



中华人民共和国国家标准化指导性技术文件

GB/Z 24357—2009

地理信息 元数据 XML 模式实现

Geographic information—Metadata—XML schema implementation

(ISO/TS 19139:2007,MOD)

2009-09-30 发布

2009-12-01 实施

中华人民共和国国家质量监督检验检疫总局
中国国家标准化管理委员会 发布

中 华 人 民 共 和 国
国家标准化指导性技术文件
地理信息 元数据 XML 模式实现
GB/Z 24357—2009

*

中国标准出版社出版发行
北京复兴门外三里河北街16号
邮政编码:100045

网址 www.spc.net.cn

电话:68523946 68517548

中国标准出版社秦皇岛印刷厂印刷
各地新华书店经销

*

开本 880×1230 1/16 印张 5.5 字数 164 千字
2010 年 1 月第一版 2010 年 1 月第一次印刷

*

书号: 155066 · 1-39448 定价 72.00 元

如有印装差错 由本社发行中心调换
版权专有 侵权必究
举报电话:(010)68533533

目 次

前言	III
引言	IV
1 范围	1
2 一致性要求	1
3 规范性引用文件	1
4 术语和定义	1
5 符号和缩略语	2
5.1 首字母缩略语	2
5.2 命名空间缩略语	2
5.3 UML 模型的关系	3
5.4 UML 模型的构造型	3
6 要求	4
6.1 gmd 概述	4
6.2 基于规则的编码	4
6.3 质量	5
6.4 Web 实现	5
6.5 外部 XML 实现的利用	5
6.6 多态	5
6.7 应用模式规则	5
7 对地理信息国家标准中 UML 模型的扩展	6
7.1 概述	6
7.2 针对万维网环境的扩展	6
7.3 针对基于元数据的地理空间信息传输的扩展	6
8 编码规则	12
8.1 编码规则概述	12
8.2 缺省 XML 类类型编码	12
8.3 XML 类全局元素的编码	14
8.4 XML 类特性类型的编码	15
8.5 特殊情况的编码	16
8.6 XML 命名空间包的编码	29
8.7 XML 模式包的编码	30
9 编码说明	32
9.1 编码说明概述	32
9.2 XML 命名空间	32
9.3 gmd 命名空间	33
9.4 gss 命名空间	37
9.5 gts 命名空间	38
9.6 gsr 命名空间	39

9.7 gco 命名空间	40
9.8 gmx 命名空间	49
9.9 从概念模式到 XML 文件实例	54
附录 A (规范性附录) 抽象测试套件	56
A.1 一致性测试工具概述	56
A.2 一致性要求——约束	56
A.3 一致性要求——扩展	57
A.4 一致性要求——限制	58
附录 B (规范性附录) 扩展内容的数据字典	59
B.1 数据字典概览	59
B.2 元数据扩展内容的数据字典	60
B.3 代码表与枚举	65
附录 C (资料性附录) 地理信息元数据 XML 资源	66
C.1 本指导性技术文件中的 XML 模式	66
C.2 本指导性技术文件外定义的 XML 模式	66
C.3 其他资源	66
附录 D (资料性附录) 实现示例	68
D.1 说明	68
D.2 元数据最小集示例	68
D.3 核心元数据示例	69
D.4 “头排,G49E023009”元数据	73
D.5 针对 Web 环境扩展的用法示例	82

前 言

本指导性技术文件修改采用 ISO/TS 19139:2007《地理信息 元数据 XML 模式实现》(英文版),并作了如下改动:

(1) 编写方法执行了 GB/T 1.1—2000《标准化工作导则 第1部分:标准的结构和编写规则》、GB/T 20000.2—2001《标准化工作指南 第2部分:采用国际标准的规则》的要求。

(2) 引用国际标准名称或编号的改变:

- a) 将“本国际标准”改为“本指导性技术文件”;
- b) 将“ISO 19100 系列标准”改为“地理信息国家标准”;
- c) 将下列国际标准名称和编号用相应的国家标准名称或编号替代:

原国际标准编号	替代的国家标准编号
ISO 19115:2003	GB/T 19710—2005
ISO 8601:2001(E)	GB/T 7408
ISO 19108	GB/T 22022—2008
ISO 19105	GB/T 19333.5—2003
ISO 19107	GB/T 23707—2009

