

DOI: 10.3969/j.issn.1000-1026.2012.22.012

使用 Eclipse 建模框架实现基于公共信息模型系统的开发

董树锋¹, 何光宇², 刘凯诚², 张王俊³, 王治华³

(1. 浙江大学电气工程学院, 浙江省杭州市 310027;

2. 电力系统及发电设备控制和仿真国家重点实验室, 清华大学, 北京市 100084; 3. 上海市电力公司, 上海市 200122)

摘要: 基于 Eclipse 建模框架, 提出了一种简洁、灵活、通用的解决方案, 帮助开发者高效地开发基于公共信息模型(CIM)的系统。该方案包含 3 个方面内容: 其一, 按照 CIM 所定义的包、类、对象以及对象之间的关联关系自动生成 Java 代码, 使开发者具备快速追踪最新 CIM 版本的能力; 其二, 能够以流的方式高速扫描 CIM 可扩展置标语言(XML)文件, 扫描过程中形成 CIM 对象, 扫描结束后建立对象间的关联关系, 且能够屏蔽不同厂家、不同版本 CIM XML 文件之间的差异性, 具有较好的兼容性; 其三, 能够验证对象间关联关系是否满足 CIM 标准规定的约束。最后, 对多个实际系统的 CIM 数据进行了测试, 实验结果验证了该方案的有效性。

关键词: Eclipse 建模框架; 公共信息模型; 文件解析

0 引言

公共信息模型(CIM)是国际电工委员会制定的一个关于电力数据共享的国际标准, 该模型涵盖了包括电力元件、厂站在内的电力分析控制中常用的各种对象。作为国际标准, CIM 已被 ABB, Alstom, Siemens, SISCO 和 ICL 等 20 多个开发商用于数据采集与监控系统、网络分析和调度培训系统等 30 多种应用^[1]。在国内, 国家电力调度控制中心组织了多次 CIM 互操作实验^[2-4], 为国内主要能量管理系统(EMS)开发单位的不同自动化系统之间能够互联、互通和互操作提供了实验依据。

CIM 应用可扩展置标语言(XML)提供电力系统数据交换格式, 国内建设的自动化系统越来越多地通过 CIM XML 方式从 EMS 中获取电网数据^[5-8]。CIM 应用方面, 国内研究主要集中在 CIM XML 的导入、导出技术, 例如: 文献[5]提出了基于 Java 反射机制的 CIM XML 导入技术, 文献[8]使用实时库作为适配器实现 CIM 数据导入、导出。上述应用中, CIM 主要是被用做数据交换的中间格式。国外学者已经开始研究 CIM 作为应用核心数据结构的潜力^[9-10], 例如: 文献[9]基于 Java 开发了处理 CIM 对象的存储、提取和交换的工具包, 文献[10]研究了直接从 CIM 对象出发做拓扑分析得到 Bus-branch 模型的技术。

与以往研究不同, 本文的重点是提出一种简洁、灵活、通用的方案, 帮助开发者高效地开发基于 CIM 的系统。基于 CIM 的系统, 应具有如下特点: ①将 CIM 作为底层数据结构; ②能够通过 CIM XML 方式与其他系统进行数据交换, 并兼容不同 CIM 版本的 XML 文件; ③对于模型实例数据, 能够验证其合法性。另外, CIM 还在不断发展和完善之中, 要求开发者具有快速追踪最新 CIM 版本的能力以获得 CIM 最新成果。本文基于 Eclipse 建模框架(EMF), 提出针对上述要求的解决方案。EMF 是一个强大的代码生成工具, 被成功应用于多个项目中^[11], 应用该框架可以实现从统一建模语言(UML)定义到程序代码的快速转化, 本文应用其实现了自动化生成和更新 CIM 代码的功能, 并在 EMF 基础上进行扩展, 开发了通用控件, 为快速、高效地开发基于 CIM 的系统提供了辅助工具。

1 模型驱动架构及其实现

CIM 是由 UML 定义的一组模型, 完整、准确地描述了电力系统资源物理特性, 对于电网分析、仿真系统的开发者来说, 使用 CIM 作为底层数据结构及数据交换形式是十分自然的想法, 然而现实的困难有: ①CIM 中模型类型繁多, 采用人工方式编码的话, 工作量巨大, 导致建立和更换模型的代价很大, 难以追踪最新版本的模型; ②不同 CIM XML 文件提供者所遵循的 CIM 版本不尽相同, 为了兼容这些 XML 文件, 不得不生成多个版本的 CIM 代码, 导致底层数据结构无法统一。本文采用了模型驱动架构

