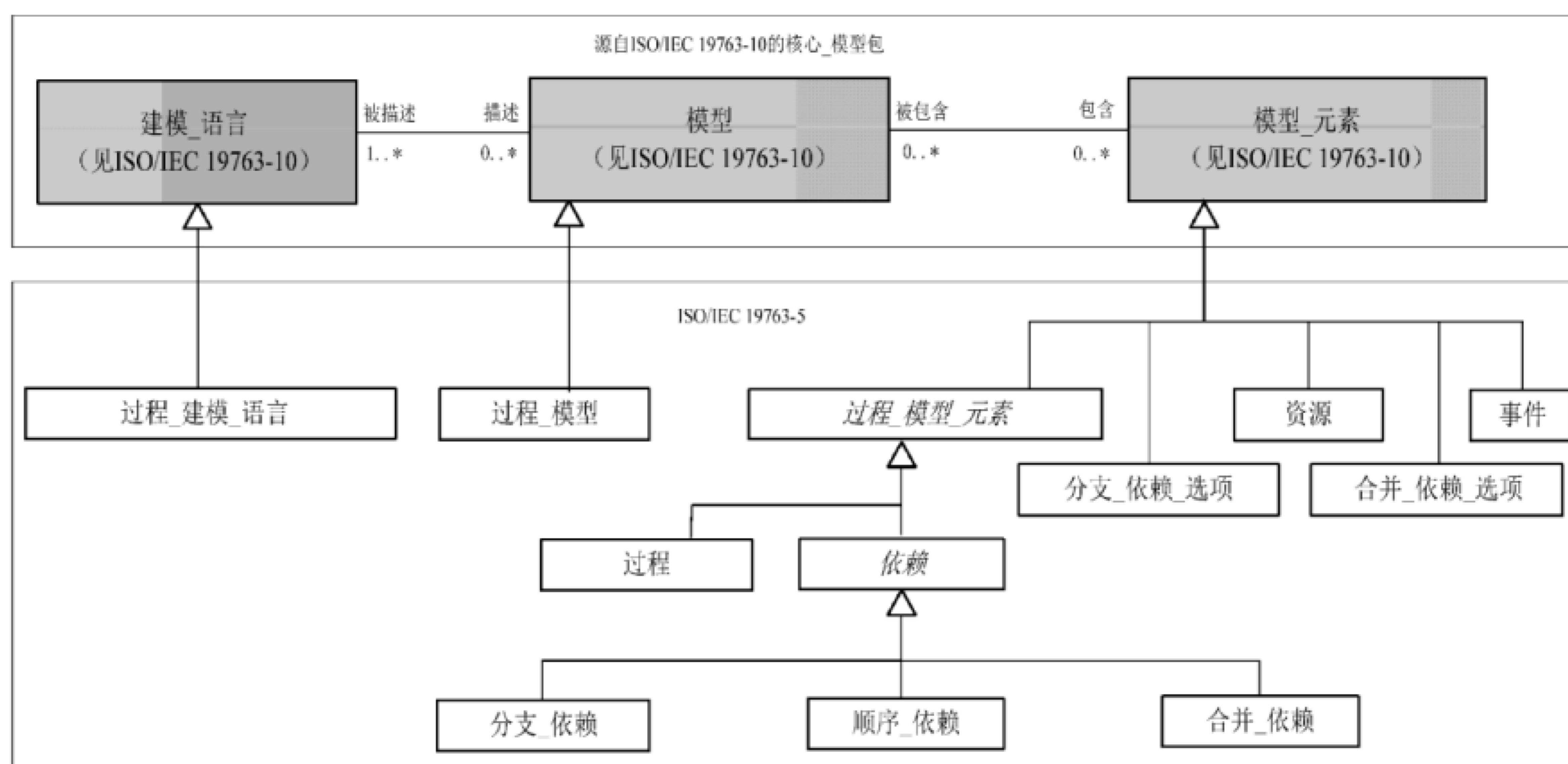
注：灰色底色表示的元类是在本标准其他部分中定义的元类。

图 3 ISO/IEC 19763-5 与 ISO/IEC 19763-7 和 ISO/IEC 19763-8 之间的关系

ISO/IEC 19763-5 与 ISO/IEC 19763-8 之间的关系指定了每个过程可以实现零个或一个目标,每个目标可以被零个、一个或多个过程实现。存在没有指定被某个特定过程实现的目标,存在没有被应用于实现特定目标的过程。类似地,每个过程可以参与零个、一个或多个过程参与,其中每个过程参与是指参与过程的角色,例如参与者或受益者。每个过程参与指定一个角色可以参与当且仅当一个过程。一个过程参与只能拥有一个相关联的过程。

ISO/IEC 19763-5 与 ISO/IEC 19763-7 之间的关系指定了每个过程可以被零个、一个或多个服务操作完全实现,每个服务操作可以完全实现零个、一个或多个过程。存在没有指定被某个特定服务实现的过程,存在没有被应用于实现某个特定过程的服务。每个过程可能拥有一个前置条件或者一个后置条件。存在没有前置条件或后置条件的过程。每个过程拥有零个或一个退出条件,用于声明在过程完成前造成过程中止的一组条件。每个前置条件、后置条件和退出条件可以通过组合表达或原子表达来定义。

图 4 描述了 ISO/IEC 19763-5 中元类与 ISO/IEC 19763-10 中元类之间的关系。



注 1：名称为斜体的元类是抽象元类。

注 2：灰色底色表示的元类是在 MFI 其他部分中定义的元类。

图 4 ISO/IEC 19763-5 与 ISO/IEC 19763-10 之间的关系

过程模型注册元模型中的过程_建模_语言元类是 ISO/IEC 19763-10 中建模_语言元类的子类。过程模型注册元模型中过程_模型元类是 ISO/IEC 19763-10 中模型元类的子类。ISO/IEC 19763-5 中的其他元类均为 ISO/IEC 19763-10 中模型_元素元类的子类。

所有子类都拥有从其超类继承得到的关联。一些继承的关联在本部分中进行了特化,具体化的细节定义见 5.3。

5.3 过程模型注册元模型中的元类

5.3.1 依赖

依赖是一个抽象元类,它的每个实例表示一个指定的特定依赖。

超类

无

属性

数据类型

多重度

描述

无

参考

类

多重度

描述

反向参考

优先

无

约束

无

5.3.2 事件

事件是一个元类,它的每个实例表示一个特定的事件。

超类

模型_元素(见 ISO/IEC 19763-10)

属性	数据类型	多重度	描述	反向参考	优先
名称	串型	1..1	事件实例的名称。		
参考	类	多重度	描述	反向参考	优先
被触发_过程_模型_元素	过程_模型_元素	0..*	一组过程模型元素。其中的每个过程模型元素由这个事件触发。	触发	是
产生者	过程_模型_元素	0..*	一组过程模型元素。其中每个过程模型元素均可以产生这个事件。	被产生_事件	是

约束

无

5.3.3 合并_依赖

合并_依赖是一个元类,它的每个实例表示一个特定的合并依赖。

超类

依赖

属性	数据类型	多重度	描述	反向参考	优先
合并_依赖_类型	串型	1..1	指定合并依赖是与合并依赖、或合并依赖还是异或合并依赖的声明。“异或”表示有且仅有一个前驱过程成功完成后,后继过程才会执行。“或”表示当一个或多个前驱过程成功完成后,后继过程才会执行。“与”表示当且仅当所有的前驱过程成功完成后,后继过程才会执行。		
是否_同步	布尔型	0..1	表示被合并的过程模型元素是否应同步执行的。真值表示同步,假值表示不同步。		
参考	类	多重度	描述	反向参考	优先
前驱_选项	合并_依赖_选项	2..*	一组合并依赖选项。其中的每个合并依赖选项指定了一个门户条件,用于决定被关联的过程模型元素是否将通过合并依赖选项与其他过程模型元素一起被关联到这个合并依赖。	后继者	否
后继_元素	过程_模型_元素	1..1	跟随这个合并依赖的过程模型元素。	前驱者	否

约束

无

5.3.4 合并_依赖_选项

合并_依赖_选项是一个元类,它的每个实例表示一个特定的合并依赖选项。

超类

模型_元素(见 ISO/IEC 19763-10)

