



中华人民共和国电力行业标准

DL / T 890.552 — 2014 / IEC 61970 – 552: 2013

能量管理系统应用程序接口 (EMS-API)

第 552 部分: CIMXML 模型交换格式

Energy management system application program interface (EMS-API) -
Part 552: CIMXML model exchange format

(IEC 61970-552: 2013, IDT)

2014-10-15 发布

2015-03-01 实施

国家能源局 发布

目 次

前言..... II

引言..... III

1 范围 1

2 规范性引用文件 1

3 术语和定义 1

4 模型交换信息头 3

5 对象标识 7

6 CIMXML 格式规则和约定 8

参考文献..... 21

前 言

本标准按照 GB/T 1.1—2009 给出的规则起草。

DL 890 标准是采用 IEC 61970 国际标准《能量管理系统应用程序接口 (EMS-API)》制定的，主要包括公共信息模型 (CIM) 和组件接口规范 (CIS) 两方面内容，由以下部分组成：

DL/T 890.1 能量管理系统应用程序接口 (EMS-API) 第 1 部分：导则和一般要求

DL/Z 890.2 能量管理系统应用程序接口 (EMS-API) 第 2 部分：术语

DL/T 890.301 能量管理系统应用程序接口 (EMS-API) 第 301 部分：公共信息模型 (CIM) 基础

DL/Z 890.401 能量管理系统应用程序接口 (EMS-API) 第 401 部分：组件接口规范 (CIS) 框架

DL/T 890.402 能量管理系统应用程序接口 (EMS-API) 第 402 部分：公共服务

DL/T 890.403 能量管理系统应用程序接口 (EMS-API) 第 403 部分：通用数据访问

DL/T 890.404 能量管理系统应用程序接口 (EMS-API) 第 404 部分：高速数据访问 (HSDA)

DL/T 890.405 能量管理系统应用程序接口 (EMS-API) 第 405 部分：通用事件和订阅 (GES)

DL/T 890.407 能量管理系统应用程序接口 (EMS-API) 第 407 部分：时间序列数据访问 (TSDA)

DL/T 890.453 能量管理系统应用程序接口 (EMS-API) 第 453 部分：基于 CIM 的图形交换

DL/T 890.501 能量管理系统应用程序接口 (EMS-API) 第 501 部分：公共信息模型的资源描述框架 (CIM RDF) 模式

DL/T 890.552 能量管理系统应用程序接口 (EMS-API) 第 552 部分：CIMXML 模型交换格式

本部分等同采用 IEC 61970-552：2013《能量管理系统应用程序接口 (EMS-API) 第 552 部分：CIMXML 模型交换格式》(英文版)。

本部分由中国电力企业联合会提出。

本部分由全国电力系统管理及其信息交换标准化技术委员会归口。

本部分起草单位：中国电力科学研究院、国家电力调度控制中心、阿尔斯通电网中国技术中心、山东大学、南瑞集团有限公司。

本部分主要起草人：曹阳、姚建国、杨胜春、陶洪铸、李晓露、梁成辉、翟明玉、米为民、潘毅、黄海峰、鄢蜜昉、高志远、王珂。

本标准在执行过程中的意见和建议反馈至中国电力企业联合会标准化管理中心(北京市白广路二条一号，100761)。

引 言

本部分是定义能量管理系统（EMS）应用程序接口（API）的 DL 890 标准的一部分。

DL/T 890.301 定义了公共信息模型（CIM）：一个与电力企业运行相关的物理逻辑视图。CIM 通过统一建模语言（UML）进行描述。UML 是一种以面向对象方式定义、展示以及描述系统的语言，是一种分析和设计语言而不是一种编程语言。软件程序使用 CIM 之前要先将其转化为一种能支持可编程接口的模式。

DL/T 890.501 描述了 UML 形式的 CIM 如何转换为机器可读的格式。这种格式采用可扩展标记语言（XML）描述，其模式使用资源描述框架（RDF）模式规范语言。

本部分定义了 DL/T 890.501 中描述的 CIM RDF 模式如何用于交换基于 XML（被称为 CIMXML）的电力系统模型。这些模型在 DL 890.45x 子集标准中定义，如在 DL 890.452 中描述的 CIM 输电网络模型交换子集。

能量管理系统应用程序接口 (EMS-API)

第 552 部分: CIMXML 模型交换格式

1 范围

DL 890 标准定义了能量管理系统应用程序接口的组件接口规范 (CIS)。本部分定义了用于交换以 CIM 为基础的建模信息所采用的格式和规则。它使用 DL/T 890.501 中描述的 CIM RDF 模式作为构建电力系统建模信息 XML 文档的元模型框架。这些文档的样式称为 CIMXML 格式。

以文件传输进行模型交换的方式有许多用途。DL 890.452 子集和 DL 890.45x 的其他子集标准解释了为交换模型设定上下文的需求和用例。虽然这种 CIMXML 格式可用于基于 CIM 的通用信息交换,但是为解决特定的交换需求还需要确定 CIM 的特定的描述 (或子集)。推动形成本规范的初始需求是为电力系统安全协调而进行的输电网建模信息交换。

本部分用于支持来自独立供应商基于公共格式形成的软件和使用 CIM 描述的建模信息。建议的解决方案应:

- 虽然主要用于编程访问,但机器和人都可读;
- 可使用支持文档对象模型 (DOM) 和其他标准 XML 应用程序接口的任何工具访问;
- 是自描述的;
- 充分采纳目前万维网联盟 (W3C) 的建议。

本部分是第二级组件接口规范文件,通过基于 CIM 的文本和例子叙述性地描述了 CIMXML 模式的详细定义。

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,仅所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。

DL 1080 电力企业应用集成——配网管理系统接口 第 11 部分: 公共信息模型 (CIM) 配电网扩展

DL/T 890.1 能量管理系统应用程序接口 (EMS-API) 第 1 部分: 导则和一般要求

DL/Z 890.2 能量管理系统应用程序接口 (EMS-API) 第 2 部分: 术语

DL/T 890.301 能量管理系统应用程序接口 (EMS-API) 第 301 部分: 公共信息模型 (CIM) 基础

DL/T 890.501 能量管理系统应用程序接口 (EMS-API) 第 501 部分: 公共信息模型资源描述框架 (CIM RDF) 模式

IEC 60050 国际电工技术词汇表

W3C RDF/XML 语法规范

W3C 可扩展标记语言 (XML) 1.0

W3C XSL 转换 (XSLT)

W3C 文档对象模型 (DOM)

3 术语和定义

IEC 60050 和 DL/Z 890.2 界定的以及下列术语和定义适用于本文件。

3.1

能量管理系统 energy management system (EMS)

一种计算机系统，包括提供基础支持服务和一组应用的软件平台。这些应用提供了使发电设备和输电设备有效运行的功能，以保证用最小的成本提供足够安全的电能。

3.2

应用程序接口 application program interface (API)

由可执行应用组件提供给其他可执行应用组件使用的一组公用函数的集合。

3.3

公共信息模型 common information model (CIM)

描述电力企业中所有主要对象的抽象模型，这些对象一般包含在能量管理系统（EMS）的信息模型中。

使用说明：通过提供一种用对象类和属性及它们之间的关系来表示电力系统资源的标准方法，CIM 方便了实现不同卖方独立开发的能量管理系统（EMS）应用的集成，多个独立开发的整个 EMS 之间的集成以及 EMS 和其他涉及电力系统运行的不同方面的系统的集成，例如发电或配电管理系统之间的集成。

3.4

CIMXML

用于交换本部分所定义的 XML 数据的序列化格式。

3.5

文档对象模型 document object model (DOM)

一种与平台和语言无关的接口，由万维网联盟（W3C）定义，它允许应用程序和脚本动态存取及交换文档的内容、结构和样式。

3.6

文档类型定义 document type definition (DTD)

一种用于描述与 XML 文档有关的词汇表和语法的标准。

使用说明：XML 模式和 RDF 是可以使用的其他形式。

3.7

超文本标记语言 hypertext markup language (HTML)

用来在万维网上格式化和表示信息的标记语言。

3.8

模型 model

描述真实或计算的对象或实体的数据集合。

使用说明 1：在 CIM 上下文中，数据语义由子集定义；参见 3.9。

使用说明 2：在电力系统分析中，模型是一组描述电力系统的静态数据。模型的例子包括静态电网模型、拓扑解以及一个潮流应用或状态估计应用生成的网络解。

3.9

子集 profile

定义一个可交换的模型结构和语义的模式。

使用说明：子集是对更为通用的 CIM 加以约束形成的集合。

3.10

子集文档 profile document

用于特定业务目的子集集合。

3.11

资源描述框架 resource description framework (RDF)

W3C 推荐的一种语言，用于表示机器可简单处理的元数据。

使用说明：RDF 使用 XML 作为其编码语法。

3.12

RDF 模式 RDF schema

用 RDF 表示的一种模式规范语言，用于描述资源及其性质，包括资源如何与其他资源相关联，它用于规定一种应用相关的模式。

3.13

现实世界对象 real-world object

属于现实世界问题域的对象，不同于实现中的接口对象和控制器对象。

使用说明 1：EMS 领域的现实世界对象定义为 DL/T 890.301 公共信息模型中的类。

使用说明 2：对电力系统中的类和对象进行建模需要以公共的方式表达给 EMS 应用。一个类是现实世界中一个对象的描述，例如需要表示为 EMS 中电力系统总体模型一部分的 PowerTransformer、GeneratingUnit 或 Load。其他类型的对象包括 EMS 应用需要处理、分析和存储的事物，如计划、量测。这些对象需要一个公共的表示以满足 EMS-API 标准即插即用和互操作性的目的。电力系统中一个有唯一标识的特定对象被建模为它所属类的一个实例。

3.14

标准通用标记语言 standard generalized markup language (SGML)