(3) 删除了原国际标准的前言。

(4) 本指导性技术文件的引言采用了原标准的引言,并作了修改。

(5) 增加了术语“元数据”及其定义。

(6) 删除了原文的 6.6、7.3、9.3.18、9.8.6、B.2.2,以及原文 9.8.8、9.8.9 和 D.5 中关于多语种支持的内容,并相应调整了章节号和图号。

(7) 表 1 中的代码取值遵照 GB/T 19710—2005 进行了调整。

(8) 修改了资料性附录 D 中的实例。

本指导性技术文件的附录 A 和附录 B 为规范性附录,附录 C 和附录 D 为资料性附录。

本指导性技术文件由国家测绘局提出。

本指导性技术文件由全国地理信息标准化技术委员会(SAC/TC 230)归口。

本指导性技术文件主要起草单位:国家信息中心、中国标准化研究院、国家基础地理信息中心。

本指导性技术文件主要起草人:徐枫、宦茂盛、石雯雯、李媛、常娜、王子亮、李小林、周旭、邓跃进。

引 言

GB/T 19710—2005《地理信息 元数据》详细说明了描述地理信息元数据的重要性。GB/T 19710—2005通过定义一系列元数据元素并建立一组共同的术语、定义和扩展规则,为各种地理信息的描述提供了一个结构体系。GB/T 19710—2005 是一个基础标准,为地理信息相关的元数据提供了一个全局视图,但没有定义相应的编码规则。

GB/T 19710—2005 没有提供编码规则,因此在地理信息元数据的具体实现时,会因为对各种元数据规则的理解不一致而产生差异。为了便于 GB/T 19710—2005 实现的标准化,本元数据实现规范提供了一套明确的基于规则的编码方法,用于执行 GB/T 19710—2005。本指导性技术文件提供了一组 XML 模式以保证互操作性,这些模式为描述、检验和交换关于地理信息数据集、数据集系列、地理要素、要素属性、要素类型等的元数据提供了统一的规范。

GB/T 19710—2005 定义了地理信息领域通用的元数据。关于地理信息数据类型和地理信息服务的更详细的元数据,在其他地理信息国家标准和用户扩展(GB/T 19710—2005)中定义。本指导性技术文件旨在提供通用元数据的实现规则,在需要引用其他地理信息国家标准的地方也一并考虑。

ISO 19118《地理信息 编码》说明了基于 UML 模式生成编码规则的需求和基于 XML 的编码规则,并对 XML 进行了介绍。本指导性技术文件利用 ISO 19118 中定义的编码规则,规定如何从 GB/T 19710—2005中的 UML 模型得出 XML 模式的技术细节。

地理信息 元数据 XML 模式实现

1 范围

本指导性技术文件规定了地理信息元数据 XML(gmd)的编码方法以及 GB/T 19710—2005 的 XML 模式实现。

本指导性技术文件适用于通过网络或离线进行地理信息元数据的共享和交换。

2 一致性要求

本指导性技术文件的一致性应使用附录 A 规定的所有相关测试进行检查。测试的框架、概念和方法以及声称与本指导性技术文件一致需达到的标准都在 GB/T 19333.5—2003 中说明。

3 规范性引用文件

下列文件中的条款通过本指导性技术文件的引用而成为本指导性技术文件的条款。凡是注日期的引用文件,其随后所有的修改单(不包括勘误的内容)或修订版均不适用于本指导性技术文件,然而,鼓励根据本指导性技术文件达成协议的各方研究是否可使用这些文件的最新版本。凡是不注日期的引用文件,其最新版本适用于本指导性技术文件。

GB/T 7408 数据元和交换格式 信息交换 日期和时间表示法(GB/T 7408—2005,ISO 8601:2000,IDT)

GB/T 19333.5—2003 地理信息 一致性与测试(ISO 19105:2000,IDT)

GB/T 19710—2005 地理信息 元数据(ISO 19115:2003,MOD)

GB/T 22022—2008 地理信息 时间模式(ISO 19108:2002,IDT)

GB/T 23707—2009 地理信息 空间模式(ISO 19107:2003,IDT)

GB/T 23708—2009 地理信息 地理标记语言(GML)

ISO/IEC 10646 信息技术 通用多字节编码字符集

ISO/TS 19103 地理信息 概念模式语言

ISO 19109 地理信息 应用模式规则

ISO 19110 地理信息 要素编目方法

ISO 19111:2007 地理信息 基于坐标的空间参照

ISO 19117 地理信息 描述

ISO 19118:2005 地理信息 编码

ISO 19138 地理信息 数据质量度量

W3C XMLName XML 命名空间(1999)