收稿日期: 2011-12-27; 修回日期: 2012-05-03。
国家自然科学基金资助项目(51207136)。

(MDA)的开发方式解决了上述问题。

MDA 是由对象管理组织(OMG)提出的,其核心思想是抽象出与实现技术无关、完整描述业务功能的核心平台无关模型(PIM),然后通过辅助工具将 PIM 转换成与具体实现技术相关的平台相关模型(PSM)。实现 PIM 向 PSM 的转换是 MDA 的关键。应用 MDA 可以提高软件开发效率,保证软件质量^[12],增强软件的可移植性、协同工作能力和可维护性,以及文档编制的便利性。

EMF 是一个开放源代码的 Java 框架,可以看做是 MDA 在 Eclipse 上的一个实现,它可以快速为 CIM 生成简单、正确、有效的 Java 代码,生成的源代码允许用户创建、查询、更新、串行化、反串行化、检验和追踪 CIM 实例的变化。本文改造了 EMF 解析 XML 的过程,可以将低版本的 CIM XML 文件解析到高版本的 CIM 代码中。这样,开发者可以使用较新的 CIM 版本作为底层数据结构,而无须担心 CIM 版本向后兼容的问题。另外,EMF 提供的验证工具能够根据 UML 定义检查数据的合法性,下面将详细介绍上述内容的细节。

2 模型代码自动生成

代码自动生成是指将 CIM 标准制定委员会发布的 XMI(XML-based metadata interchange)文件或 mdl 文件变成 Java 源代码。在 EMF 内部使用一个被称为 Ecore 的核心元模型来存储模型信息,可以将 Ecore 看做是模型的模型,Ecore 元模型是 EMF 框架的核心,它描述 EMF 模型并且提供模型的运行时支持。Ecore 中的类型定义如图 1 所示。

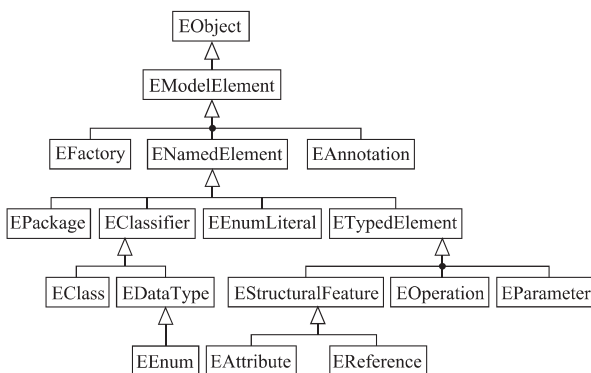


图 1 Ecore 中的类型定义
Fig. 1 Classes definition in Ecore

对图 1 中主要几个类型的说明如下。

1)EObject:EObject 是 Ecore 中一个关键接口,由图 1 可以看出,EObject 为所有 EMF 建模对象的基类型,在概念上类似于 Java 中的 java.lang.

Object 对象。为了区别于用户建模中的方法名,EObject 接口中所定义的方法名都以“e”开头。如 eClass()方法返回一个 EMF 对象的元模型 EClass。

2)EReference:用来描述类之间的关联关系,由于关联关系是 2 个类型之间的关系,所以 EReference 总是指向组合数据类型。

3)EAttribute:用来描述一个属性(简单数据),它包含属性的名字和类型。

4)EFactory:该类为一个抽象工厂类,它包含创建建模对象的方法。

5)EPackage:在 Ecore 中,EPackage 包含关于元模型类 EClass 和数据类 EDataType 的信息。

从上述介绍可以看出,Ecore 为面向对象的分析和设计提供一个标准化元模型的 Java 实现。EMF 加载器解析 mdl/XMI 文件后,将信息用 Ecore 中的对象进行存储,并用于生成 Java 代码。

Eclipse 平台上的 EMF 插件,提供强有力的代码生成工具来支持模型的生成,通过该工具,按照操作说明在图形界面经过简单几个步骤^[11]即可实现 mdl/XMI 文件的导入和模型代码的生成。本文试验了 10r7,11v01,12r002 等版本的 CIM,均能成功生成准确、完整的 Java 代码,只在版本 10r7 生成的代码中出现 EnumeratedType 类中 VALUES 变量重名问题,通过将其中一个变量改名即可解决。