一种国际标准，定义了以电子形式表示文本且与设备及系统无关的方法。

使用说明：HTML 和 XML 都源于 SGML。

3.15

统一建模语言 unified modeling language (UML)

一种面向对象的建模语言和方法论，用于对一个系统密集过程规范化、可视化、构造和文档化的方法。

3.16

统一资源标识符 unified resource identifier (URI)

一种用于标识（引用）资源（如文件、文档、图像等物）的 Web 标准语法和语义。

3.17

可扩展标记语言 extensible markup language (XML)

标准通用标记语言（SGML）（ISO 8879）的一个子集，用于将结构化数据放入文本文件。

使用说明：这是 W3C 赞成的推荐。它具有免费、与平台无关和良好的软件工具支持性等特点。

3.18

可扩展样式表语言 extensible stylesheet language (XSL)

一种表示 XML 文档的样式表语言。

4 模型交换信息头

4.1 概述

模型交换通常涉及一组文档的交换，每个文档都包含实例数据（称作模型）和信息头。每个模型的结构和语义由一个不包含在交换数据里的子集来描述。整个交换受控于子集文档（Profile Document）中的一组子集。

信息头描述了文档中的模型内容，比如模型创建的日期、描述等。信息头也可以标识其他模型以及它们与现有模型的关系。当模型是工作流的一部分时，这些信息是重要的。例如，模型彼此间都有关联，

如模型继承和/或依赖于另一个模型。

4.2~4.4 定义了包含信息头数据的模型以及设计要支持的工作流。

4.2 CIMXML 文档和信息头

一个 CIMXML 文档由一个而不能由多个信息头描述，因此一个 CIMXML 文档中的实例数据对应于一个子集。

如果有多个可能相关的 CIMXML 文档需要放在一起，应将它们集合在类似 zip 的一个归档文件中。

4.3 模型和信息头数据描述

模型的描述作为信息头数据附加到模型上。图 1 显示了带信息头信息的模型。

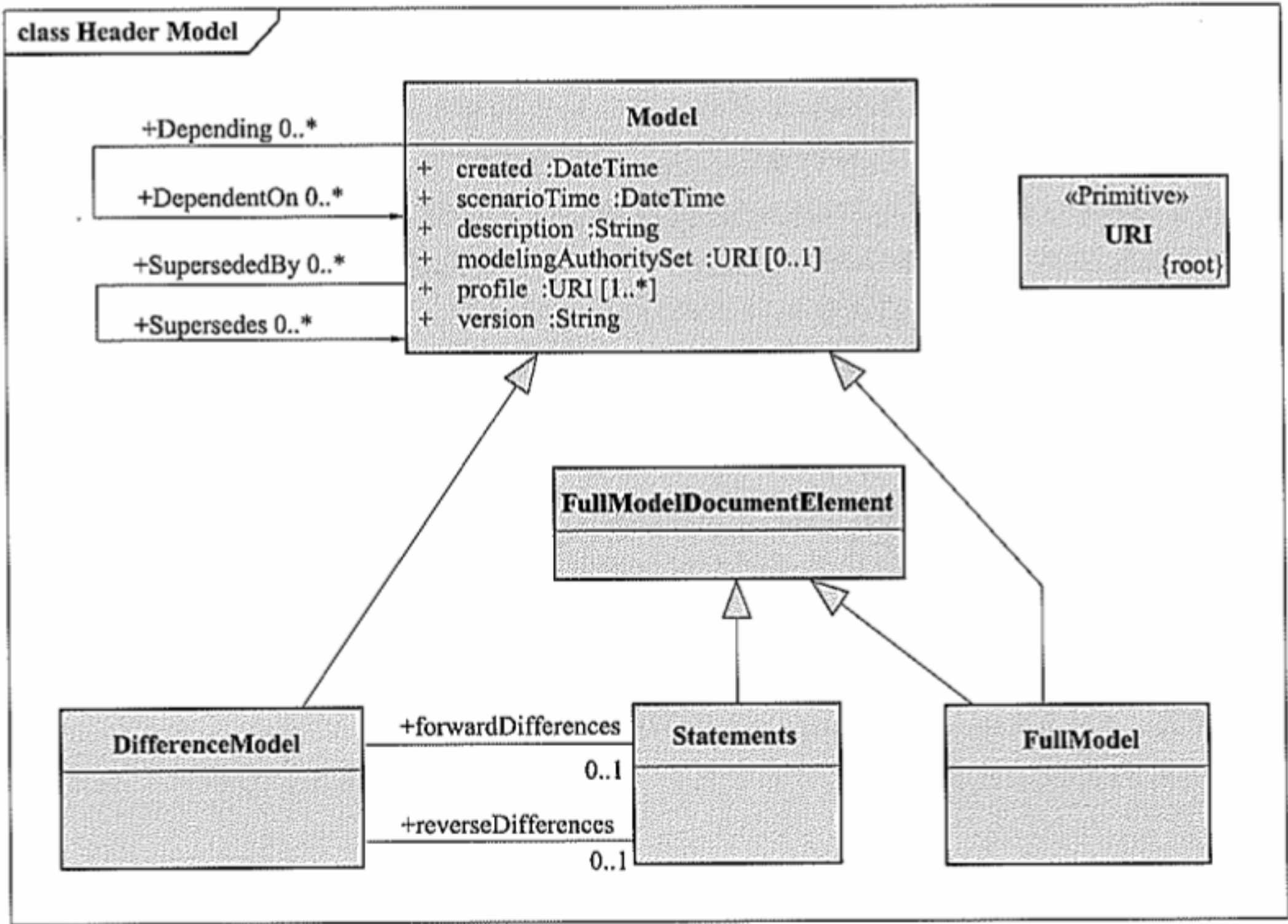


图 1 带信息头信息的模型

在图 1 中，全模型类（FullModel）、差异模型类（DifferenceModel）和声明类（Statements）描述了模型数据，而信息头由模型类（Model）描述。以下按自下而上的顺序对这些类进行描述：

- 全模型文档元素类（FullModelDocumentElement）表示所有可能出现在整个模型文档中的元素。它有两个子类型：声明（Statements）或全模型（Full Model），下面将对这两个子类型进一步描述。一个全模型文档通常包括一个全模型（FullModel）元素和一组定义（Definition）元素。
- 声明类（Statements）表示一组定义（Definition，参见 6.2.3.5）和/或描述（Description，参见 6.2.3.6）元素。
- 全模型（FullModel，参见 6.2.3.4）类表示全模型信息头，它的内容由模型类（Model）描述。
- 差异模型（DifferenceModel，参见 6.2.4.6）类表示差异模型的信息头。其内容由模型类（Model）、前向差异关联角色、后向差异关联角色来描述。两种关联角色可共用一组声明（Statements）。
- 模型类（Model）描述的信息头内容对于全模型（FullModel）和差异模型（DifferenceModel）来说是相同的。模型（Model）通过 rdf:about 属性标识。rdf:about 属性唯一描述了模型而非包含信息头的文档。因此由同样的未变化数据模型创建的多个文档具有相同的 rdf:about。这也意味着模型改变将导致下一个文档创建时具有新的 rdf:about。

● 模型类 (Model) 的属性见表 1。

表1 信息头属性

模型类	属性	描述
Model	created	模型创建的日期（注意这通常不是 CIMXML 文档创建的时间，CIMXML 文档在这之后才创建）
Model	scenarioTime	模型代表的日期和时间，例如当前运行模型、历史模型或未来模型对应的时间
Model	description	对模型的描述，例如模型创建者的名字和创建目的
Model	modelingAuthoritySet	描述源自 CIMXML 文档数据的设备模型的一个统一资源名（Uniform Resource Name, URN），例如一个国家的全部或部分模型
Model	profile	描述用于规范模型的子集的一个 URN。它唯一标识子集及其版本
Model	version	源自 CIMXML 文档数据的模型版本描述。例如： ——建模授权集的设备模型变化； ——导致不同结果的不同研究用例。 版本属性是一个随 rdf:about 标识同步变化的定制字符串，参见上述模型类 (Model) 描述
Model	DependentOn	由此文档描述的模型所依赖模型的引用，例如： ——潮流计算结果依赖于拓扑模型； ——拓扑计算出的拓扑模型依赖于网络模型
Model	Depending	依赖这个模型的所有模型。此角色不包含于任何交换实例数据的文件中
Model	Supersedes	当更新模型时，生成的模型替代原先作为更新基础的模型。因此这指向描述被更新模型的 CIMXML 文档的引用
Model	SupersededBy	替代这个模型的所有模型。此角色不包含于任何交换实例数据的文档中

子集属性是拥有以下格式的 URI:

http://iec.ch/<committee>/<year>/<standard>-<part>/<profile>/<version>

例如，http://iec.ch/TC57/2011/61970-452/Equipment/2。

图 1 中的 UML 转化为如下 CIMXML 元素:

- a) 图 1 中的叶子类（差异模型 DifferenceModel、说明 Statements 和全模型 FullModel）表示为文档元素（6.2.3.3）下的类元素。
- b) 说明元素 (Statement) 表示为定义 (Definition, 6.2.3.5) 或描述元素 (Description, 6.2.3.6)。
- c) 文字属性，例如 Model.created，表示为文字属性元素（6.2.3.8）。
- d) 角色，例如 Model.Supersedes，表示为资源属性元素（6.2.3.10）。
- e) 继承属性和角色直接表示为上述遵循 c)、d)、e) 规则的叶子类下的元素。
- f) CIMXML 模型文档由模型 (Model) rdf:about 属性（隐含在 UML 中）标识。因此 DependentOn 和 Supersedes 角色都是对模型 (Model) rdf:about 属性的引用。
- g) 一个全模型文档可从同一个数据源多次生成。这种从不变数据源重新生成的全模型文档应保持模型标识 (Model rdf:about) 与初始全模型文档一致。
- h) 当生成一个替代差异的全模型文档时，如果自差异创建开始模型没有改变，新的全模型文档具有与差异相同的模型标识 (Model rdf:about)。因此它是差异的一种替代。