属性	数据类型	多重度	描述	反向参考	优先
门户_条件	串型	0..1	条件规范,对指定的前驱过程模型元素而言应为真。这个前驱过程模型元素通过关联到相同合并依赖的其他合并依赖选项与其他的过程模型元素进行关联。		
后继者	类	多重度	描述	反向参考	优先
	合并_依赖	1..1	与合并依赖选项相关联的合并依赖,这个合并依赖选项为一个相关联的后继过程模型元素指定了门户条件。	后继_选项	是
前驱_元素	过程_模型_元素	1..1	在这个合并依赖选项之前的过程模型元素。	前驱_选项	否

约束

无

5.3.5 过程

超类

过程_模型_元素

属性	数据类型	多重度	描述	反向参考	优先
名称	串型	1..1	过程实例的名称。		
描述_模型	类	多重度	描述	反向参考	优先
	过程_模型	0..*	一组过程模型,每个过程模型可能描述这个过程的构成。	被描述_过程	否
组成_过程_模型_元素	过程_模型_元素	0..*	一组过程模型元素,包括过程和过程之间的依赖,每个过程模型元素组成了一个过程。	被分解_过程	否
前驱者	顺序_依赖	0..*	一组顺序依赖,每个顺序依赖表示这个过程跟随着一个作为特定顺序依赖的前驱过程的过程。	后继_过程	否
后继者	顺序_依赖	0..*	一组顺序依赖,每个顺序依赖表示这个过程先于一个作为特定顺序依赖的后继过程的过程。	前驱_过程	否

完全_实现_服务_操作	服务_操作(见 ISO/IEC 19763-7)	0..*	一组服务操作,每个服务操作可以完全实现这个过程。一个过程可以被一组服务操作完全实现,意味着这个过程应能够被分解到一定的层次,使得这个过程的每一个子过程都能够被一个服务操作完全实现。	完全_被实现_过程	否
被参与_过程_参与	过程_参与(见 ISO/IEC 19763-8)	0..*	一组过程参与,每个过程参与表示一个特定的角色如何参与到这个过程中。	参与_过程	是
被完成_目标	目标(见 ISO/IEC 19763-8)	0..1	被这个过程完成的目标。	完成_过程	是
包含_前置条件	前置条件(见 ISO/IEC 19763-7)	0..1	这个过程被调用时应为真的约束。	被包含_过程	是
包含_后置条件	后置条件(见 ISO/IEC 19763-7)	0..1	这个过程完成时应为真的约束。	被包含_过程	是
包含_退出_条件	退出_条件(见 ISO/IEC 19763-7)	0..1	约束,为真时这个过程将在完成前终止。	被包含_过程	否

约束
无

5.3.6 过程_模型

过程_模型是一个元类,它的每个实例表示一个特定的过程模型。

超类

模型(见 ISO/IEC 19763-10)

属性	数据类型	多重度	描述	反向参考	优先
名称	串型	1..1	这个过程模型实例的名称。		
参考	类	多重度	描述	反向参考	优先
描述_语言	过程_建模_语言	1..1	用于描述这个过程模型的过程建模语言。本参考特化了从超类继承的“被描述”参考。	被表示_模型	是
被包含_过程_模型_元素	过程_模型_元素	1..*	在这个过程模型中描述的一组过程模型元素。本参考特化了从超类继承的“包含”参考。	包含_模型	是
被描述_过程	过程	0..1	由这个过程模型描述其分解的过程。	描述_模型	是

约束
无

5.3.7 过程_模型_元素

超类

模型(见 ISO/IEC 19763-10)

属性	数据类型	多重度	描述	反向参考	优先
无					
参考	类	多重度	描述	反向参考	优先
包含_模型	过程_模型	0..*	包含这个过程模型元素的一组过程模型。本参考特化了从超类继承的“被包含”参考。	被包含_过程_模型_元素	否
被分解_过程	过程	0..*	分解到这个过程模型元素的一组过程。	组成_过程_模型_元素	是
被创建_资源	资源	0..*	一组资源,其中每个资源可以由这个过程模型元素创建。	创建者	是
被消耗_资源	资源	0..*	一组资源,其中每个资源可以被这个过程模型元素消耗。	消耗者	是
被使用_资源	资源	0..*	一组资源,其中每个资源可以在这个过程模型元素的执行中被使用。	使用者	是
被产生_事件	事件	0..*	一组事件,其中每个事件可以由这个过程模型元素产生。	产生者	是
触发者	事件	0..*	一组事件,其中每个事件可能触发或激活这个过程模型元素。	被触发_过程_模型_元素	否
后继者	分支_依赖	0..*	一组分支依赖,其中每个适合的分支依赖将跟随这个过程模型元素。	前驱_元素	是
前驱_选项	分支_依赖_选项	0..*	一组分支依赖选项,其中每个分支依赖选项指定了一个门户条件,用于判定这个过程模型元素是否被执行。	后继_元素	是
前驱者	合并_依赖	0..*	一组合并依赖,其中每个合适的合并依赖将位于这个过程模型元素之前。	后继_元素	是
后继_选项	合并_依赖_选项	0..*	一组合并依赖选项,其中每个合并依赖选项指定了一个门户条件,用于判断这个过程模型元素是否将与合并依赖中的其他过程模型元素合并。	前驱_元素	是

约束

无

5.3.8 过程_建模_语言

过程_建模_语言是一个元类,它的每个实例表示一种具体的过程建模语言。

超类

建模_语言(见 ISO/IEC 19763-10)

属性	数据类型	多重度	描述	反向参考	优先
名称	串型	1..1	过程建模语言实例的名称。		
版本	值	0..1	过程建模语言实例的版本。		
参考	类	多重度	描述	反向参考	优先
被表示_模型	过程_模型	0..*	使用这个过程建模语言表示的一组过程模型。本参考特化了从超类继承的“描述”参考。	描述_语言	否

约束

无

5.3.9 资源

资源是一个元类,它的每个实例表示一个具体的资源。

超类

模型_元素(见 ISO/IEC 19763-10)

属性	数据类型	多重度	描述	反向参考	优先
名称	串型	1..1	资源实例的名称。		
参考	类	多重度	描述	反向参考	优先
消耗者	过程_模型_元素	0..*	一组过程,其中每个过程能够在过程模型元素的执行时消耗这个资源。	被消耗_资源	否
创建者	过程_模型_元素	0..*	一组过程,其中每个过程能够在过程模型元素的执行时创建这个资源。	被创建_资源	否
使用者	过程_模型_元素	0..*	一组过程,其中每个过程能够在过程模型元素的执行时使用这个资源。	被使用_资源	否

约束

“消耗者”“创建者”和“使用者”参考的值对于任何一个过程模型元素都是相对唯一的,即:一个过程模型元素可以消耗、创建或使用一个具体的资源。

5.3.10 顺序_依赖

顺序_依赖是一个元类,它的每个实例标示一个具体的顺序依赖。

超类

依赖

属性	数据类型	多重度	描述	反向参考	优先
无					
参考	类	多重度	描述	反向参考	优先
后继_过程	过程	1..1	跟随这个顺序依赖的过程。	前驱者	是
前驱_过程	过程	1..1	在这个顺序依赖之前的过程。	后继者	是
约束					
无					

5.3.11 分支_依赖

分支_依赖是一个元类,它的每个实例标示一个具体的分支依赖。

超类

依赖

属性	数据类型	多重度	描述	反向参考	优先
分支_依赖_类型	串型	1..1	指定分支依赖是与分支依赖、或分支依赖还是异或合并依赖的声明。“异或”表示有且仅有一个后继过程模型元素可以被执行。“或”表示一个或多个后继过程模型元素可以被执行。“与”表示所有的后继过程模型元素都应执行。		
是否_同步	布尔型	0..1	表示被分开的过程模型元素是否应同步执行。真值表示同步,假值表示不同步。		
参考	类	多重度	描述	反向参考	优先
后继_选项	分支_依赖_选项	2..*	一组分支依赖选项。其中的每个分支选项指定了一个门户条件,用于决定被关联的过程模型元素是否被执行。	前驱者	否
前驱_元素	过程_模型_元素	1..1	在这个分支依赖之前的过程模型元素。	后继者	否
约束					
无					