3 模型实例文件导入

为实现不同 EMS、不同应用程序间 CIM 数据交换,通常使用基于资源描述框架(RDF)^[13]模式的 XML 文件来描述 CIM 数据,被称为 CIM XML。模型实例文件导入就是读取这种特定的 XML 文件,并使用第 2 节中生成的 CIM 代码进行存储的过程。本节总结了解析国内外 EMS 厂家 CIM XML 文件时的常见问题,并提出了通用解决方法。

3.1 CIM XML 文件解析原理

EMF 中的 XML 加载器使用 SAX(simple API for XML)来提供其加载支持。在分析文档时,SAX 事件用于在内存中构建 EMF 对象。在加载期间将从命名空间声明中读取命名空间统一资源标志符(URI)并使用它来定位对应的包。因此需要调用如下代码注册生成的 CIM 包。

```

void registerPackage (ResourceSet resourceSet, EPackage
ePackage) {
    //register
    resourceSet. getPackageRegistry(). put ( ePackage.
getNsURI(), ePackage. getEFactoryInstance());
    //recursion
    EList list=ePackage. getESubpackages();
    if (list==null) return;

```

```

for (Object obj: list)
    registerPackage(resourceSet, (EPackage) obj);
}

```

当加载器识别出包以后,它可以获取创建 EMF 对象的工厂,利用该工厂可以生成类的实例,EMF 对象是使用反射进行创建和实例化的,EMF 对象均提供了反射 API(eGet(),eSet())来存取它的数据。对于遇到的每个 XML 属性,加载器将按照名称在合适的包中定位对应的构造方法。处理一个属性时,加载器通过 XML 属性的名称会获取一个匹配的 Ecore 特性:如果该特性是一个 EAttribute,那么它使用对应的工厂将字符串值转换为一个对象,并相应地设置属性的值;如果该特性是 EReference,则使用属性值或 ID 来获取对应的 EObject,如果对应的 EObject 还没有创建,则存储该值,并将引用值的设置推迟到分析完成后进行。

3.2 常见问题

解析 CIM XML 文件过程中常见的问题如下。

1) 文件中存在无用信息

某些 EMS 厂商提供的 CIM XML 文件包含了该厂商扩展的内容,包括扩展的设备、扩展的属性等,通常这些内容专用于 EMS 内部,对于其他系统来说是无用信息,这类信息的特点是通常具有特别的前缀,例如下面的例子中,以 cimNARI 为前缀的属性都属于扩展信息。

```

<cim:BaseVoltage rdf:ID="200000023">
  <cimNARI:BaseVoltage, name>21 kV</cimNARI:
  BaseVoltage, name>
  <cim:BaseVoltage, nominalVoltage>21.000000<
  /cim:BaseVoltage, nominalVoltage>
  <cimNARI:BaseVoltage, nominalI>0.000000<
  /cimNARI:BaseVoltage, nominalI>
  <cimNARI:BaseVoltage, v_exm>23.000000<
  /cimNARI:BaseVoltage, v_exm>
  <cimNARI:BaseVoltage, mva_exm>114.000000<
  /cimNARI:BaseVoltage, mva_exm>
</cim:BaseVoltage>

```

2) 文件和代码中的属性名不一致

例如:美国通用电气 xa21 系统提供的 CIM XML 文件中,InitiallyUsesMVarCapabilityCurve 为发电机设备的一个属性,而在 CIM 定义中属性名为 MVarCapabilityCurves。该文件片段如下。

```

<cim:SynchronousMachine rdf:ID="SMEPC1Generator17">
  <cim:SynchronousMachine.
  InitiallyUsesMVarCapabilityCurve rdf:resource=
  " # SMEPC1GenMVarCapabilityCurve17"/>
  :
</cim:SynchronousMachine>

```

3) XML 文件和 CIM 代码版本不一致

不同的 CIM 提供者可能采用不同版本的 CIM,而不同 CIM 之间不是完全兼容的,例如:在 10r7 版本及之前的 CIM 中使用 Compensator 类来描述电容/电抗设备,而在 11 及之后版本中使用 ShuntCompensator 和 SeriesCompensator 这 2 个类来描述。