W3C XMLSchema-1 XML 模式 第 1 部分:结构,W3C 推荐(2001)

W3C XMLSchema-2 XML 模式 第 2 部分:数据类型,W3C 推荐(2001)

W3C XML 可扩展标记语言 1.0(第 2 版),W3C 推荐(2000)

W3C XLink XML 链接语言(XLink) 1.0,W3C 推荐(2001)

4 术语和定义

下列术语和定义适用于本指导性技术文件。

4.1

命名空间 namespace

通过一个 URI 进行引用的若干名称的集合,这些名称用作 XML 文档中的元素和属性的名称。

[W3C XML]

4.2

包 package

把多个元素组织为组的通用方法。

如:标识信息、元数据实体集信息、限制信息。

[ISO/TS 19103]

4.3

实现 realization

分类器之间的语义关系,其中一个分类器表明约定,另一个保证实施。

4.4

多态 polymorphism

在不同上下文环境中可以对同一事物赋予不同含义或用途的特性,确切地说,某一实体如变量、函数或对象可以具有 1 个以上的形态的特性。

注:多态具有几种不同类型。

4.5

元数据 metadata

关于数据的数据。即数据的标识、覆盖范围、质量、空间和时间模式、空间参照系和分发等信息。

[GB/T 19710—2005]

5 符号和缩略语

5.1 首字母缩略语

UML 统一建模语言(Unified Modelling Language)

URI 统一资源标识符(Universal Resource Identifier)

XCT XML 类类型(XML Class Type)

XCPT XML 类特性类型(XML Class Property Type)

XCGE XML 类全局元素(XML Class Global Element)

XML 可扩展标记语言(Extensible Markup Language)

XPath XML 路径语言(XML Path Language)

XSD XML 模式定义(XML Schema Definition)

XSL 可扩展样式语言(Extensible Style Language)

XSLT XSL 转换(XSL Transformation)

5.2 命名空间缩略语

下面的列表中第 1 列说明通用的命名空间前缀,用于描述该命名空间中的元素,第 2 列为命名空间前缀说明。

下面的列表为本指导性技术文件定义的命名空间:

gco 地理信息通用可扩展标记语言 (Geographic Common Extensible markup language)

(<http://www.isotc211.org/2005/gco>)

gmd 地理信息元数据可扩展标记 (Geographic MetaData extensible markup language)

语言 (<http://www.isotc211.org/2005/gmd>)

gmx	地理信息元数据 XML 模式	(Geographic Metadata XML Schema) (http://www.isotc211.org/2005/gmx)
gss	地理信息空间模式可扩展标记语言	(Geographic Spatial Schema extensible markup language) (http://www.isotc211.org/2005/gss)
gsr	地理信息空间参照可扩展标记语言	(Geographic Spatial Referencing extensible markup language) (http://www.isotc211.org/2005/gsr)
gts	地理信息时间模式可扩展标记语言	(Geographic Temporal Schema extensible markup language) (http://www.isotc211.org/2005/gts)

下面的列表为本指导性技术文件用到的外部命名空间：

gml	地理信息标记语言(Geography Markup Language)
xlink	xml 链接语言(xml linking language)
xs	W3C XML 基本模式 (W3C XML base schemas)

5.3 UML 模型的关系

本指导性技术文件中的图示都采用 ISO/TS 19103 中规定的 UML 作为概念模式语言表示。GB/T 19710—2005中的图 2 也表示了用于说明元数据的 UML 标记法。除此之外,本指导性技术文件还使用如图 1 所示的符号：

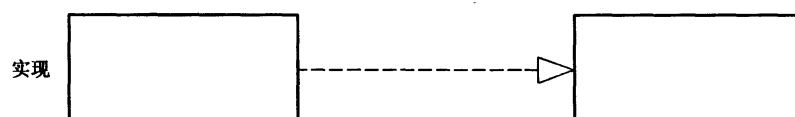


图 1 实现关系的 UML 标记法

连接起点的类(图 1 左侧),应确保执行(或实现)连接终点的类(图 1 右侧)的规格。

5.4 UML 模型的构造型

UML 构造型是对已有的 UML 概念的一种扩展机制(GB/T 19710—2005)。除了 GB/T 19710 中规定的用于描述元数据的构造型外,为了基于规则的 XML 模式编码的需要,本指导性技术文件还定义了一些必要的构造型。

本指导性技术文件第 9 章中给出的 UML 图中的元素可以包括说明 XML 实现的构造型,这些构造型包含在表示 XML 元素或 XML 类型、UML 属性、UML 连接(实现或依赖)以及 UML 包的类中。