5.3.12 分支_依赖_选项

分支_依赖_选项是一个元类,它的每个实例表示一个特定的分支依赖选项。

超类

模型_元素(见 ISO/IEC 19763-10)

属性	数据类型	多重度	描述
门户_条件	串型	0..1	条件规范,对将被执行的后继过程模型元素而言应为真。如果没有指定门户条件,这个后继过程模型元素将会被执行。

参考	类	多重度	描述	反向参考	优先
前驱者	分支_依赖	1..1	与分支依赖选项相关联的分支依赖,这个分支依赖选项为一个相关联的后继过程模型元素指定了门户条件。	后继_选项	是
后继_元素	过程_模型_元素	1..1	跟随这个分支依赖选项的过程模型元素。	前驱_选项	否
约束					
无					

附录 A (资料性附录)

过程模型注册元模型的注册案例

本附录将通过四个案例详细说明如何基于过程模型注册元模型对不同类型的过程模型进行注册。具体地,四个案例中的过程模型分别使用 UML 活动图、BPMN、EPC 和 OWL-S 进行描述,对应的转换示范和注册信息充分说明过程模型注册元模型能够用于注册异构的过程模型。同时,这也表示过程模型注册元模型能够与已有的过程建模规范相互协调。

A.1 案例 A.1:基于 UML 活动图的“车票_预订”过程

“车票_预订”过程描述了在线进行车票预定的流程。如图 A.1 所示,车票预订过程使用 UML 活动图进行建模。

车票预订过程包含了一组子过程,这个过程将在客户成功登录后开始执行。首先,客户选择符合他们旅行规划的火车信息。接着,客户需要填写和保存他们的详细预订信息。客户确认预订后,系统会将详细的预订信息自动告知用户,用户可以选择邮件或短信的方式接收系统通知。

为了提升过程模型的重用性,需要对每一个子过程的细节分别进行建模。例如在本案例中,可以使用一个独立的模型来描述“通知_客户”这个过程的分解信息。因此,本案例中注册了两个过程模型:一个是描述整个“车票_预订”过程的过程模型,另一个是描述“通知_客户”子过程的过程模型。

根据 MFI 过程模型注册元模型提供的元模型机制,“通知_客户”过程的过程模型描述了“车票_预订”过程中“通知_客户”子过程的分解信息。“通知_客户”过程模型是在“通知_客户”子过程之后被命名的,所以可以使用两种方式进行过程搜索。

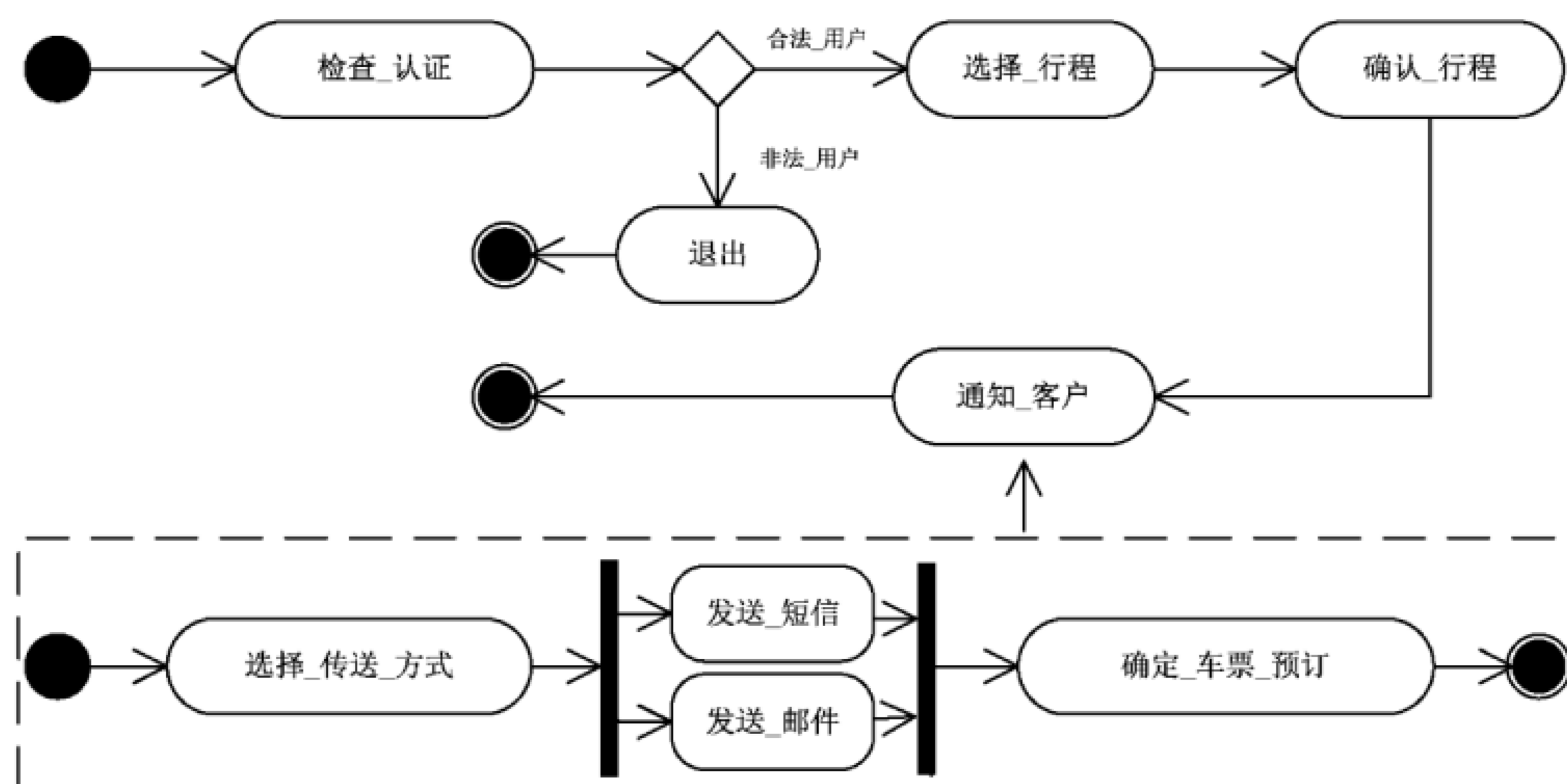


图 A.1 基于 UML 活动图的“车票_预订”过程模型

表 A.1 列出了 MFI 过程模型注册元模型中的哪些元类可以用于表示图 A.1 中“车票_预订”过程模型中的元素。使用这些元类的目的是,将 UML 活动图描述的过程模型(例如“车票_预订”过程模型)注册到基于 MFI 过程模型注册元模型的注册库中。

表 A.1 案例 A.1 的转换关系示例


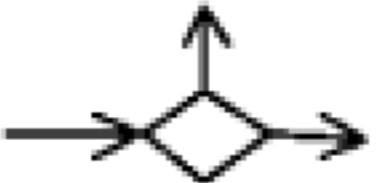

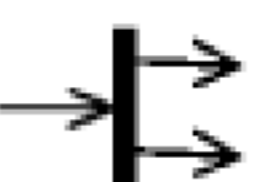
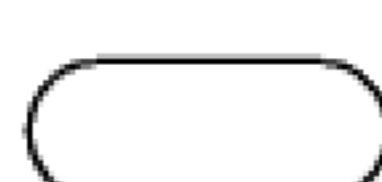


UML 活动图的符号	MFI 过程模型注册元模型中的元类	UML 活动图的符号	MFI 过程模型注册元模型中的元类
	事件		分支_依赖
			
	过程		合并_依赖
			顺序_依赖

图 A.2 和图 A.3 分别展示了用于描述“车票_预订”过程模型(组成父过程的子过程)注册信息和“通知_客户”过程模型(分解子过程的子子过程)注册信息的对象实例。