4.4 workflow

一个 workflow 描述为一交换事件。4.3 节模型描述支持与 Model.Supersedes 属性在时间上关联以及与 Model.DependentOn 属性在子集上关联的 workflow 事件。workflow 事件举例如图 2 所示。

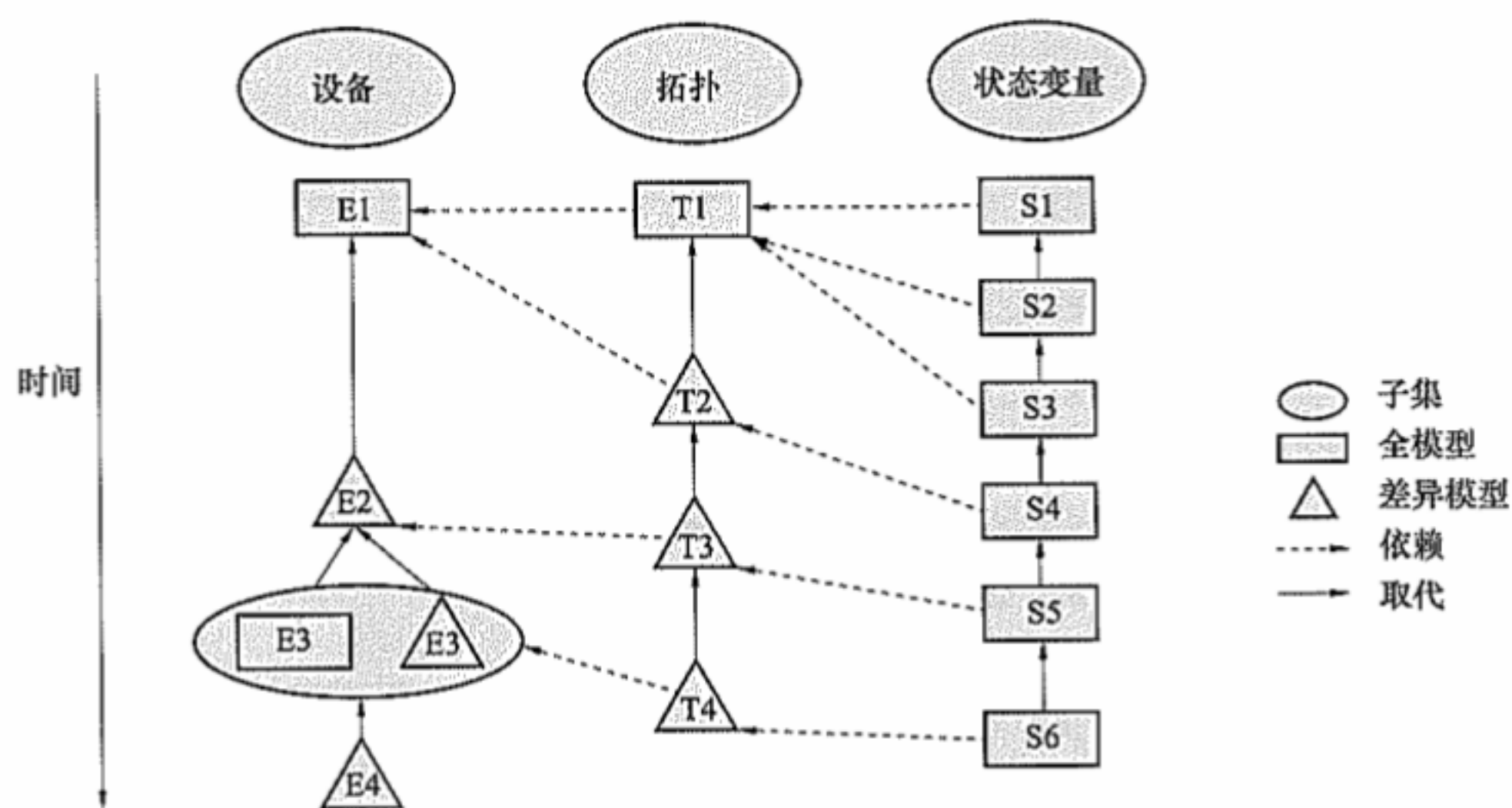


图2 workflow事件举例

在这个例子中，被交换的一个可解的网络模型是由子集文档约束的一组包含设备、拓扑和状态变量文档的模型。图2中左侧的时间轴表示设备模型文档是如何随时间交换的。中间的时间轴显示新拓扑结果如何随时间交换以及每个拓扑依赖的设备模型。右侧的时间轴表示多个状态变量文档如何交换以及状态变量文档所依赖的拓扑文档。还应注意到，设备模型E3由一个全模型和一个差异模型两种方式表示。图2显示的是一个比较简单的例子，图3显示了更为复杂的情况。

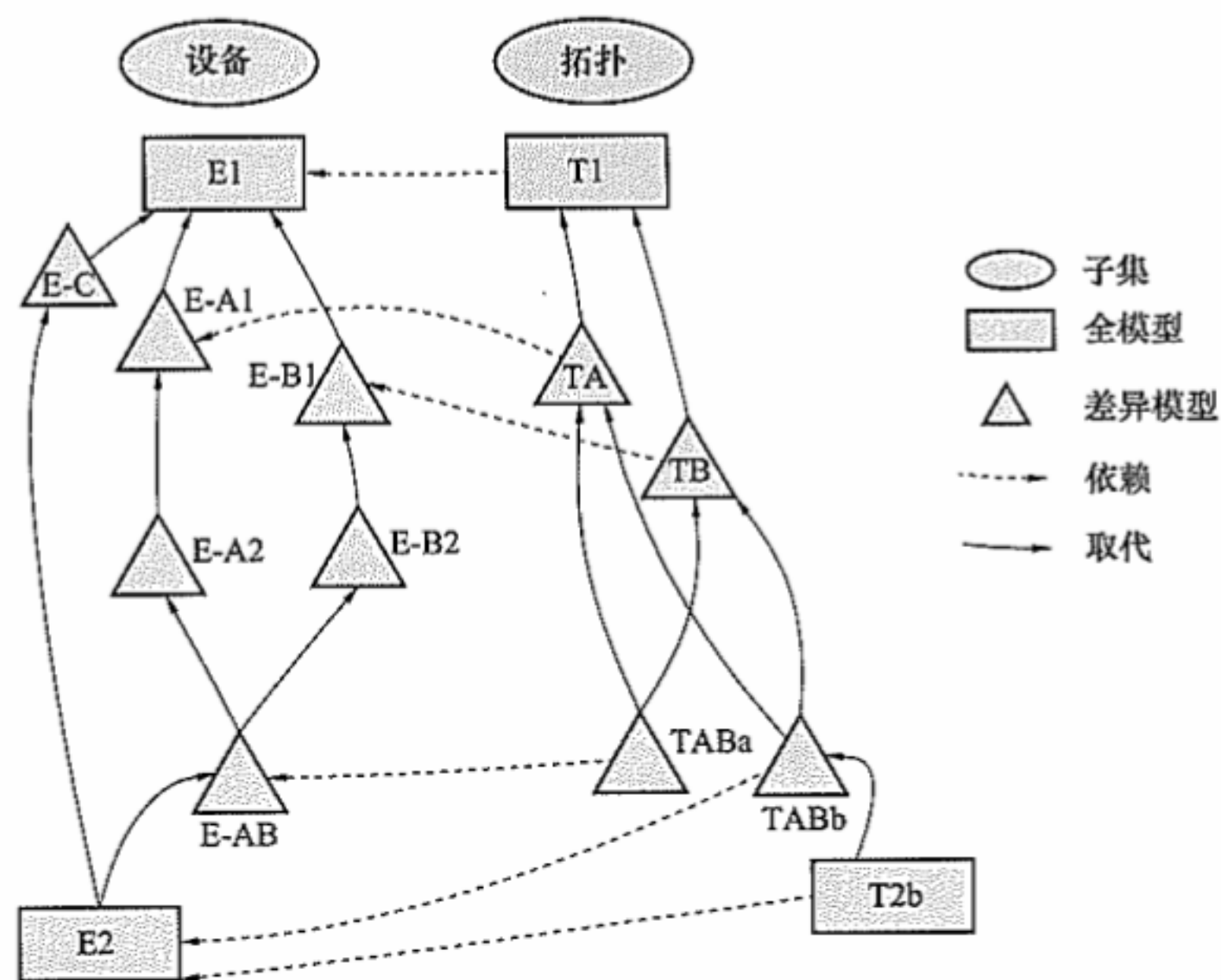


图3 有更多依赖关系的工作流事件举例

图3的CIMXML文档可能是在一个数据建模环境下创建的，在此环境中，一个模型的多个变化路径并存，例如设备模型有三个路径E-Ax、E-Bx和E-C，最终合并为全模型E2替代以上多个设备路径。

CIMXML文档的接收者可能会任意使用拓扑文档TA、TB、TABa或T2b之一以及E2设备模型。因为发送者（在这个例子里指数据建模环境）仅用E2来验证T2b，所以这被认为可相互协同的唯一组合。至于T2b，接收者也可能选择在T1上应用TB和TABb来代替使用T2b。

5 对象标识

5.1 URIs 作为标识符

UUID (通用统一标识符), 也叫 GUID (全局唯一标识符), 可用于以下方式标识资源:

- 标识符可由不同的授权方独立、唯一地分配。这是 UUID 的一大优势。
- 标识符在时间上和文档间是稳定的。

此外, 如果 UUID 嵌在一个统一资源名称 (URN) 中, 那么通过去掉 XML 基命名空间声明可简化文档。URN 是一个简洁的、固定长度的、绝对的 URI。