本指导性技术文件中使用的类的构造型包括：

- 《《选择》》(《《xs:choice》》):表示可编码为 XML choice 块的类,该类的每一个属性实现为该 choice 的一个元素。
- 《《复合类型》》(《《xs:complexType》》):表示可编码为 XML 复杂类型的类。
- 《《元素》》(《《xs:element》》):表示为 XML 全局元素的类。
- 《《简单类型》》(《《xs:simpleType》》):表示为 XML 简单类型的类。
- 《《简单内容》》(《《xs:simpleContent》》):表示可编码为具有简单内容的 XML 复杂类型的类。

本指导性技术文件中使用的属性的构造型包括：

- 《《属性》》(《《xs:attribute》》):编码为 XML 属性的特性。
- 《《元素属性》》(《《xs:element attribute》》):编码为带有名称和类型的 XML 元素的属性(《《xs:element name="propertyName" type="propertyType"/》》)。
- 《《模式》》(《《pattern》》):该构造型仅适用于内建 XML 类型衍生的 XML 简单类型。具有《《pattern》》构造型的属性对内建简单类型的值域进行进一步限定。

- i) `<<XML 类全局元素>><<XCGE>>`:编码为对某一 XML 全局元素的引用的特性(`<xs:element ref="XCGE"/>`)。

本指导性技术文件中使用的链接的构造型包括:

- j) `<<XML 类类型>><<XCT>>`: (包含在实现关系中) 通过特定外部实现来替代的抽象概念的 `XCT(XML 类类型)`。
- k) `<<XML 类全局元素>><<XCGE>>`: (包含在实现关系中) 通过特定外部实现来替代的抽象概念的 `XCGE(XML 类全局元素)`。
- l) `<<XML 类特性类型>><<XCPT>>`: (包含在实现关系中) 通过特定外部实现来替代的抽象概念的 `XCPT(XML 类特性类型)`。
- m) `<<实现>><<implement>>`: (包含在依赖关系中) 表示 XML 模式的连接起点, 该 XML 模式实现了连接终点中定义的抽象概念。
- n) `<<包含>><<include>>`: (包含在依赖关系中) 连接的起点和终点代表的 XML 模式。连接起点包含(`<xs:include ... />`)连接终点。
- o) `<<引用>><<import>>`: (包含在依赖关系中) 连接的起点和终点代表同一命名空间中的 XML 对象集。连接起点引入(`<xs:import ... />`)连接终点。

本指导性技术文件中使用的包的构造型包括:

- p) `<<XML 模式>><<xmlSchema>>`: 代表一个 XML 模式的包。
- q) `<<XML 命名空间>><<xmlNamespace>>`: 代表同属一个命名空间的 XML 对象集的包。

6 要求

6.1 gmd 概述

在 GB/T 19710—2005 中, 地理信息元数据表示为一组包括一个或多个 UML 类的 UML 包。GB/T 19710—2005 提供了与编码无关的一个地理信息元数据的统一视图。本指导性技术文件则根据 ISO 19118:2005 的规则提供一个基于 XML 模式的 GB/T 19710—2005 的通用实现方法。

本指导性技术文件虽然定义了几个命名空间的内容, 但不包括 XML 命名空间的详细信息。命名空间实际上是一个可用作 XML 文档中元素和属性名的一个名称集合。命名空间用于区别不同模式的名称。命名空间是一个 URI, 本指导性技术文件用到的 URI 均在 5.2 中列出。URI 往往不易于读写和交流, 因此本指导性技术文件在需要指定某一命名空间内容时通常引用通用命名空间前缀。本指导性技术文件定义的主要命名空间是 <http://www.isotc211.org/2005/gmd>, 其前缀为 `gmd`, 代表地理信息元数据可扩展标记语言 (Geographic MetaData extensible markup language)。

XML 模式提供了许多以交换为目的的信息组织方法。ISO 19118:2005 定义了把地理信息国家标准中定义的 UML 概念模式转换为 XML 模式的一组规则。即使在 ISO 19118:2005 的较为严格的限制规则下, 创建特定 XML 模式的时候仍然有多种选择。第 7 章、第 8 章、第 9 章中描述了对第 3 章列出的 GB/T 19710—2005 中的 UML 概念模式和其他相关地理信息国家标准中的 UML 模型进行 XML 编码的详细过程。附录 C 和附录 D 中给出了本指导性技术文件的 XML 资源描述和元数据实例示例文档。