<p><过程_模型> 对象101</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>名称</td><td>"车票_预订_模型"</td></tr> <tr><td>描述_语言</td><td>对象102</td></tr> <tr><td>被包含_过程_模型_元素</td><td>对象104, 对象105, 对象108, 对象110, 对象111, 对象112, 对象113, 对象114</td></tr> </tbody> </table>	属性/参考	值/实例	名称	"车票_预订_模型"	描述_语言	对象102	被包含_过程_模型_元素	对象104, 对象105, 对象108, 对象110, 对象111, 对象112, 对象113, 对象114	<p><分支_依赖_选项> 对象106</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>门户_条件</td><td>"非法_用户"</td></tr> <tr><td>后继_元素</td><td>对象108</td></tr> <tr><td>先驱者</td><td>对象105</td></tr> </tbody> </table>	属性/参考	值/实例	门户_条件	"非法_用户"	后继_元素	对象108	先驱者	对象105						
属性/参考	值/实例																						
名称	"车票_预订_模型"																						
描述_语言	对象102																						
被包含_过程_模型_元素	对象104, 对象105, 对象108, 对象110, 对象111, 对象112, 对象113, 对象114																						
属性/参考	值/实例																						
门户_条件	"非法_用户"																						
后继_元素	对象108																						
先驱者	对象105																						
<p><过程_建模_语言> 对象102</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>名称</td><td>"UML"</td></tr> <tr><td>版本</td><td>"2.1.2"</td></tr> <tr><td>被表示_模型</td><td>对象101, 对象121</td></tr> </tbody> </table>	属性/参考	值/实例	名称	"UML"	版本	"2.1.2"	被表示_模型	对象101, 对象121	<p><分支_依赖_选项> 对象107</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>门户_条件</td><td>"合法_用户"</td></tr> <tr><td>后继_元素</td><td>对象110</td></tr> <tr><td>先驱者</td><td>对象105</td></tr> </tbody> </table>	属性/参考	值/实例	门户_条件	"合法_用户"	后继_元素	对象110	先驱者	对象105						
属性/参考	值/实例																						
名称	"UML"																						
版本	"2.1.2"																						
被表示_模型	对象101, 对象121																						
属性/参考	值/实例																						
门户_条件	"合法_用户"																						
后继_元素	对象110																						
先驱者	对象105																						
<p><事件> 对象103</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>名称</td><td>"车票_预订_请求"</td></tr> <tr><td>被触发_过程_模型_元素</td><td>对象104</td></tr> </tbody> </table>	属性/参考	值/实例	名称	"车票_预订_请求"	被触发_过程_模型_元素	对象104	<p><过程> 对象108</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>名称</td><td>"退出"</td></tr> <tr><td>包含_模型</td><td>对象101</td></tr> <tr><td>先驱_选项</td><td>对象106</td></tr> <tr><td>被产生_事件</td><td>对象109</td></tr> </tbody> </table>	属性/参考	值/实例	名称	"退出"	包含_模型	对象101	先驱_选项	对象106	被产生_事件	对象109						
属性/参考	值/实例																						
名称	"车票_预订_请求"																						
被触发_过程_模型_元素	对象104																						
属性/参考	值/实例																						
名称	"退出"																						
包含_模型	对象101																						
先驱_选项	对象106																						
被产生_事件	对象109																						
<p><过程> 对象104</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>名称</td><td>"检查_认证"</td></tr> <tr><td>包含_模型</td><td>对象101</td></tr> <tr><td>触发者</td><td>对象103</td></tr> <tr><td>后继者</td><td>对象105</td></tr> </tbody> </table>	属性/参考	值/实例	名称	"检查_认证"	包含_模型	对象101	触发者	对象103	后继者	对象105	<p><事件> 对象109</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>名称</td><td>"用户_退出"</td></tr> <tr><td>产生者</td><td>对象108</td></tr> </tbody> </table>	属性/参考	值/实例	名称	"用户_退出"	产生者	对象108						
属性/参考	值/实例																						
名称	"检查_认证"																						
包含_模型	对象101																						
触发者	对象103																						
后继者	对象105																						
属性/参考	值/实例																						
名称	"用户_退出"																						
产生者	对象108																						
<p><分支_依赖> 对象105</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>分支_依赖_类型</td><td>"异或"</td></tr> <tr><td>是否_同步</td><td>"否"</td></tr> <tr><td>包含_模型</td><td>对象101</td></tr> <tr><td>先驱_元素</td><td>对象104</td></tr> <tr><td>产生者_选项</td><td>对象106,对象107</td></tr> </tbody> </table>	属性/参考	值/实例	分支_依赖_类型	"异或"	是否_同步	"否"	包含_模型	对象101	先驱_元素	对象104	产生者_选项	对象106,对象107	<p><过程> 对象110</p> <table border="1"> <thead> <tr><th>属性/参考</th><th>值/实例</th></tr> </thead> <tbody> <tr><td>名称</td><td>"选择_行程"</td></tr> <tr><td>包含_模型</td><td>对象101</td></tr> <tr><td>先驱_选项</td><td>对象107</td></tr> <tr><td>后继者</td><td>对象111</td></tr> </tbody> </table>	属性/参考	值/实例	名称	"选择_行程"	包含_模型	对象101	先驱_选项	对象107	后继者	对象111
属性/参考	值/实例																						
分支_依赖_类型	"异或"																						
是否_同步	"否"																						
包含_模型	对象101																						
先驱_元素	对象104																						
产生者_选项	对象106,对象107																						
属性/参考	值/实例																						
名称	"选择_行程"																						
包含_模型	对象101																						
先驱_选项	对象107																						
后继者	对象111																						

图 A.2 “车票_预订”过程模型的注册信息(父过程为“车票_预订”过程时)

<顺序_依赖> 对象111	
属性/参考	值/实例
包含_模型	对象101
前驱_过程	对象110
后继_过程	对象112

<过程> 对象112	
属性/参考	值/实例
名称	"确认_行程"
包含_模型	对象101
后继者	对象113
前驱者	对象111

<顺序_依赖> 对象113	
属性/参考	值/实例
包含_模型	对象101
前驱_过程	对象112
后继_过程	对象114

<过程> 对象114	
属性/参考	值/实例
名称	"通知_客户"
包含_模型	对象101
前驱者	对象113
被产生_事件	对象115
描述_模型	对象121

<事件> 对象115	
属性/参考	值/实例
名称	"车票_预订_关闭"
产生者	对象114

图 A.2 (续)

<过程_模型> 对象121	
属性/参考	值/实例
名称	"通知_客户_模型"
描述_语言	对象102
被描述_过程	对象114
被包含_过程_模型_元素	对象123, 对象124, 对象127, 对象128, 对象129, 对象132

<Event> 对象122	
属性/参考	值/实例
名称	"通知_行程_开始"
被触发_过程_模型_元素	对象123

<过程> 对象123	
属性/参考	值/实例
名称	"选择_发送_方式"
包含_模型	对象121
触发者	对象122
后继者	对象124

<分支_依赖> 对象124	
属性/参考	值/实例
分支_依赖_类型	"或"
是否_同步	"假"
包含_模型	对象121
前驱_元素	对象123
后继_选项	对象125, 对象126

<分支_依赖_选项> 对象125	
属性/参考	值/实例
门户_条件	"通过_短信_发送"
后继_元素	对象127
前驱者	对象124

<分支_依赖_选项> 对象126	
属性/参考	值/实例
门户_条件	"通过_短信_发送"
后继_元素	对象128
前驱者	对象124

<过程> 对象127	
属性/参考	值/实例
名称	"发送_邮件"
包含_模型	对象121
前驱_选项	对象125
后继_选项	对象130

<过程> 对象128	
属性/参考	值/实例
名称	"发送_短信"
包含_模型	对象121
前驱_选项	对象126
后继_选项	对象131

<合并_依赖> 对象129	
属性/参考	值/实例
合并_依赖_类型	"或"
是否_同步	"假"
包含_模型	对象121
前驱_选项	对象130, 对象131
后继_元素	对象132

图 A.3 “通知_客户”过程模型的注册信息
(子过程“通知_客户”可以被进一步分解且拥有独立的注册信息时)

<合并_依赖_选项> 对象130		<过程> 对象132	
属性/参考	值/实例	属性/参考	值/实例
门户_条件	"通过_邮件_发送_成功"	名称	"确认_车票_预订"
前驱_元素	对象127	包含_模型	对象121
后继者	对象129	前驱者	对象129
		被产生_事件	对象133