包含 UUID 的 URN 标准在互联网工程任务组 RFC4122 文档中定义。

RFC4122 定义了 URN 的语法以及最后一个冒号之后的 UUID 部分如何分配。分配算法符合并在技术上兼容 IEC 9834-8:2004 “信息技术.开放系统互连 (OSI). OSI 注册授权方操作规程: 唯一标识符 (UUIDs) 的生成、注册以及其作为 ASN.1 对象标识符元素的使用” 和 ITU-T Rec.X.667,2004。

CIMXML 元素由 URI 标识。URI 有两种形式:

- URL 统一资源定位符;
- URN 统一资源名称。

URL 和 URN 两种形式有完全不同的结构, 即:

- URL 形式: 协议://授权/路径?查询#片段, CIMXML 中的协议为 http。
- URN 形式: urn: 命名空间: 规范, CIMXML 中的命名空间是 uuid (参见 6.4)。

URN 中规范的格式汇总如下:

- 8 字符的 16 进制数;
- 短划符“-”;
- 4 字符的 16 进制数;
- 短划符“-”;
- 4 字符的 16 进制数;
- 短划符“-”;
- 4 字符的 16 进制数;
- 短划符“-”;
- 12 字符的 16 进制数。

其中的字母是小写。

URN 形式示例如下:

”urn:uuid:26cc8d71-3b7e-4cf8-8c93-8d9d557a4846”。

5.2 关于 rdf:ID 和 rdf:about

CIMXML 元素可由两个不同的 RDF 结构标识:

- rdf:ID;
- rdf:about。

rdf:ID 和 rdf:about 的使用与它们在 RDF 里的定义并不一致, 而有其特定的含义:

- rdf:ID 全局说明对象的生命周期, 即它是创建或是删除。
- rdf:about 是对现有对象的引用。

5.3 CIMXML 元素标识

对象标识在 RDF 中极其重要, 以至于表示对象的所有元素都用 rdf:ID 或 rdf:about 属性来标识。CIM 中所有继承 IdentifiedObject 的类都有 UML 对象标识属性 IdentifiedObject.mRID。这个属性隐含映射到 rdf:ID/rdf:about XML 属性。

CIMXML 文档可以只使用 URN 形式参见 5.1, 详情如下。

CIMXML 文件包含描述 CIM 对象（ACLineSegment, Substation 等）的 XML 元素。CIM 有许多在 XML 元素中显示为引用的关联角色（典型的如 rdf:resource 或 rdf:about 属性）。CIM 数据通过不同的 CIMXML 文档进行交换，这些文档参见 4.4 中所述相互依赖。有的引用超越了 CIMXML 文档的界限，这造成的结果是在 CIM 对象的生命周期中其标识必须稳定，否则在文档间引用对象将失效。

面向对象系统中一个普遍的做法是假设所有的对象都有一个在空间和时间上唯一的标识符，也就是：

- 不同的对象被分配了不同的标识符。
- 即使拥有它的原始对象已消失，被分配的标识符也不能重复使用。

5.1 中描述的 URN 形式被用作 CIMXML 元素标识，它有以下不同的形式：

- 前缀 “urn:uuid:” 替换为下划线 “_”。下划线避免了标识的非基本部分以数字字符开头。标识的非基本部分以数字字符开头是无效的 RDF。即使 UUID 以非数字字符开头，在所有标识前加下划线也将简化解析工作。
- 前缀被定义为 xml:base=“urn:uuid:”。

示例如下：

- rdf:ID=“_26cc8d71-3b7e-4cf8-8c93-8d9d557a4846”。
- rdf:about=“#_26cc8d71-3b7e-4cf8-8c93-8d9d557a4846”。

6 CIMXML 格式规则和约定

6.1 概述

按照 DL/T 890.501 中描述的 CIM RDF 模式，电力系统模型能被转换导出为 XML 文档，如图 4 所示。这个文档被称为 CIMXML 文档。CIMXML 文档中使用的所有标签（资源描述）都由 CIM RDF 模式提供。CIMXML 模型交换文档解析后的信息能够导入外部系统。

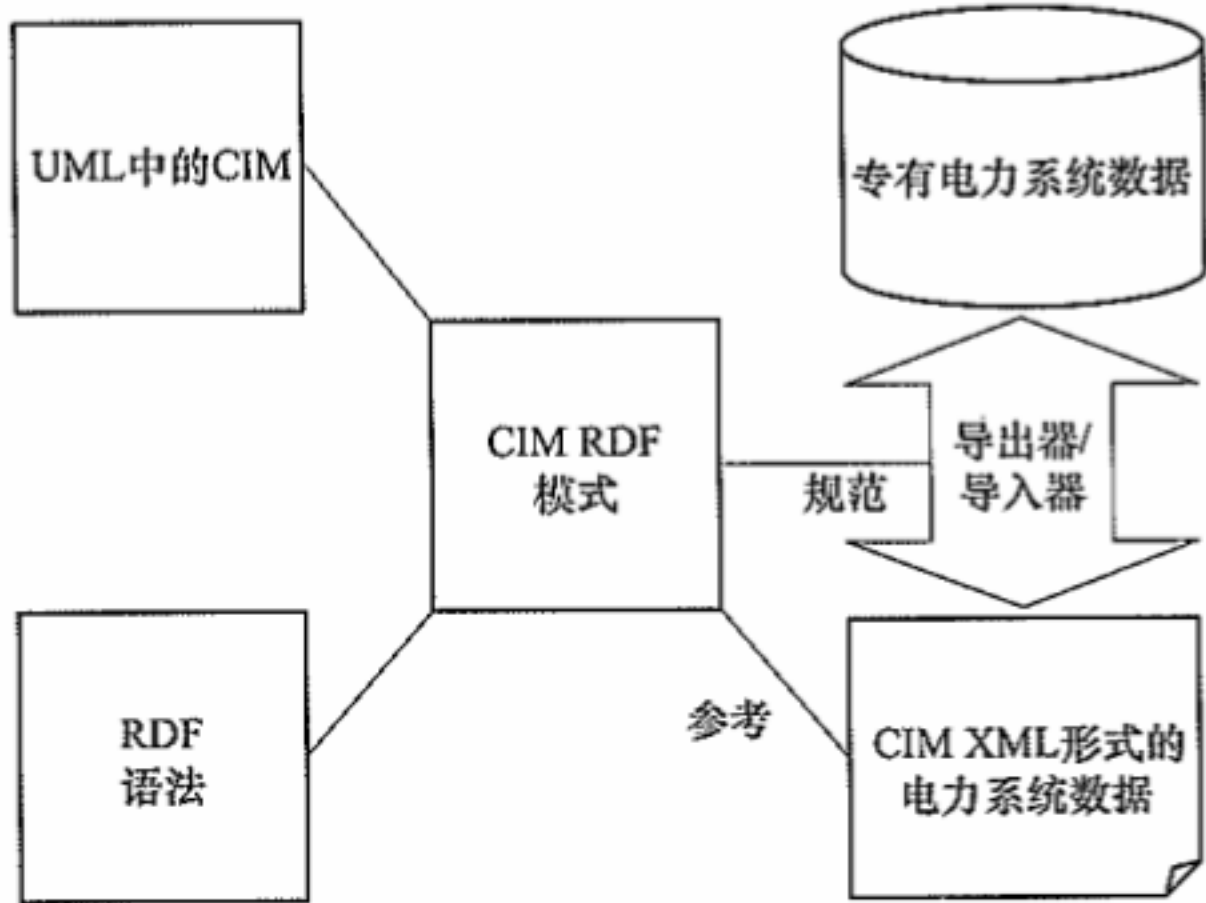


图 4 基于 CIMXML 的电力系统模型交换机制

6.2 简化 RDF 语法

6.2.1 概述

RDF 语法提供表示相同数据集的许多方法。例如，两个资源间的关联可以用一个资源的属性代表也可以通过将一个元素嵌套在另一个元素里。这可能造成难以使用一些 XML 工具（如 XSL 处理器）来处理 CIMXML 文档。

因此，创建 CIMXML 文档仅需使用 RDF 语法的一个子集。这个语法简化了实现者建立模型的序列化和反序列化软件的工作，同时提高了通用 XML 工具处理 CIMXML 文档的效率。简化的语法是标准 RDF 语法的一个真子集；因此，它可通过可用的 RDF 反序列化软件读取。

以下小节定义 RDF 语法的一个子集。这种简化语法适用于企业间电力系统模型交换。规范的目的是方便实现者建立 RDF 数据的反序列化软件，简化他们序列化 RDF 数据时的选择，并提高通用 XML 工具（如 XSLT 处理器）序列化 RDF 数据的效率。

简化的语法是标准 RDF 语法的一个真子集；因此，它可通过 RDF 反序列化软件如 SirPAC^[8]读取。这一点与简化语法的其他方案不同，如[9]、[10]。

简化的语法不丢失任何 RDF 数据模型的能力。也就是说，任何 RDF 数据都可以使用这个语法进行转换。此外，RDF 的特征仍然保留，如将定义在一个文档中的模型以声明的形式扩展到第二个文档中的功能。

6.2.2 标记方法

简化的语法在这一节定义。每种元素以一个小节定义，其开头是元素的模型，后面跟着一些定义的文本以及对 RDF 语法的引用。元素的语义不再详细描述（参见 RDF 建议^[3]提供的信息）。元素模型的标记方法如下：

- 元素类型、属性名称或属性值中的斜体符号说明了所需名称或值的类型。此符号将在文本中定义。
- “rdf”符号代表了实现所选择的命名空间前缀属于 RDF 命名空间。类似的，“cim”符号代表了选择的 CIM 命名空间前缀。
- 元素模型内的注释说明了允许的内容。斜体符号代表一种元素或定义在文本中的其他内容。
(a|b)结构表示 a 和 b 互相可替代。a*结构表示 0 或多个 a。
- 模型中所有其他文本都是文字性的。