在深入编码细节之前, 理解建立 `gmd` 时为什么需要使用某些编码规则非常重要。理解这些规则可以对 `gmd` 的功能、限制以及最佳用法更清楚。`gmd` 的主要目标包括保证与其他地理信息国家标准的互操作性, 保证可预见性、可扩展性以及可用性。6.2 和 6.7 将进一步讨论这些目标。

6.2 基于规则的编码

本 XML 模式实现是按照 ISO 19118:2005 的要求, 根据地理信息国家标准中的 UML 模型建立的基于规则的编码。通过该方法, 达到 6.1 中描述的两个目标。首先, 得到的 `gmd` 模式直接基于地理信息国家标准, 从而增加了互操作性的机会; 其次, 生成的模式是可预期的, 因为任何类、属性、关联等的编码与其他同类型的 UML 元素的编码方法一样。

本指导性技术文件虽未做详细讨论,但基于规则的编码使自动或半自动方式生成 XML 模式成为可能。

6.3 质量

XML 模式实现的质量高低体现在产生的 XML 模式的简单性和据此生成的 XML 文件的易读性。地理信息国家标准,尤其是 GB/T 19710—2005 的结构复杂性,意味着不大可能提供地理信息元数据的简单 XML 模式。但只要定义了编码规则,用户就可以建立或理解以 GB/T 19710—2005 的 UML 模型为基础的 XML 实例文档的内容。另外,开发人员了解编码规则后即可对 GB/T 19710—2005 的 UML 模型进行 XML 模式实现。

gmd 质量的另一方面体现在完整性。本指导性技术文件对 GB/T 19710—2005 的整个 UML 模型进行编码,不考虑 gmd 的特定用途或使用 gmd 的特定应用模式。

6.4 Web 实现

6.1 提到的其中一个目标是可用性。gmd 的可用性主要关注地理信息元数据的交换,这种需求在 web 环境中尤为常见。由于基于 gmd 创建地理信息元数据实例文档可以在网络上传输,gmd 很多方面的设计都旨在支持通过因特网或万维网的数据传输。

6.5 外部 XML 实现的利用

另外一个有助于互操作性和可用性的设计原则是利用现有 XML 模式。如果一个对地理信息元数据相关地理信息系列的某一部分进行编码的 XML 模式标准已经存在,则优先将该 XML 模式标准并入到 gmd 中。如果 gmd 直接是用外部 XML 模式,将增强互操作性。也可能已经存在能够处理与外部 XML 模式标准一致的 XML 实例文件的软件。此外,如果外部模式进行了充分设计,可能会比根据一系列编码规则生成的 XML 模式更高效,从而有助于提高可用性。

尽管使用已有实现有明显的好处,但外部 XML 模式不应与 gmd 的设计原则冲突。例如,如果某一外部 XML 模式以一种繁琐、难用的方式实现了地理信息国家标准的某一部分,则不能并入到 gmd 中。

6.6 多态

术语“多态”在第 4 章进行了定义。概括地说,多态意味着表现出多种形态的能力。就本指导性技术文件而言,多态使用户可以细化元数据以满足自身的需求。例如,GB/T 19710—2005 在 CI_负责单位(CI_ResponsibleParty)类中有一个字符串类型的属性 individualName,但在某一机构中,个人信息可以通过一种更细化的方法进行描述(如名、中间名、姓)。多态可以让开发人员在其自己的命名空间中扩展 individualName 这一较为通用的格式,但仍然充分利用 gmd 并为该机构之外的用户提供可用和可读的实例文档。gmd 支持多态的特性主要来自于 8.4 描述的特性类型编码。

6.7 应用模式规则

ISO 19109 定义了应用模式规则,由两类与元数据相关的模型组成。

——通用要素模型,确定元数据和质量元素与地理要素相关的具体途径;

——两种交换模型,传统的数据传输模型和互操作模型,每一种模型都包含元数据与其描述对象的相互关系。

关于 ISO 19109 通用要素模型,可以为要素类型定义以下两类属性:

——元数据属性(如 GF_MetadataAttributeType 的实例),其数据类型为 MD_Metadata 或其子类。

——质量属性(如 GF_QualityAttributeType 的实例),其数据类型为 DQ_Element 或其子类。

在针对应用模式的规则中使用地理信息元数据的 XML 模式,主要是使用定义的元数据和质量属性数据类型的 XML 模式。根据 GB/T 23708—2009 的要素类型编码方法在 9.9.2 中描述。