<合并_依赖_选项> 对象131		<事件> 对象133	
属性/参考	值/实例	属性/参考	值/实例
门户_条件	"通过_短信_发送_成功"	名称	"通知_客户_完成"
前驱_元素	对象128	产生者	对象132
后继者	对象129		

图 A.3 (续)

A.2 案例 A.2: 基于 BPMN 的“处理_在线_汽车_维修”过程

“处理_在线_汽车_维修”过程描述了在线汽车维修订单的处理过程。如图 A.4 所示, 汽车维修订单的处理过程使用 BPMN 模型进行描述。

汽车维修的在线订单处理过程包含一组子过程, 这个过程将在维修请求被登记之后开始执行。首先, 需要使用信用卡支付汽车维修费用。如果信用卡无法支付, 将调用“错误_处理”过程以返回出错信息, 并提前终止整个过程。如果信用卡能够支付成功, 客户则可以开始预约维修车库。

这时, 客户有两个选择: 一是请求和使用“预订_拖车”过程, 用于预订一个拖车拖走损坏的汽车; 二是使用“租用_临时汽车”过程, 用于在汽车维修期间预订一个替代车。“租用_临时汽车”可以被分解为三个子过程, 即: “选择_临时汽车”“评估_临时汽车”和“预订_临时汽车”。“选择_临时汽车”和“评估_临时汽车”用于评估所选的临时汽车是否满足客户的需求。这两个过程将会被重复执行, 直到客户选到满意的汽车。接着, 将执行“预订_临时汽车”过程。

客户确认订单后, “处理_在线_汽车_维修”过程则执行完成。

本案例使用 BPMN 扩展子过程对“租用_临时汽车”过程(可被进一步分解为选择、评估和预订临时汽车)进行建模。在 BPMN 中, 扩展子过程用于展示父过程中子过程的分解流信息。扩展子过程使用圆弧矩形表示, 包含和显示了所有被分解的子过程。

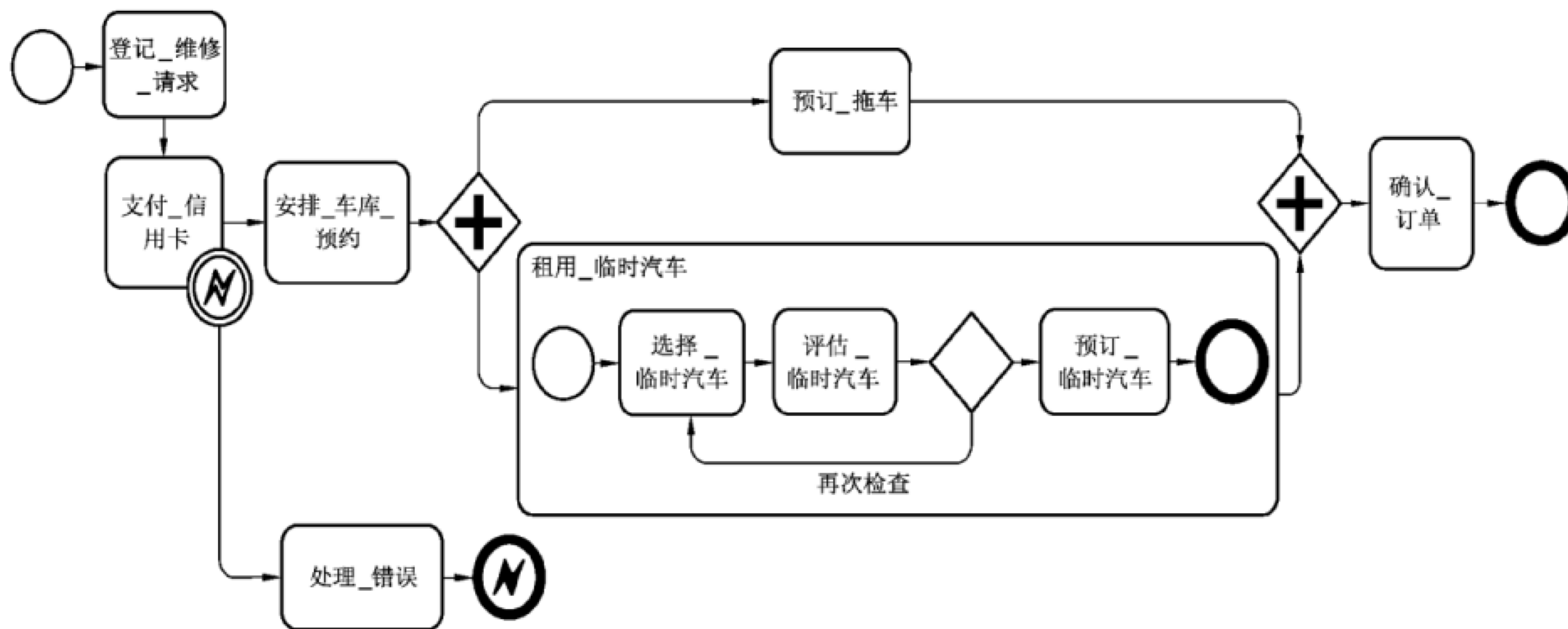



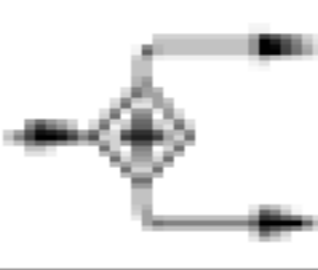







图 A.4 基于 BPMN 的“处理_在线_汽车_维修”过程模型

表 A.2 列出了 MFI 过程模型注册元模型中的哪些元类可用于表示图 A.2 中“处理_在线_汽车_维修”过程模型中的元素。使用这些元类的目的是, 将 BPMN 描述的过程模型(例如“处理_在线_汽车_维修”过程模型)注册到基于 MFI 过程模型注册元模型的注册库中。

图 A.5 展示了用于描述“处理_在线_汽车_维修”过程模型注册信息的对象实例。

表 A.2 案例 A.2 的转换关系示例

BPMN 的符号	MFI 过程模型注册元模型中的元类	BPMN 的符号	MFI 过程模型注册元模型中的元类
	事件		分支_依赖
			
			
	过程		合并_依赖
			顺序_依赖

<过程_模型>

对象201

属性/参考	值/实例
名称	"处理_在线_汽车_维修_模型"
描述_语言	对象202
被包含_过程_模型_元素	对象204, 对象205, 对象206, 对象207, 对象210, 对象212, 对象213, 对象216, 对象217, 对象219, 对象220, 对象221, 对象222, 对象225, 对象229, 对象230

<过程_建模_语言>

对象202

属性/参考	值/实例
名称	"BPMN"
版本	"2.0"
被表示_模型	对象201

<事件>

对象203

属性/参考	值/实例
名称	"维修_请求"
被触发_过程_模型_元素	对象204

<过程>

对象204

属性/参考	值/实例
名称	"注册_维修_请求"
包含_模型	对象201
触发者	对象203
后继者	对象205

<顺序_依赖>

对象205

属性/参考	值/实例
包含_模型	对象201
前驱_过程	对象204
后继_过程	对象206

<过程>

对象206

属性/参考	值/实例
名称	"支付_信用卡"
包含_模型	对象201
前驱者	对象205
后继者	对象207

<分支_依赖>

对象207

属性/参考	值/实例
包含_模型	对象201
分支_依赖_类型	"异或"
是否_同步	"假"
前驱_元素	对象206
后继_选项	对象208, 对象209

<分支_依赖_选项>

对象208

属性/参考	值/实例
门户_条件	"支付_信用卡_成功"
后继_元素	对象212
前驱者	对象207

<分支_依赖_选项>

对象209

属性/参考	值/实例
门户_条件	"支付_信用卡_失败"
后继_元素	对象210
前驱者	对象207

<过程>

对象210

属性/参考	值/实例
名称	"处理_错误"
包含_模型	对象201
前驱_选项	对象209
被产生_事件	对象211

<事件>

对象211

属性/参考	值/实例
名称	"信用卡_错误"
产生者	对象210

<过程>

对象212

属性/参考	值/实例
名称	"安排_车库_预约"
包含_模型	对象201
前驱_选项	对象208
后继者	对象213

图 A.5 “处理_在线_汽车_维修”过程模型的注册信息

<分支_依赖>

对象213

属性/参考	值/实例
包含_模型	对象201
分支_依赖_类型	"或"
是否_同步	"假"
前驱_元素	对象212
后继_选项	对象214,对象215

<分支_依赖_选项>

对象214

属性/参考	值/实例
门户_后继	"需要_拖车"
后继_元素	对象216
前驱者	对象213

<分支_依赖_选项>

对象215

属性/参考	值/实例
门户_后继	"需要_临时汽车"
后继_元素	对象217
前驱者	对象213

<过程>

对象216

属性/参考	值/实例
名称	"预订_拖车"
包含_模型	对象201
前驱_选项	对象214
后继_选项	对象227

<过程>

对象217

属性/参考	值/实例
名称	"租用_临时汽车"
包含_模型	对象201
前驱_选项	对象215
后继_选项	对象228
组合_过程_模型_元素	对象219, 对象220, 对象221, 对象222, 对象225