6.2.3 语法定义（规范性）

6.2.3.1 概述

语法定义使用例子来充实。举例可以帮助更好地理解正式的语法定义。同一个例子可用于几个语法定义。示例中的重点语法加粗显示。

6.2.3.2 本规范定义的命名空间 URI

本部分定义以下命名空间：

- cim-model-description_uri，描述为 `xmlns:md`。
- difference-model-namespace-uri，描述为 `xmlns:dm`。

它们的值定义为：

- `xmlns:md="http://iec.ch/TC57/61970-552/ModelDescription/1#"`。
- `xmlns:dm="http://iec.ch/TC57/61970-552/DifferenceModel/1#"`。

6.2.3.3 文档元素

```
<rdf:RDF xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns#
xmlns:cim="cim-namespace-uri"
xmlns:md="cim-model-description_uri">
<!-- 内容: 全模型 (定义|描述)* -->
</rdf:RDF>
```

```
<rdf:RDF xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns#
xmlns:cim="cim-namespace-uri"
xmlns:md="cim-model-description_uri"
xml:base="urn:uuid:">
<!-- 内容: 全模型 (定义|描述)* -->
</rdf:RDF>
```

示例:

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:cim="http://iec.ch/TC57/2004/CIM-schema-cim10#"
    xmlns:md="http://iec.ch/TC57/61970-552/ModelDescription/1#"
    xml:base="urn:uuid:">
</rdf:RDF>
```

- a) 元素类型是 rdf:RDF。
- b) RDF 命名空间必须声明为 http://www.w3.org/1999/02/22-rdf-syntax-ns#
- c) 必须声明 CIM 命名空间, CIM 命名空间中的版本需随 CIM 模式的新版本进行调整。交换 CIMXML 文档的各方需对使用的版本达成一致。
- d) 可以声明其他命名空间。
- e) 应有 xml:base 属性, 参见 5.3。
- f) RDF^[3]语法条款: 参见 6.1 节。

6.2.3.4 全模型 (FullModel) 元素

```
<md:FullModel rdf:about=model-uri>
    <!-- 内容: (文本属性|资源属性|复合属性)* -->
</md:FullModel >
```

示例:

```
<md:FullModel rdf:about="urn:uuid:_26cc8d71-...">
    <md:Model.created>2008-12-24</md:Model.created>
    <md:Model.Supersedes rdf:resource="urn:uuid:_26cc8d71-a002-4c2b-bcf4-7bc97430bf87"/>
    <md:Model.DependentOn rdf:resource="urn:uuid:_26cc8d71-a002-4c2b-bcf4-7bc97430bf88"/>
    <md:Model.version>V32</md:Model.version>
    <md:Model.modelingAuthoritySet>http://polarenergy.com/2008/NorthPoleTSO</md:Model.modelingAuthoritySet>
    <md:Model.description>Santa Claus made a study case peak load summer base topology solution</md:Model.
description>
    <md:Model.profile>http://iec.ch/TC57/61970-456/StateVariables/1</md:Model.profile>
</md:FullModel>
```

- a) 全模型元素引入一个新模型。
- b) about 属性的值: model-uri 是由实现所选择的一个名称。model-uri 唯一标识一个文档, 也是其他文档引用此模型的名称, 如图 1 所示的 Supersedes 或 DependentOn。

6.2.3.5 定义 (Definition) 元素

```
<classname rdf:ID=identity>
    <!-- 内容:
        (文本属性|资源属性|复合属性)*
    -->
</classname>

<classname rdf:about=resource-uri>
    <!-- 内容:
        (文本属性|资源属性|复合属性)*
    -->
</classname>
```


-->

</classname>

示例:

```
<cim:SynchronousMachine rdf:about="#_31dcf429-6bfb-4e2e-b2996-42491b3abc1">
  <cim:IdentifiedObject.name>IN-2</cim:IdentifiedObject.name>
  <cim:SynchronousMachine.minimumMVar>-9999</cim:SynchronousMachine.minimumMVar>
  <cim:SynchronousMachine.operatingMode
rdf:resource="http://iec.ch/TC57/2001/CIM-schema-cim10#SynchronousMachineOperatingMode.generator"/>
  <cim:RegulatingCondEq.RegulationSchedule rdf:resource="#_ca32746f-a002-4c2b-bcf4-7bc97430bf87"/>
  <cim:Equipment.EquipmentContainer rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a75"/>
```

</cim:SynchronousMachine>

- 定义元素引入一个新的资源并赋予其类型。有两种形式: rdf:ID 属性和 rdf:about 属性。
- 元素类型: classname, 是类的 XML 限定名, 其来自于 CIM 模式或其他模式, 在文档元素中声明为命名空间。
- ID 属性的值: identity, 由实现选择。它在文档中必须是唯一的。(不必与电力系统资源名称关联)

6.2.3.6 描述 (Description) 元素

```
<rdf:Description rdf:about=resource-uri>
```

<!-- 内容:

(文本属性|资源属性|复合属性)*

-->

</rdf:Description >

示例:

```
<rdf:Description rdf:about="#_26cc8d71-a002-4c2b-bcf4-7bc97430bf87">
<cim:IdentifiedObject.name>TROY</cim:IdentifiedObject.name>
</rdf:Description>
```

- 描述元素增加了有关该文档或其他文档中引入资源的信息。
- resource-uri 是一个标识主体资源的 URN-reference。
- 描述 (Description) 元素只能用于差异模型中 (参见 6.2.4), 而不能用于全模型。

6.2.3.7 复合元素

```
<classname>
```

<!-- 内容:

(文本属性|资源属性|复合属性)*

-->

</classname>

示例:

```
<cim:DateTimeInterval>
  <cim:DateTimeInterval.start>2013-02-28</cim:DateTimeInterval.start>
  <cim:DateTimeInterval.end>2013-02-29</cim:DateTimeInterval.end>
</cim:DateTimeInterval>
```

- 复合元素引入了一个结构化值。这个值不代表资源也没有标识, 它只作为属性的对象出现。
- 元素类型: classname, 是复合类的 XML 限定名。
- 复合元素是一个原子单元, 因此复合元素在多个具有不同成员集的元素中是不能拆分的。参见 6.2.4.7.4。

6.2.3.8 文字属性元素

<propname>

<!-- 内容: 文本 -->

</propname>

示例:

<cim:SynchronousMachine rdf:ID="_31dcf429-6Bfb-4e2e-b2996-42491b3abc1">

<cim:IdentifiedObject.name>IN-2</cim:IdentifiedObject.name>

<cim:SynchronousMachine.minimumMVar>-9999</cim:SynchronousMachine.minimumMVar>

<cim:SynchronousMachine.operatingMode

rdf:resource="http://iec.ch/TC57/2001/CIM-schema-cim10#SynchronousMachineOperatingMode.generator"/>

<cim:RegulatingCondEq.RegulationSchedule rdf:resource="#_ca32746f-a002-4c2b-bcf4-7bc97430bf87"/>

<cim:Equipment.EquipmentContainer rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a75"/>

</cim:SynchronousMachine>

- 文字属性元素引入了一个属性和一个应用于所包含资源的文本值。
- 元素类型: propname, 是一个来自于 CIM 模式或在文档元素中声明为命名空间的其他模式的属性的 XML 限定名。
- 内容文本是除<, >和&以外的代表属性值的任意 XML 文本。
- 当导入和再导出时, 浮点数可以通过取整做稍许改变。这是允许的但需要由应用进行管理, 例如, 采用死区进行数值比较是允许的。

6.2.3.9 复合属性元素 Compound-Property element

<propname>

<!-- 内容: (复合) -->

</propname>

示例:

<cim:TimeSchedule>

<cim:TimeSchedule.scheduleInterval>

<cim:DateTimeInterval>

<cim:DateTimeInterval.start>2013-02-28</cim:DateTimeInterval.start>

<cim:DateTimeInterval.end>2013-02-29</cim:DateTimeInterval.end>

</cim:DateTimeInterval>

</cim:TimeSchedule.scheduleInterval>

</cim:TimeSchedule>

6.2.3.10 资源属性元素

<propname rdf:resource=resource-uri/>

- 资源属性元素引入了一个属性和一个资源, 而资源的值应用于所包含的资源。
- 元素类型: propname, 是一个来自于 CIM 模式或在文档元素中声明为命名空间的其他模式的属性的 XML 限定名。
- resource-uri 是一个标识资源的 URN-引用。
- 对角色重数大于 1 的关联, 资源属性元素应重复多次引用。

示例 1 URN-引用:

这个例子包括两个引用, 一个指向 RegulationSchedule (调节计划), 另一个指向 EquipmentContainer (设备容器)。

<cim:SynchronousMachine rdf:ID="_31dcf429-6Bfb-4e2e-b299-642491b3abc1">


```

<cim:IdentifiedObject.name>IN-2</cim:IdentifiedObject.name>
<cim:SynchronousMachine.minimumMVar>-9999</cim:SynchronousMachine.minimumMVar>
<cim:SynchronousMachine.operatingMode
rdf:resource="http://iec.ch/TC57/2001/CIM-schema-cim10#SynchronousMachineOperatingMode.generator"/>
<cim:RegulatingCondEq.RegulationSchedule rdf:resource="#_cd32746fa0024c2bbcf47bc97430bf87"/>
<cim:Equipment.EquipmentContainer rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a75"/>
</cim:SynchronousMachine>

```

示例 2 枚举：

这个例子定义了 SynchronousMachine.operatingMode 的属性值为“generator”。operatingMode 在 CIM 模式中被指定为枚举类型 SynchronousMachineOperatingMode。