3.3 解决方案

本文做了如下 2 个方面工作,解决了 3.2 节所述的前 2 类问题。

1) 在加载 XML 文件之前,首先读入一个配置文件,该文件中 SpecialFields 指定了改名的规则,IgnoredProperties 指定了应忽略的属性名称,IgnoredPrefix 指定了应忽略的前缀,其片段如下。

```

SpecialFields = *.InitiallyUsesMVarCapabilityCurve,
MVarCapabilityCurves; *.iec61970CIMVersion,
Iec61970CIMVersion;
IgnoredProperties = *.MemberOf __SubControlArea;
Measurement, maxValue; Measurement, minValue;
IgnoredPrefix = cimNARI; cimNC

```

2) 重载 EMF 中 SAXXMLHandler 类的 processElement 方法,该方法在读入 XML 文件中的每个 Element 后会被调用。在此方法中首先判断该 Element 是否该忽略、改名,然后交给 EMF 处理,该方法的流程如图 2 所示。

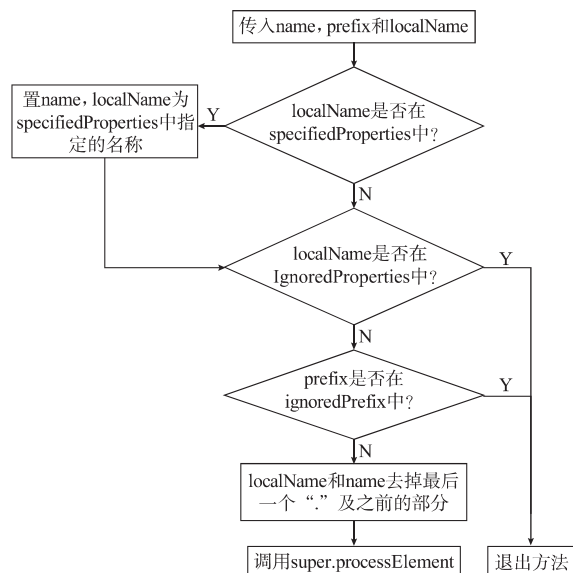


图 2 processElement 方法流程
Fig. 2 Flow chart of processElement method

本文提出一种使用 CIM 的模式,解决了 3.2 节中所述的第 3 类问题,该模式的示意图见图 3。该模式的要点是:①应用内部使用某一版本的 CIM 代码作为整个应用的底层数据结构;②使用 CIM 建模

组件将不同 CIM 版本的 XML 文件解析到内部 CIM。为了说明了这一模式的应用过程,下面的例子中,CIM 建模组件将 CIM 版本为 10r7 的 XML 文件解析到版本为 11 的 CIM 代码。

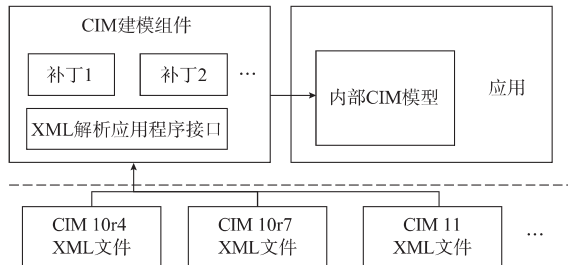


图 3 CIM 应用模式

Fig. 3 Implementation mode of CIM

版本 10r7 到版本 11 最主要的困难是将 10r7 版本中的 Compensator 转换成 ShuntCompensator 和 SeriesCompensator,本文通过下面 2 个步骤解决了该问题。

步骤 1: 定义 Java 接口 Compensator 和 CompensatorType,并用 EMF 从 Java annotation 生成模型代码^[11],作为 CIM 建模组件中的补丁。

步骤 2:重载 SAXMIHandler 类的 endDocument 方法,该方法在 XML 文件扫描结束后被调用。在此方法中将内存中的 Compensator 对象根据 CompensatorType 转换成 ShuntCompensator 或 SeriesCompensator。关键代码如下。

```
for(EObject obj; extent) {
    if (obj.eClass() == Cim2003Package.eINSTANCE.
        getCompensator()) {
        Compensator compensator = (Compensator) obj;
        if (compensator.getCompensatorType() ==
            CompensatorType.SERIES) {
            SeriesCompensator seriesCompensator = WiresFactory.
                eINSTANCE.createSeriesCompensator();
            ...//set properties here
        } else {
            ShuntCompensator shuntCompensator = WiresFactory.
                eINSTANCE.createShuntCompensator();
            ...//set properties here
        }
    }
}
```