<事件>

对象218

属性/参考	值/实例
名称	"租用_临时汽车_请求"
被触发_过程_模型_元素	对象219

<过程>

对象219

属性/参考	值/实例
名称	"选择_临时汽车"
包含_模型	对象201
被分解_过程	对象217
触发者	对象218
后继者	对象220
前驱_选项	对象223

<顺序_依赖>

对象220

属性/参考	值/实例
包含_模型	对象201
被分解_过程	对象217
前驱_过程	对象219
后继_过程	对象221

<过程>

对象221

属性/参考	值/实例
名称	"评估_临时汽车"
包含_模型	对象201
被分解_过程	对象217
前驱者	对象220
后继者	对象222

<分支_依赖>

对象222

属性/参考	值/实例
包含_模型	对象201
被分解_过程	对象217
分支_依赖_类型	"异或"
是否_同步	"假"
前驱_元素	对象221
后继_选项	对象223, 对象224

<分支_依赖_选项>

对象223

属性/参考	值/实例
门户_后继	"不满意_评估"
前驱者	对象222
后继_元素	对象219

<分支_依赖_选项>

对象224

属性/参考	值/实例
门户_后继	"满意_评估"
前驱者	对象222
后继_元素	对象225

<过程>

对象225

属性/参考	值/实例
名称	"预订_临时汽车"
包含_模型	对象201
被分解_过程	对象217
前驱_选项	对象224
被产生_事件	对象226

<事件>

对象226

属性/参考	值/实例
名称	"租用_临时汽车_关闭"
产生者	对象225

<合并_依赖_选项>

对象227

属性/参考	值/实例
门户_后继	"预订_拖车_成功"
前驱_元素	对象216
后继者	对象229

<合并_依赖_选项>

对象228

属性/参考	值/实例
门户_后继	"预订_临时汽车_成功"
前驱_元素	对象217
后继者	对象229

图 A.5 (续)

<合并_依赖> 对象229		<过程> 对象230	
属性/参考	值/实例	属性/参考	值/实例
包含_模型	对象201	名称	"确认_订单"
合并_依赖_类型	"或"	包含_模型	对象201
是否_同步	"假"	先驱者	对象229
跟随_元素	对象230	被产生_事件	对象231
前驱_选项	对象227, 对象228		

<事件> 对象231	
属性/参考	值/实例
名称	"汽车_维修_关闭"
产生者	对象230

图 A.5 (续)

A.3 案例 A.3: 基于 EPC 的“录制_唱片”过程

“录制_唱片”过程描述了将歌曲录制为唱片的过程。如图 A.6 所示,制作唱片的过程使用 EPC 进行描述。

制作唱片的过程包含一组子过程,这个过程由“开始”事件触发。在相关设备准备好之后,可以选择一首歌曲进行录制。如果要求录制更多的歌曲(如专辑需要录制多首歌曲),则需要重复选择和录制歌曲,直到所需的歌曲都被录制到唱片中。如果不需要录制更多的歌曲,则唱片将会被送到市场部门。至此,“录制_唱片”过程被执行完成。

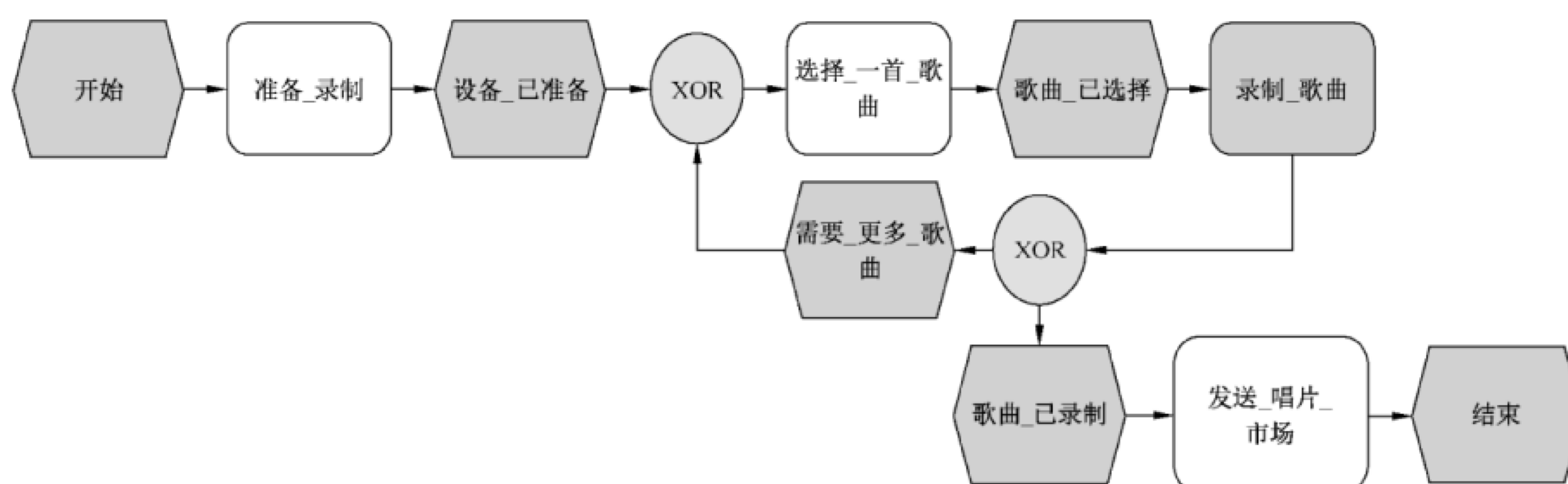


图 A.6 基于 EPC 的“录制_唱片”过程模型

表 A.3 列出了 EPC 过程模型中元素到 MFI 过程模型注册元模型中对应元类的转换示范。

表 A.3 案例 A.3 的转换关系示例

序号	EPC 的符号	MFI 过程模型注册元模型中的元类	序号	EPC 的符号	MFI 过程模型注册元模型中的元类
1		事件	4		分支_依赖
2		顺序_依赖	5		合并_依赖
3		过程	6		门户_条件

EPC 模型中的开始事件和结束事件将被注册为 MFI 过程模型注册元模型中“事件”元类的实例(见表 A.3 的序号 1)。功能被注册为过程(见表 A.3 的序号 3)。

具有一个输入和多个输出的操作被注册为分支_依赖的实例(见表 A.3 的序号 4)。在一个操作(本

案例中为异或类型)之后且与该操作直接相连的事件被注册为分支_依赖_选项(本案例中为异或类型)实例的门户条件(见表 A.3 的序号 6)。

具有多个输入和一个输出的操作被注册为合并_依赖的实例(见表 A.3 的序号 5)。在一个操作(本案例中为异或类型)之后且与该操作直接相连的事件被注册为合并_依赖_选项(本案例中为异或类型)实例的门户条件(见表 A.3 的序号 6)。

两个功能之间的事件被注册为顺序_依赖的实例(见表 A.3 的序号 2)。

“录制_唱片”过程的详细注册信息见图 A.7。

<过程_模型>
对象301

属性/参考	值/实例
名称	“录制_唱片_模型”
描述_语言	对象302
被包含_过程_模型_元素	对象304, 对象306, 对象307, 对象308, 对象309, 对象310, 对象314