```

<cim:SynchronousMachine rdf:ID="_31dcf429-6bfb-4e2e-b2996-42491b3abc1">
  <cim:IdentifiedObject.name>IN-2</cim:IdentifiedObject.name>
  <cim:SynchronousMachine.minimumMVar>-9999</cim:SynchronousMachine.minimumMVar>
  <cim:SynchronousMachine.operatingMode
rdf:resource="http://iec.ch/TC57/2001/CIM-schema-cim10#SynchronousMachineOperatingMode.generator"/>
  <cim:RegulatingCondEq.RegulationSchedule rdf:resource="#_cd32746fa0024c2bbcf47bc97430bf87"/>
  <cim:Equipment.EquipmentContainer rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a75"/>
</cim:SynchronousMachine>

```

示例 3 重数大于 1 的角色

```

<cim:SynchronousMachine rdf:ID="_31dcf429-6bfb-4e2e-b2996-42491b3abc1">
  <cim:IdentifiedObject.name>IN-2</cim:IdentifiedObject.name>
  <cim:SynchronousMachine.minimumMVar>-9999</cim:SynchronousMachine.minimumMVar>
  <cim:SynchronousMachine.operatingMode
rdf:resource="http://iec.ch/TC57/2001/CIM-schema-cim10#SynchronousMachineOperatingMode.generator"/>
  <cim:RegulatingCondEq.RegulationSchedule rdf:resource="#_cd32746fa0024c2bbcf47bc97430bf87"/>
  <cim:Equipment.EquipmentContainer rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a75"/>
  <cim:Equipment.ReactiveCapabilityCurves rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a76"/>
  <cim:Equipment.ReactiveCapabilityCurves rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a77"/>
  <cim:Equipment.ReactiveCapabilityCurves rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a78"/>
</cim:SynchronousMachine>

```

6.2.4 差异模型的语法扩展

6.2.4.1 概述

本条款的第一部分中定义的一般语法用于部分和全模型的数据交换。一旦交换了初始完整模型数据集，当模型发生变化时仅需通过更新来维护模型。通常，这些变化可被规范成两个模型之间的一组差异。差异文件是一个 RDF 模型（一组 RDF 语句集），因此可由 RDF 基础工具进行处理。

6.2.4.2 用例示例

下面通过一个用例示例来说明差异文档处理差异模型更新的方法。在这个例子中，参与者是 Regional Energy 公司和 Network Power 公司：

- 每个参与者有一份电力系统模型 B1 的拷贝。
- Regional Energy 公司更新 B1，以反映即将到来的电力系统改造，产生 B2。
- Regional Energy 公司将 B1 和 B2 之间的差异作为差异模型发送给 Network Power 公司。
- Network Power 公司审查和验证差异模型。
- Network Power 公司将差异模型与它们那份 B1 模型合并，产生 B2。

另一种方法是 Regional Energy 公司直接将 B2 发送给 Network Power 公司。但是, B2 是一个非常大的模型, 无法在合理的时间周期内完成对它的校验。因为校验不是完全自动的, 还需要专家分析。事实上, B2 的最佳验证策略是将它与已校验过的 B1 进行比较。这使我们回到差异模型的需求上。

一个更复杂的用例涉及两个以上的参与者。Regional Energy 公司的几个同行业公司提供差异模型给 Network Power 公司。这样的用例将引入并行模型改变和并发冲突的问题。

6.2.4.3 要求

给定两个 RDF 模型 B1 和 B2 (称之为基本模型), 对差异模型的要求是:

- 表示两个基本模型之间的差异。
- 自身是一个 RDF 模型 (一个 RDF 语句集), 因此可由 RDF 基础设施处理。
- 高效描述两个大的基本模型之间的小差异。
- 当一个对象被删除, 应用差异的系统负责执行“级联删除”, 即查找和删除所包含的所有对象。与删除对象间的关联也应删除。
- 删除操作不可逆 (至少不会来自于差异模型中的信息)。
- 可能包含自身的信息, 如来源、目的和日期。
- 可能包含一些信息以防止当同一个基本模型同时生成两个差异模型时产生冲突。

把差异文件当做数据库提交操作的要求不在这项服务范围内 (即回滚功能, 如果需要的话, 是接收侧应用而不是发送端应用的责任)。这就认可了发送端应用可能不知道自 B1 的上次更新后由其他代理对 B2 模型作出的改变。

6.2.4.4 差异文件的结构

给定两个基本 RDF 模型, B1 和 B2, 差异模型是由四组语句组成, 每一个都编码为一种资源描述结构的序列:

- 信息头语句, 包含差异模型自身信息的语句。
- 前向差异语句, B2 中包含而 B1 中不含的语句。
- 后向差异语句, B1 中包含而 B2 中不含的语句。
- 前提语句, B1 和 B2 中都包含的语句, 被认为是差异模型在应用定义层面的依赖关系。

任何一组或所有组都可以是空的。

差异模型自身信息由资源类型 `dm:DifferenceModel` 表示, 对这种资源通常使用模型自身的 URN。

下列属性适用于差异模型资源:

- `dm:forwardDifferences` 是一个差异模型的属性, 它的值是一个语句集 (即资源类型 `rdf:Statement`), 描述前向差异语句。
- `dm:reverseDifferences` 是一个差异模型的属性, 它的值是一个后向差异说明。
- `dm:preconditions` 是一个差异模型的属性, 它的值是前提条件语句集。

信息头属性也适用于差异模型资源。这些可以表明来源、日期和目的。这些属性可以从都柏林核心词汇集 (Dublin Core) 或任何其他方便的模式得到。

差异模型词汇的命名空间由刚提及的前缀 `dm:` 表示: <http://iec.ch/TC57/61970-552/DifferenceModel/1#>。

6.2.4.5 前提和并发

前提语句是 B1 和 B2 的子集, 不包含差异信息。简单地说, 在连续的模型修订场景中它们可以省略。

对于一个大的共享模型, 连续修订并不总是可行的。修订有可能由不同的参与者同时进行, 且并没有顾及对方。由此带来的并发问题必须要处理, 但是传统的面向数据库的方法使用锁来检测不一致的并发事务, 这在网络层面并不适用。

前提语句是锁的一种替代。它们非正式地代表等同于数据库事务的只读锁的信息。处理差异模型的软件代理可以检查前提是否正确, 如果不对将发出不一致的模型修订告警。

前提中包含语句的选择是与应用相关的 (这与在数据库事务中锁定信息的选择类似)。前提应包括

会影响形成模型版本的代理进行决策的语句。

6.2.4.6 差异模型模板

下面是差异模型约定语法的一个模板。

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cim="cim-namespace-uri"
  xmlns:md="cim-model-description_uri"
  xmlns:dm="difference-model-namespace-uri"
  xml:base="urn:uuid:">
  <dm:DifferenceModel rdf:about=model-uri>
    <!-- 内容: (文本属性|资源属性|复合属性)* -->
    <dm:preconditions parseType="Statements">
      <!--内容: (定义|描述)* -->
    </dm:preconditions>
    <dm:forwardDifferences parseType="Statements">
      <!--内容: (定义|描述)* -->
    </dm:forwardDifferences>
    <dm:reverseDifferences parseType="Statements">
      <!--内容: (定义|描述)* -->
    </dm:reverseDifferences>
  </dm:DifferenceModel>
</rdf:RDF>
```

为了说明命名空间“dm”，通过一个新的属性 `rdf:parseType` 引入新的声明，这是标准 RDF 语法的有效扩展，称之为“Statements”。

```
<property parseType="Statements">
  <!--内容: (定义|描述)* -->
</property>
```

包含一个 `rdf:parseType="Statements"` 元素的内容模型与 `rdf:RDF` 元素的内容模型相同。

和出现在一个 `rdf:RDF` 元素里一样，内容会产生相同的 RDF 语句。

6.2.4.7 差异模型的使用

6.2.4.7.1 概述

以下示例说明了差异模型的使用。

6.2.4.7.2 增加资源

如果增加特定资源，对于给出的资源，差异模型只包括前向差异语句。

示例：

下面的例子添加了两条新交流线路段（ACLineSegment），以及其相邻的端子（Terminal）。这些端子（Terminal）连接到新的连接节点（ConnectivityNode）。这些连接节点（ConnectivityNode）被分配给现有厂站（Substation）内新的电压等级（VoltageLevel）。

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cim="cim-namespace-uri"
  xmlns:md="cim-model-description_uri"
  xmlns:dm="difference-model-namespace-uri"
  xml:base="urn:uuid:">
  <dm:DifferenceModel rdf:about="#_26cc8d71-3b7e-4cf8-8c93-8d9d557a4846">
```