4 模型合法性验证

对于像“每个变电站里至少有一个电压等级”这样的约束,EMF 提供了一套验证框架用于解决该问题,EMF 验证框架是 EMF 的一个重要部分,通过它可验证生成的模型是否满足模型定义的约束,基本约束包括:①实际的属性和引用的实例数量在模

型中定义的边界范围内;②属性的数据类型符合模型定义;③任何交叉引用的对象都包含在资源中。

实际应用时,使用 Diagnostician 类的 validate 方法验证 EObject 是否满足约束,验证结果保存在 Diagnostic 中,关键代码如下。

```
Diagnostic diagnostic = Diagnostician.INSTANCE.
```

```
validate(eObject);
```

```
if (diagnostic.getSeverity() == Diagnostic.ERROR ||
    diagnostic.getSeverity() == Diagnostic.WARNING) {
```

```
    System.err.println(diagnostic.getMessage());
```

```
}
```

5 算例

为了验证本文方法的有效性,本文对国内常见 EMS 提供的 CIM XML 文件进行了解析和合法性验证的测试。测试条件和结果分别如表 1—表 3 所示。表 2 中,编号为 3 的 XML 文件由于扩展信息较多,用在字符串比较上的时间较多,因此解析所用时间较长。测试结果表明,本文方法能够快速地解析不同厂家、不同版本的 CIM XML 文件,对于文件中存在的违反约束的情况也能正确检测出来,说明了本文方法的有效性。

表 1 测试条件

Table 1 Test conditions

操作系统	中央处理器	Eclipse 版本
Win7	Intel i3-2310m, 2.10 GHz	20110615-0604
EMF 版本	内部 CIM 版本	JDK 版本
2.6.0	11	1.6

表 2 XML 文件解析测试结果

Table 2 Test results of XML files parsing

编号	EMS	CIM 版本	大小/kbit	耗时/s
1	GE xa21	11	11 352	2.8
2	Open3000	10	8 264	2.1
3	Open3000	10	18 923	15.5
4	CC200A	10	49 216	12.3

表 3 合法性验证测试结果

Table 3 Test results of validity checking

编号	验证结果
1	786 个 Analog 没有设置 AnalogValue 属性
2	12 个 HydroGeneratingUnit 没有设置 MemberOf_HydroPowerPlant 属性
3	20 180 个 Measurement 没有设置 Unit 属性
4	1 676 个 MeasurementValue 没有设置 MeasurementValueSource 属性