<过程_建模_语言>
对象302

属性/参考	值/实例
名称	“EPC”
被表示_模型	对象301

<事件>
对象303

属性/参考	值/实例
名称	“开始”
被触发_过程_模型_元素	对象304

<过程>
对象304

属性/参考	值/实例
名称	“准备_设备”
包含_模型	对象301
触发者	对象303
后继者	对象305

<合并_依赖_选项>
对象305

属性/参考	值/实例
门户_条件	“设备_已准备”
前驱_元素	对象304
后继者	对象306

<合并_依赖>
对象306

属性/参考	值/实例
合并_依赖_类型	“异或”
是否_同步	“假”
包含_模型	对象301
后继_元素	对象307
前驱_选项	对象305, 对象313

<过程>
对象307

属性/参考	值/实例
名称	“选择_一首_歌曲”
包含_模型	对象301
前驱者	对象306
后继_元素	对象308

<顺序_依赖>
对象308

属性/参考	值/实例
包含_模型	对象301
前驱_过程	对象307
后继_元素	对象309

<过程>
对象309

属性/参考	值/实例
名称	“录制_歌曲”
包含_模型	对象301
前驱者	对象308
后继者	对象310

<分支_依赖>
对象310

属性/参考	值/实例
分支_依赖_类型	“异或”
是否_同步	“假”
包含_模型	对象301
前驱_元素	对象309
后继_选项	对象311, 对象312

<分支_依赖_选项>
对象311

属性/参考	值/实例
门户_条件	“需要_更多_歌曲”
前驱者	对象310
后继_元素	对象313

<分支_依赖_选项>
对象312

属性/参考	值/实例
门户_条件	“录制_完成”
前驱者	对象310
后继_元素	对象314

<合并_依赖_选项>
对象313

属性/参考	值/实例
门户_条件	“需要_更多_歌曲”
前驱_元素	对象311
后继者	对象306

<过程>
对象314

属性/参考	值/实例
名称	“发送_唱片_市场”
包含_模型	对象301
前驱_选项	对象312
被产生_事件	对象315

<事件>
对象315

属性/参考	值/实例
名称	“结束”
产生者	对象314

图 A.7 “录制_唱片”过程模型的注册信息

A.4 案例 A.4: 基于 OWL-S 的“查询_公交车_信息”过程

“查询_公交车_信息”过程描述了公交车信息的查询过程。如图 A.8 所示,“查询_公交车_信息”过程使用 OWL-S 进行建模。

公交车信息的查询过程包含一组子过程。这个过程将在选择查询端口后开始执行。接着,用户可以通过地名或公交车线路查询公交车的信息。最后,可视化的公交车信息将被提供给用户。

在本案例中,这些过程被表示为文本格式,并且大多都具有一样的标签结构,例如输入、输出及其之间的执行细节。图 A.8 展示了“查询_公交车_信息”过程的部分 OWL-S 代码。

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Instance Definition of Query Bus Information Process Model -->
<process:ProcessModel rdf:ID="查询_公交_信息_模型"> <process:hasProcess
rdf:resource="#Query_Bus_Information_Process"/> <service:describeS rdf:resource="http://www.daml.org/services/owl-
s/1.0/QueryBusInformationService.owl#Query_Bus_Information_Agent"/> </process:ProcessModel>

<process:AtomicProcess rdf:ID="选择_查询_端口">
  <process:comment>No Comments</process:comment>
  <process:hasInput>
    <process:Input rdf:ID="Port_Name">
      <process:parameterType rdf:datatype="http://www.w3.org/2001/XMLSchema#string"/>
    </process:Input>
  </process:hasInput>
  <process:hasOutput>
    <process:Output rdf:ID="Port_Select_Succeed">
      <process:parameterType rdf:datatype="http://www.owl-ontologies.com/UrbanTransportation.owl#Bus_Information"/>
    </process:Output>
  </process:hasOutput>
</process:AtomicProcess>

<process:AtomicProcess rdf:ID="根据_地点_查询">
  <process:comment>No Comments</process:comment>
  <process:hasInput>
    <process:Input rdf:ID="Location_Name">
      <process:parameterType rdf:datatype="http://www.w3.org/2001/XMLSchema#string"/>
    </process:Input>
  </process:hasInput>
  <process:hasOutput>
    <process:Output rdf:ID="Raw_Bus_Information_Data_With_Location">
      <process:parameterType rdf:datatype="http://www.owl-ontologies.com/UrbanTransportation.owl#Bus_Information"/>
    </process:Output>
  </process:hasOutput>
</process:AtomicProcess>

<process:AtomicProcess rdf:ID="根据_路线_查询">
  <process:comment>No Comments</process:comment>
  <process:hasInput>
    <process:Input rdf:ID="Route_Number">
      <process:parameterType rdf:datatype="http://www.w3.org/2001/XMLSchema#string"/>
    </process:Input>
  </process:hasInput>
  <process:hasOutput>
    <process:Output rdf:ID="Raw_Bus_Information_Query_With_Route">
      <process:parameterType rdf:datatype="http://www.owl-ontologies.com/UrbanTransportation.owl#Bus_Information"/>
    </process:Output>
  </process:hasOutput>
</process:AtomicProcess>
```

图 A.8 基于 OWL-S 的“查询_公交_信息”过程模型


```

<process:AtomicProcess rdf:ID="提供_公交_信息">
  <process:comment>No Comments</process:comment>
  <process:hasInput>
    <process:Input rdf:ID="Raw_Bus_Information_Data">
      <process:parameterType rdf:datatype="http://www.w3.org/2001/XMLSchema#string"/>
    </process:Input>
  </process:hasInput>
  <process:hasOutput>
    <process:Output rdf:ID="Graphic_Bus_Information_Data">
      <process:parameterType rdf:datatype="http://www.owl-ontologies.com/UrbanTransportation.owl#Bus_Information"/>
    </process:Output>
  </process:hasOutput>
</process:AtomicProcess>