```

<md:Model.created>2008-12-24</md:Model.created>
<md:Model.Supersedes rdf:resource="#_26cc8d71-3b7e-4cf8-8c93-8d9d557a4847"/>
<md:Model.DependentOn rdf:resource="#_26cc8d71-3b7e-4cf8-8c93-8d9d557a4848"/>
<md:Model.version>V32</md:Model.version>
<md:Model.modelingAuthoritySet>http://polarenergy.com/2008/NorthPoleTSO</md:Model.modelingAuthoritySet>
<md:Model.description>Santa Claus made a study case peak load summer base topology solution</md:Model.description>
<md:Model.profile>http://iec.ch/TC57/61970-452/EquipmentModel/1</md:Model.profile>
<dm:forwardDifferences rdf:parseType="Statements">
  <!--增加交流线路段（ ACLineSegment） ACLine_New1 -->
  <cim:ACLineSegment rdf:ID="_26cc8d71-12f1-4de9-9e68-125d95073a75">
    <cim:IdentifiedObject.name>New 1</cim:IdentifiedObject.name>
    <cim:Conductor.r>0.0646</cim:Conductor.r>
    <cim:Conductor.x>0.5961</cim:Conductor.x>
    <cim:Conductor.bch>0.4066</cim:Conductor.bch>
  </cim:ACLineSegment>
  <cim:Terminal rdf:ID="_26cc8d71-... ">
    <cim:IdentifiedObject.name>T1</cim:IdentifiedObject.name>
    <cim:Terminal.ConnectivityNode rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75"/>
    <cim:Terminal.ConductingEquipment rdf:resource="#_26cc8d71-..."/>
  </cim:Terminal>
  <cim:Terminal rdf:ID="_26cc8d71-12f1-4de9-9e68-125d95073a756">
    <cim:IdentifiedObject.name>T2</cim:IdentifiedObject.name>
    <cim:Terminal.ConnectivityNode rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75"/>
    <cim:Terminal.ConductingEquipment rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75"/>
  </cim:Terminal>
  <!--增加交流线路段（ ACLineSegment） ACLine_New2 -->
  <cim:ACLineSegment rdf:ID="_26cc8d71-12f1-4de9-9e68-125d95073a75">
    <cim:IdentifiedObject.name>New 2</cim:IdentifiedObject.name>
    <cim:Conductor.r>0.0646</cim:Conductor.r>
    <cim:Conductor.x>0.5961</cim:Conductor.x>
    <cim:Conductor.bch>0.4066</cim:Conductor.bch>
  </cim:ACLineSegment>
  <cim:Terminal rdf:ID="_26cc8d71-... ">
    <cim:IdentifiedObject.name>T1</cim:IdentifiedObject.name>
    <cim:Terminal.ConnectivityNode rdf:resource="#_26cc8d71-..."/>
    <cim:Terminal.ConductingEquipment rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75"/>
  </cim:Terminal>
  <cim:ConnectivityNode rdf:ID="_26cc8d71-12f1-4de9-9e68-125d95073a75">
    <cim:IdentifiedObject.name>ND New1</cim:IdentifiedObject.name>
    <cim:ConnectivityNode.EquipmentContainer
rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75"/>
    <cim:ConnectivityNode.Terminals rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75"/>
  </cim:ConnectivityNode>
  <cim:Terminal rdf:ID="_26cc8d71-12f1-4de9-9e68-125d95073a75">

```

```

<cim:IdentifiedObject.name>T2</cim:IdentifiedObject.name>
<cim:Terminal.ConnectivityNode rdf:resource="#_26cc8d71-..." />
<cim:Terminal.ConductingEquipment rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75" />
</cim:Terminal>
<cim:ConnectivityNode rdf:ID="_26cc8d71-12f1-4de9-9e68-125d95073a75">
  <cim:IdentifiedObject.name>ND New2</cim:IdentifiedObject.name>
  <cim:ConnectivityNode.EquipmentContainer
rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75" />
  <cim:ConnectivityNode.Terminals rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75" />
</cim:ConnectivityNode>
<cim:VoltageLevel rdf:ID="_26cc8d71-12f1-4de9-9e68-125d95073a75">
  <cim:IdentifiedObject.name>230K</cim:IdentifiedObject.name>
  <cim:VoltageLevel.Substation rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75" />
  <cim:VoltageLevel.BaseVoltage rdf:resource="#_26cc8d71-12f1-4de9-9e68-125d95073a75" />
</cim:VoltageLevel>
</dm:forwardDifferences>
</dm:DifferenceModel>
</rdf:RDF>

```

6.2.4.7.3 删除资源

如果特定资源被删除，对于给出的资源，差异模型只包括后向差异语句。

级联删除是指删除对象及其子对象（如果有）。在级联删除中，CIMXML 文件中有可能只包括根或父对象。接收方需要找出删除哪些子对象。为了要明确级联删除包含哪些对象，CIMXML 文件创建者需要包含所有在级联中作为元素的对象。只包含根或父对象是不容许的。

设备容器（Equipment Container）与设备（Equipment）间的关系是父子关系，当删除设备容器时也将删除它的子设备。其他具有父子关系的例子还有：

- 设备容器还有一些父子关系，如厂站（Station）与电压等级（Voltage Level）。
- 变压器（Power Transformer）和它的绕组（Transformer End）。
- 导电设备（Conducting Equipment）和它的端子（Terminal）。

CIM 目前没有对包含关系进行规定。由于没有此信息，将哪种关系当做包含关系由实现者决定。这不利于互操作。因此通过在级联删除中包括所有的对象，以说明发送系统对包含关系的解释。

在级联删除中对象所拥有的非包含关系的关联将被切除，例如，如果一个连接节点（Connectivity Node）不受级联删除的影响，但与之相连的一个导电设备（Conducting Equipment）受影响，那么端子（Terminal）与连接节点（Connectivity Node）间的关联将被切除。这意味着如果指向一个切除对象的引用来自于保留对象，那么该保留对象的引用将被删除，这也意味着需对保留对象进行更新。

删除元素应该包含其所有属性元素，原因是这样可以撤销删除操作并重新创建对象。

示例：

下面的例子中包含对一台电力变压器（PowerTransformer）及其下面所有从属资源的删除。

```

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:cim="cim-namespace-uri"
xmlns:dm="difference-model-namespace-uri"
xml:base="urn:uuid:">
  <dm:DifferenceModel rdf:about="#_26cc8d71-12f1-4de9-9e68-125d95073a75">
    <!--为简洁起见忽略信息头 -->

```



```

<!-- 删除变压器 -->
<dm:reverseDifferences rdf:parseType="Statements" >
    <cim:PowerTransformer rdf:ID="_41bb4445-6756-43fa-9e5a-48B6cd71790e">
        ...后面跟着变压器的所有属性...
    </cim:PowerTransformer>
    ...后面跟着变压器的所有部分...
</dm:reverseDifferences>
</dm:DifferenceModel>
</rdf:RDF>

```

6.2.4.7.4 更新资源

如果资源发生变化, 差异模型包括了给定资源的前向差异和后向差异语句。

示例:

下面的例子定义了负荷 (Energy Consumer) 通过将它的端子 (Terminal) 连接到另一个连接节点 (Connectivity Node) 来体现从 115kV 移到 230kV 的变动。

```

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cim="cim-namespace-uri"
  xmlns:dm="difference-model-namespace-uri"
  xml:base="urn:uuid:">
  <dm:DifferenceModel rdf:about="#_26CC8D71-...">
    <!--为简洁起见忽略信息头 -->