6 结语

本文针对开发基于 CIM 的系统的要求,提出了一种解决方案,该方案具有通用性、灵活性、编程工

作量小等特点。

值得指出的是,基于本文的建模技术,电网分析、仿真系统可以使用较新的 CIM 版本作为底层数据结构,而无须担心 CIM 版本向后兼容的问题。

参考文献

- [1] 张慎明,刘国定. IEC 61970 标准系列简介[J]. 电力系统自动化, 2002, 26(14): 1-6.
ZHANG Shenming, LIU Guoding. Introduction of standard IEC 61970[J]. Automation of Electric Power Systems, 2002, 26(14): 1-6.
- [2] 刘崇茹,孙宏斌,张伯明,等. 基于 CIM XML 电网模型的互操作研究[J]. 电力系统自动化, 2003, 27(14): 45-48.
LIU Chongru, SUN Hongbin, ZHANG Boming, et al. An investigation on a common information model for energy management system [J]. Automation of Electric Power Systems, 2003, 27(14): 45-48.
- [3] 全国电力系统控制及其通信标准化技术委员会 EMS-API 工作组. 国内第 4 次 EMS-API 互操作实验介绍[J]. 电力系统自动化, 2004, 28(16): 1-4.
- [4] 吴文传,孙宏斌,张伯明,等. 基于 IEC 61970 标准的 EMS/DTS 一体化系统的设计与开发[J]. 电力系统自动化, 2005, 29(4): 23-27.
WU Wenchuan, SUN Hongbin, ZHANG Boming, et al. Design of the integrated EMS/DTS system based on IEC 61970[J]. Automation of Electric Power Systems, 2005, 29(4): 23-27.
- [5] 邓显达,何光宇,陈颖,等. 上海电网基于 Java 反射机制的 CIM 导入[J]. 电力系统自动化, 2007, 31(18): 21-25.
DENG Xianda, HE Guangyu, CHEN Ying, et al. CIM lead-in based on Java reflection mechanism in AEMS of Shanghai power grid[J]. Automation of Electric Power Systems, 2007, 31(18): 21-25.
- [6] 王治男,吴文传,张伯明,等. 基于 IEC 61970 的 CIS 服务与 SVG 的研究和实践[J]. 电力系统自动化, 2005, 29(22): 60-63.
WANG Zhinan, WU Wenchuan, ZHANG Boming, et al. Study and implementation of CIS and SVG based on IEC 61970[J]. Automation of Electric Power Systems, 2005, 29(22): 60-63.
- [7] 章坚民,徐爱春,李海翔,等. 基于 SVG/XML/CIM 的变电站自动化工程配置系统[J]. 电力系统自动化, 2004, 28(14): 54-57.
ZHANG Jianmin, XU Aichun, LI Haixiang, et al. An automatic engineering configuration system for substation automation based on SVG/XML/CIM [J]. Automation of Electric Power Systems, 2004, 28(14): 54-57.
- [8] 董越,孙宏斌,吴文传,等. EMS 中公共信息模型的导入/导出技术[J]. 电力系统自动化, 2002, 26(3): 10-14.
DONG Yue, SUN Hongbin, WU Wenchuan, et al. Import/export technique on common information model(CIM) in EMS [J]. Automation of Electric Power Systems, 2002, 26(3): 10-14.
- [9] MCMORRAN A W, AULT G W, MORGAN C, et al. A common information model (CIM) toolkit framework implemented in Java[J]. IEEE Trans on Power Systems, 2006, 21(1): 194-201.
- [10] MCMORRAN A W, AULT G W, ELDERS I M, et al. Translating CIM XML power system data to a proprietary format for system simulation [J]. IEEE Trans on Power Systems, 2004, 19(1): 229-235.
- [11] STEINBERG D, BUDINSKY F, PATERNOSTRO M, 等. Eclipse Modeling Framework 2.0 中文版[M]. 2 版. 战晓苏,邓中亮,傅效群,等译. 北京:清华大学出版社, 2010.
- [12] 王赞华,陈蔚威. 模型驱动开发方法的应用研究[J]. 计算机工程, 2006, 32(13): 63-65.
WANG Yunhua, CHEN Weiwei. Application research on advanced model-driven developing method [J]. Computer Engineering, 2006, 32(13): 63-65.
- [13] W3C. Resource description framework (RDF) model and syntax specification [EB/OL]. [2011-07-15]. <http://www.w3c.org/TR/1999/REC-rdf-syntax-19990222>.

董树锋(1982—),男,通信作者,博士,讲师,主要研究方向:电力系统状态估计和优化运行。E-mail: dongshufeng@gmail.com

何光宇(1972—),男,博士,副教授,主要研究方向:电力系统经济运行、优化理论在电力系统中的应用。

刘凯诚(1989—),男,博士研究生,主要研究方向:电力系统优化运行。

System Development Based on Common Information Model Using Eclipse Modeling Framework

DONG Shufeng¹, HE Guangyu², LIU Kaicheng², ZHANG Wangjun³, WANG Zhihua³

(1. College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China;

2. State Key Laboratory of Control and Simulation of Power Systems and Generation Equipments, Tsinghua University, Beijing 100084, China; 3. Shanghai Power Grid Corporation, Shanghai 200122, China)

Abstract: A simple, flexible and universal solution using the Eclipse modeling framework (EMF) for developing systems based on the common information model (CIM) is proposed. The scheme contains three aspects. Firstly, the Java code of package, class and object relations defined in CIM are automatically generated. Secondly, the CIM extendable markup language (XML) file can be efficiently scanned through streams, objects are formed in the memory during the scanning process, the relations between objects are constructed after the scan is finished. The differences of different energy management system (EMS) suppliers and different CIM versions are shielded, so good compatibility is reached. Thirdly, the constraints of objects defined in CIM are validated. The validity of the solution is testified by test results of CIM data of several real systems.

This work is supported by National Natural Science Foundation of China (No. 51207136).

Key words: Eclipse modeling framework (EMF); common information model (CIM); file parsing