<process:CompositeProcess rdf:ID="查询_公交_信息">
  <process:composedof> <process:Choice> <process:components> <process:ControlConstructList>
    <list:first>
      <process:Perform rdf:ID="Perform_Query_With_Route_AP">
        <process:process rdf:resource="根据_路线_查询"/>
      </process:Perform>
    </list:first>
    <list:rest> <process:ControlConstructList>
      <list:first>
        <process:Perform rdf:ID="Perform_Query_With_Location_AP">
          <process:process rdf:resource="根据_地点_查询"/>
        </process:Perform>
      </list:first>
      <list:rest rdf:resource="http://www.daml.org/services/owl-s/1.1/generic/ObjectList.owl#nil"/>
    </list:rest>
    </process:ControlConstructList> </list:rest>
  </process:ControlConstructList> </process:components> </process:Choice> </process:composedof>
</process:CompositeProcess>
</rdf:RDF>

```

图 A.8 (续)

根据 MFI 过程模型注册元模型进行转换和注册基于 OWL-S 的过程模型时,将自动增加至少一个起点和多个终点。在案例 A.4 中,只增加了一个起点和终点。

表 A.4 列出了从本案例中元素到 MFI 过程模型注册元模型中对应元类的转换示范。OWL-S 不是基于图的描述语言;且表现为描述属性和能力的功能块结构,而不是使用大多数图形符号的结构化片段。在前 3 个案例中使用的转换示范(例如一对一映射)不适用于表示 OWL-S 和 MFI 过程模型注册元模型中元类之间的转换。因此,我们使用 MFI 过程模型注册元模型汇中元类的复合结构用于表达 OWL-S 中的控制结构。例如,在本案例中,OWL-S 中的选择块结构被转换为异或分支依赖和异或合并依赖中的可选分支结构。OWL-S 中的原子过程被注册为 MFI 过程模型注册元模型中的过程元类。原子过程之间的所有连接将作为输入和输出,而不被认为是组合过程。“查询_公交车_信息”过程模型的详细注册信息见图 A.9。

表 A.4 案例 A.4 的转换关系示例

OWL-S 的符号	MFI 过程模型注册元模型中的元类	OWL-S 的符号	Metaclass in MFI 过程模型注册元模型中的元类
<pre> <process:AtomicProcess rdf:ID="..." ... </process:AtomicProcess> </pre>	过程	<pre> <process:composedof> <process:Choice> <process:components> <process:ControlConstruct List> ... </process:composedof> </pre>	分支_依赖(开始)+ 合并_依赖(结束) (类型相同;异或)

<过程_模型>

对象401

属性/参考	值/实例
名称	"查询_公交_信息_模型"
描述_语言	对象402
被包含_过程_模型_元素	对象404, 对象405, 对象408, 对象409, 对象410, 对象413

<过程_建模_语言>

对象402

属性/参考	值/实例
名称	"OWL-S"
版本	"1.2"
被表示_模型	对象401

<事件>

对象403

属性/参考	值/实例
名称	"查询_公交_信息_开始"
被触发_过程	对象404

<过程>

对象404

属性/参考	值/实例
名称	"选择_查询_端口"
包含_模型	对象401
触发者	对象403
后继者	对象405

<分支_依赖>

对象405

属性/参考	值/实例
分支_依赖_类型	"或"
是否_同步	"假"
包含_模型	对象401
前驱_元素	对象404
后继_选项	对象406, 对象407

<分支_依赖_选项>

对象406

属性/参考	值/实例
门户_条件	"获得_地址"
后继_元素	对象408
前驱者	对象405

<分支_依赖_选项>

对象407

属性/参考	值/实例
门户_条件	"获得_线路编号"
后继_元素	对象409
前驱者	对象405

<过程>

对象408

属性/参考	值/实例
名称	"根据_地点_查询"
包含_模型	对象401
前驱_选项	对象406
后继者	对象411

<过程>

对象409

属性/参考	值/实例
名称	"根据_路线_查询"
包含_模型	对象401
前驱_选项	对象407
后继者	对象412

<合并_依赖>

对象410

属性/参考	值/实例
合并_依赖_类型	"或"
是否_同步	"假"
包含_模型	对象401
后继_元素	对象413
前驱_选项	对象411, 对象412

<合并_依赖_选项>

对象411

属性/参考	值/实例
门户_条件	"获得_公交_信息"
前驱_元素	对象408
后继_元素	对象410

<合并_依赖_选项>

对象412

属性/参考	值/实例
门户_条件	"获得_公交_信息"
前驱_元素	对象409
后继者	对象410

<过程>

对象413

属性/参考	值/实例
名称	"提供_公交_信息"
包含_模型	对象401
前驱者	对象410
后继者	对象414

<事件>

对象414

属性/参考	值/实例
名称	"公交_信息_查询_关闭"
产生者	对象413

图 A.9 “查询_公交车_信息”过程模型的注册信息

附 录 B
(资料性附录)
过程建模语言列表

建议“过程_建模_语言”实例为表 B.1 中“名称”列中的取值。

表 B.1 过程_建模_语言列表

名称	描述
BPMN	业务过程模型和记法,对象管理组织,2011[1]
BPEL	网络服务的业务过程执行语言 (BPEL/BPEL4WS), 2003-05-03,1.1 版[2]
UML	符合 ISO/IEC 19505-2“信息技术 - OMG 统一建模语言 (OMG UML) 2.1.2 版 - 第 2 部分:超结构”的一种语言
PSL	符合 ISO/IEC 18629“过程规约语言”的一种语言[3]
IDEF3	面向过程描述捕获方法的集成定义是一种业务过程建模方法。该方法是一种情景驱动的过程流描述捕获方法,其目标是捕获与具体系统如何工作相关的知识[4]
EPC	事件驱动过程链是一种用于业务过程建模的流程图。该语言是萨尔兰大学和 SAP 在 90 年代早期共同研究和开发的[6]
OWL-S	面向服务的网络本体描述语言为网络服务的供应商提供了一组核心的置标语言,用于描述网络服务的属性和能力,而这些网络服务以计算机可理解的格式进行描述的[7]

参 考 文 献

- [1] Business Process Model and Notation (BPMN) Version 2.0, OMG Document Number: formal/2011-01-03, 2011.
 - [2] Business Process Execution Language for Web Services (BPEL 1.1), 2003-05-05.
 - [3] ISO 18629-1:2004 Industrial automation systems and integration—Process specification language—Part 1: Overview and basic principles.
 - [4] IDEF3 Process Description Capture Method Report, 1995.
 - [5] ISO/IEC 19505-2 Information technology—OMG Unified Modeling Language (OMG UML) Version 2.1.2—Part 2: Superstructure.
 - [6] August—Wilhelm Acheer. ARIS—From Business Process to Application System. Springer. 2002.
 - [7] Ontology Web Language for Service (OWL-S), Version 1.2, 2008.
-

中华人民共和国
国家标准
信息技术 互操作性元模型框架(MFI)
第5部分:过程模型注册元模型
GB/T 32392.5—2018/ISO/IEC 19763-5:2015

*

中国标准出版社出版发行
北京市朝阳区和平里西街甲2号(100029)
北京市西城区三里河北街16号(100045)

网址:www.spc.org.cn

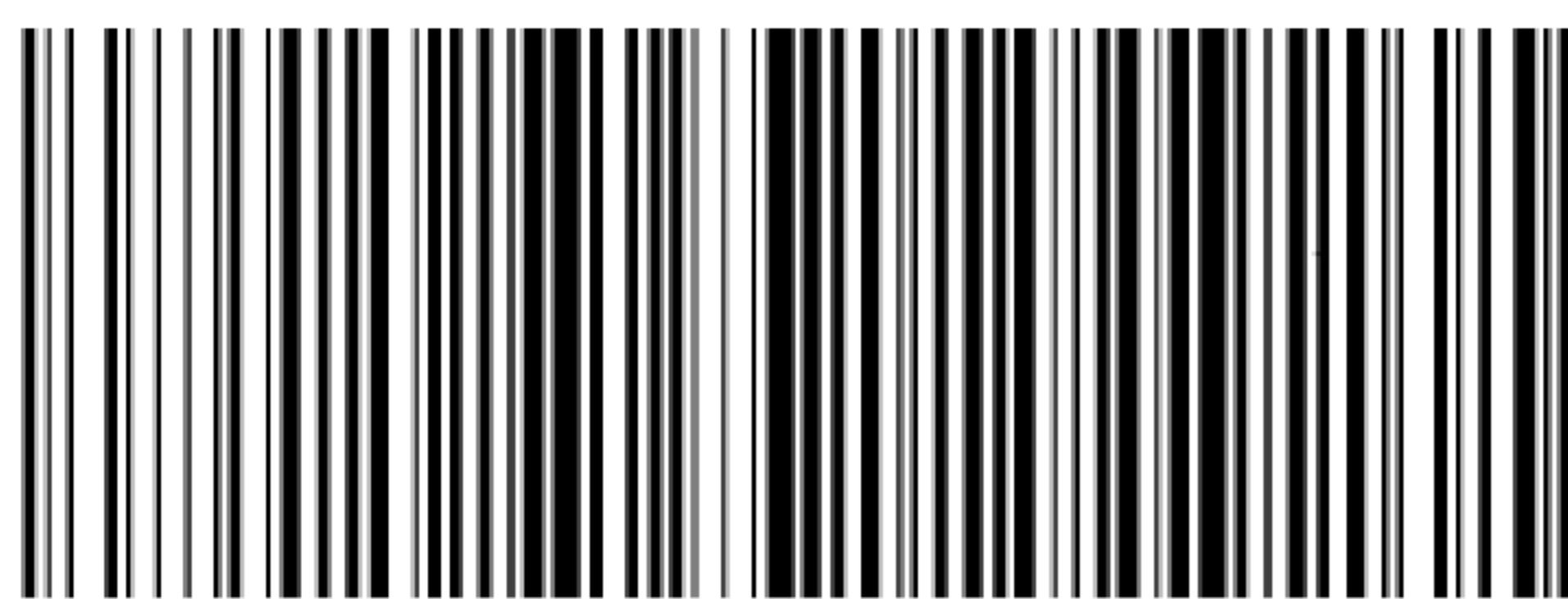
服务热线:400-168-0010

2018年3月第一版

*

书号:155066·1-59732

版权专有 侵权必究



GB/T 32392.5-2018