    <!-- 将负荷(EnergyConsumer)从 115kV 移到 230kV-->
    <dm:forwardDifferences rdf:parseType="Statements" >
      <rdf:Description rdf:about="#_39e4e305-1c70-4dcc-a423-45e4812dcd07">
        <cim:Terminal.ConnectivityNode rdf:resource="#_612fa147-902c-4f88-be3f-0302b3750b18"/>
      </rdf:Description>
    </dm:forwardDifferences>
    <dm:reverseDifferences rdf:parseType="Statements" >
      <rdf:Description rdf:about="#_39e4e305-1c70-4dcc-a423-45e4812dcd07">
        <cim:Terminal.ConnectivityNode rdf:resource="#_5d74fc6a-b518-4a3e-9e72-4827efd197cf"/>
      </rdf:Description>
    </dm:reverseDifferences>
  </dm:DifferenceModel>
</rdf:RDF>

```

对复合元素的更新是完整的将其替换, 亦即, 通过后向差异语句将旧元素及其所有成员删除, 并通过前向差异语句将其添加。

6.3 CIMXML 格式指南 (资料性)

RDF 语法的一个有用的功能是它可以在一个文件中实现电力系统模型的任意子集的序列化, 但这是一把双刃剑。由一方生成的文档如果不包含所有预期的属性, 可能不能由另一方使用。此外, 如果资源的 URN 与其他文档不一致, 包含部分模型的文档可能无法使用。

下列准则适用于一个 CIMXML 文档的内容, 有助于最大化使用 CIMXML 文档的应用的范围:

- a) 在每个资源的引入点处包含可能的主键属性。例如, `cim:IdentifiedObject.name` 和 `cim:Equipment.EquipmentContainer` 属性很可能是必需的属性。

原因：许多应用需要装入一个有模型数据的数据库，许多数据库模式在插入对象时需要主键值。

- b) 包括单值属性而不是反向多值属性。例如，使用 `cim:Equipment.EquipmentContainer` 而不是 `cim:EquipmentContainer.Contains_Equipments`。

原因：因为这些属性是相反的，基于一侧的语句等同于基于另一侧的反向语句。只包含一侧属性不容易出错，而且更容易编辑或转换文档。

- c) 当遇到多对多关系时，通常有一个主引用方向。包含主引用而不是其对侧多值。例如，使用 `cim:SynchronousMachine.MVArCapabilityCurves` 而不是 `cim:MVArCapabilityCurve.SynchronousMachines`，因为主要关系来自于 `SynchronousMachine` 到 `MVArCapabilityCurve`。

原因：与上述 b) 同样的原因。

- d) 当遇到一个一对一关系，包含其一而不是同时包含两个。导入软件需要能够处理每个方向的引用，并推断出对侧。

原因：因为这些属性是相反的，基于一侧的语句等同于基于另一侧的反向语句。这不易产生误差和争议，使文档的编辑或变换更容易。

- e) 如果使用多值属性，它将以具有相同属性名的重复属性元素出现。

6.4 以 CIMXML 元素表示新增、删除和更新的对象（规范性）

下列情况描述了元素标识以及它们如何出现在全模型和差异模型中：

- 新增对象由全模型或差异模型中 `rdf:ID` 属性标识的定义元素（参见 6.2.3.5）来表示。
- 删除对象由差异模型中 `rdf:ID` 属性标识的定义元素（参见 6.2.3.5）来表示。
- 更新对象由差异模型中 `rdf:about` 属性标识的描述元素（参见 6.2.3.6）来表示。
- 添加属性（例如，内部的一个空值变为一个有效值）是一个只出现在前向差异模型的变化。
- 删除属性（例如，内部的一个有效值变为一个空值）是一个只出现在后向差异模型的变化。

6.5 CIM 子集生成 CIM RDF 模式

DL/T 890.501 讨论 CIM RDF 模式的生成。一个 CIMXML 模型交换文档使用 CIM 子集解决特定用例的模型交换需求，参见 DL/T 890.400 系列部分子集文档。一个 CIM 子集定义了 CIMXML 文档导入和导出时需处理的 CIM 部分内容。子集的 RDF 模式只包括为该子集定义的类和属性。

应用能用 CIM UML 模型产生 RDF 模式文件。应用具有一个可交互式定义 CIM UML 模型子集的用户接口。应用能用 RDF 模式文件校验 CIMXML 文档，见图 5。

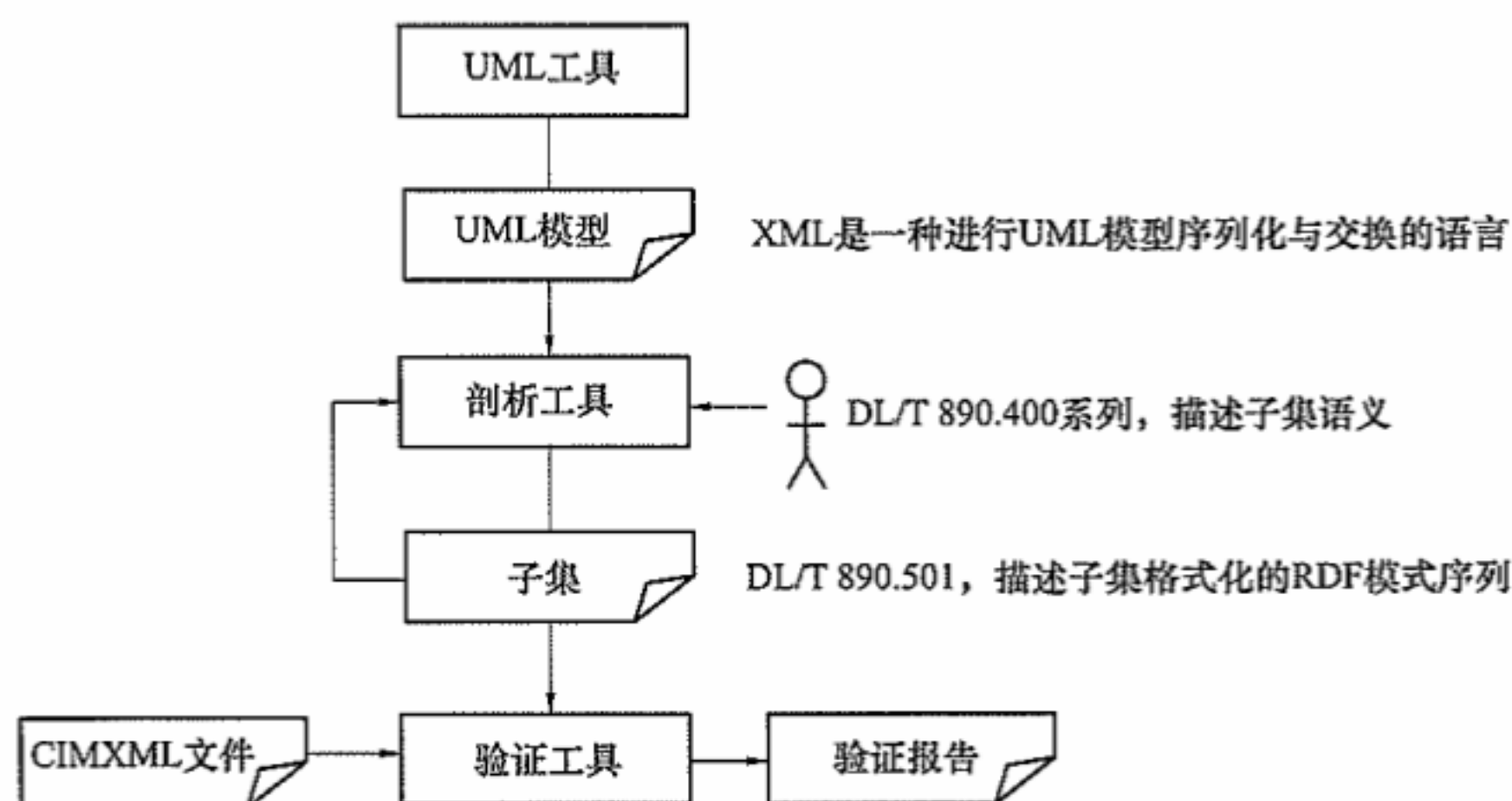


图 5 UML、子集和 CIMXML 工具间的关系图

6.6 CIM 扩展

通过提供单独的命名空间，CIM RDF 模式可以扩展新的类和属性。因为使用一个单独命名空间时，

定制的 CIMXML 文档中清晰界定了哪些是 CIM 标准, 哪些是定制的。可以在同一个 XML 文档中清晰标识几种不同的自定义扩展。当这些定制文档导入事先对扩展并不知晓的信息系统时, 具有未知标签的元素可以被忽略。下面的声明标识一个扩展命名空间——bpa。

xmlns:bpa="http://www.bpa.gov/schema/cim_extension/2001may"。

例如, 可以在断路器类中添加一个非 CIM 属性——OriginalPO (如下所示)。当一个系统的导入程序不关心这些扩展时, 可直接忽略这些 bpa 自定义标签。

```
<cim:SynchronousMachine rdf:ID="_31dcf429-6Bfb-4e2e-b2996-42491b3abc1">
  <cim:IdentifiedObject.name>IN-2</cim:IdentifiedObject.name>
  <cim:SynchronousMachine.minimumMVar>-9999</cim:SynchronousMachine.minimumMVar>
  <cim:SynchronousMachine.operatingMode
rdf:resource="http://iec.ch/TC57/2001/CIM-schema-cim10#SynchronousMachineOperatingMode.generator"/>
  <bpa:OriginalPO>PO1234378</bpa:OriginalPO>
  <cim:RegulatingCondEq.RegulationSchedule rdf:resource="#_cd32746fa0024c2bbcf47bc97430bf87"/>
  <cim:Equipment.EquipmentContainer rdf:resource="#_6cb8701a-12f1-4de9-9e68-125d95073a75"/>
</cim:SynchronousMachine>
```

对应扩展的 RDF 模式可添加入一个单独的 RDF 模式文档中, 从而保持 CIM RDF 模式的清晰独立性, 并允许每一个模式独立变化。

6.7 RDF 简化语法设计原则

以下几点解释了一些可选的简化语法。

- a) 文字属性可由 property 属性 (RDF^[3]语法条款 6.10) 表示, 这将更加紧凑。但是, 之所以选择使用 property 元素, 是因为它们更容易用 XSLT 表达式处理 (例如, 它们可以进行分类)。它们也更容易表示多行文本。
- b) 语法是一种带有两级资源/属性的扁平化结构, 更深层嵌套结构可能更紧凑。此外, 适当的嵌套结构可允许常用的查询更容易编码到 XSLT 表达式中。另一方面, 选用扁平结构可能是因为它是最简单的结构, 易于生成和解释。通过避免应用对嵌套结构细节的任何依赖, 使得语法更易于移植。
- c) 所有资源在引入时都 (通过定义元素) 给定一个类型。虽然在目前的应用中不需要非类型化资源, 但 RDF 模型允许资源是非类型化的。差异模型对非类型资源使用的描述参见 6.2.3.6。

参 考 文 献

- [1] XML for CIM Model Exchange, IEEE PES PICA 2001 Conference Paper, May 2001, A. deVos, S. Widergren, J. Zhu.
- [2] Simplified RDF Syntax for Power System Model Exchange, CIMXML Interoperability Group Paper, 16 November 2000, A. deVos.
- [3] Resource Description Framework (RDF) Model and Syntax Specification, W3C Recommendation 22 February 1999 <http://www.w3.org/TR/REC-rdf-syntax>, Ora Lassila, Ralph R. Swick.
- [4] Resource Description Framework (RDF) Schema Specification, W3C Proposed Recommendation 03 March 1999 <http://www.w3.org/TR/PR-rdf-schema>, Dan Brickley, R.V. Guha, Netscape.
- [5] Uniform Resource Identifiers (URI): Generic Syntax; Berners-Lee, Fielding, Masinter, Internet Draft Standard August, 1998; RFC 2396.
- [6] Namespaces in XML; Bray, Hollander, Layman eds, W3C Recommendation; <http://www.w3.org/TR/1999/REC-xml-names-19990114>.
- [7] Extensible Markup Language 1.0 (Second Edition), W3C Recommendation 6 October 2000, <http://www.w3.org/TR/REC-xml>, Bray, Paoli Sperberg-McQueen, Maler.
- [8] “SiRPAC – Simple RDF Parser & Compiler”, <http://www.w3.org/RDF/Implementations/SiRPAC>.
- [9] “A Strawman Unstripped Syntax for RDF in XML”, Tim Berners-Lee, November 1999, <http://www.w3.org/DesignIssues/Syntax>.
- [10] “Simplified Syntax for RDF”, Sergey Melnik, November 1999, <http://www-db.stanford.edu/~melnik/rdf/syntax.html>.

中 华 人 民 共 和 国
电 力 行 业 标 准
能量管理系统应用程序接口
(EMS-API)

第 552 部分: CIMXML
模型交换格式

DL/T 890.552 — 2014 / IEC 61970-552: 2013

*

中国电力出版社出版、发行

(北京市东城区北京站西街 19 号 100005 <http://www.cepp.sgcc.com.cn>)

北京九天众诚印刷有限公司印刷

*

2015 年 6 月第一版 2015 年 6 月北京第一次印刷

880 毫米×1230 毫米 16 开本 1.75 印张 45 千字

印数 0001—3000 册

*

统一书号 155123 · 2407 定价 15.00 元

敬告读者

本书封底贴有防伪标签,刮开涂层可查询真伪

本书如有印装质量问题,我社发行部负责退换

版 权 专 有 翻 印 必 究



中国电力出版社官方微信



掌上电力书屋



上架建议: 15

5/计算机