关于 ISO 19109 交换模型,互操作模型是基于事务方式的数据交换,是为大量包含简单交换的事务操作设计的。相反,传输模型是针对包含大量格式化数据的较少量事务操作设计的。在为用户群体内部或相互间提供更一致的地理数据传输方面(如传输模型),GB/T 19710—2005 元数据概念模式和本

指导性技术文件无疑是一个起点。

此外,应理解互操作模型适用于用户系统和服务提供者之间的交互,而且交互完全由服务接口决定。虽然服务接口可以不需要使用特定的模式,但强烈建议在信息共享的圈子内尽可能采用地理信息元数据的 XML 模式。通过传输的互操作交换在标准化方面需要更加深入。这正是 7.3 描述的“针对基于元数据的地理空间信息传输的扩展”的目的。

7 对地理信息国家标准中 UML 模型的扩展

7.1 概述

GB/T 19710—2005 提供了一个公认的与编码无关的地理信息元数据视图,本指导性技术文件则提供了一个通用的 GB/T 19710—2005 的 XML 模式编码实现。由于与实现技术相关,可能需要对地理信息国家标准中的 UML 模型进行某些扩展。这些扩展的目的可能不尽相同,但都主要是满足第 6 章的要求。如针对互操作性、易用性、万维网环境等。第 7 章给出的 UML 图就是为了满足 GB/T 19710—2005 及其相关的地理信息国家标准实现 XML 模式编码所需要的扩展。

7.2 针对万维网环境的扩展

为了便于使用 XML 文档,需要对 ISO/TS 19103 中的 `CharacterString` 类进行一些扩展。这些扩展均针对 XML 文档的典型处理环境——万维网。图 2 定义了万维网环境中描述元素需要的元数据。相应的数据字典在 B.2.1 中给出。

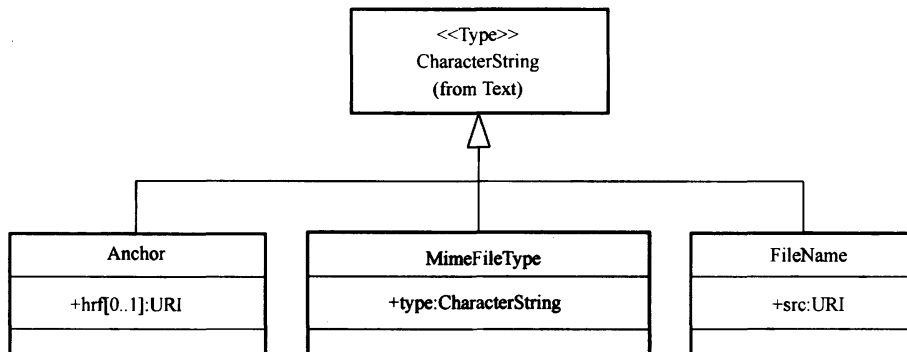


图 2 针对万维网环境的扩展

`Anchor` 类用于支持超链接功能,并用来以类似于网络连接方式实现 GB/T 19710—2005 中的 `RS_Identifier` 和 `MD_Identifier` 类。因为 `Anchor` 是 `CharacterString` 的子类,`RS_Identifier` 和 `MD_Identifier` 的属性 `code` 可以用一个 `Anchor` 实例来实现,因而提供了一个到代码信息位置的引用。

`FileName` 类用于明确引用某一外部文件,这对于 GB/T 19710—2005 中包含 `fileName` 属性的 `MD_BrowseGraphic` 类很有用。该属性的值应该采用人可以理解的文件名,而 `FileName` 类另外的 `src` 属性可以给出一个计算机可理解的该文件所在位置的绝对路径。

`MimeType` 类用于通过 MIME 媒体类型名称和子类型名称识别文件类型。这对于 GB/T 19710—2005 的 `MD_BrowseGraphic` 类的 `fileType` 属性有用,其值若是“JPEG”,则 `MimeType` 类可以表示为计算机可以理解的“image/jpeg”。

7.3 针对基于元数据的地理空间信息传输的扩展

7.3.1 数据集与聚集数据集的传输

图 3 引用了 GB/T 19710—2005 的图 3,但把重点放在元数据驱动的地理空间信息传输中需要的类和关联上。数据集 (`DS_Dataset`) 可以是聚集 (`DS_Aggregate`) 的一部分,该聚集可以是另一聚合体的子集或超集。聚集和数据集都被链接到一或多组元数据元素 (`MD_Metadata`)。

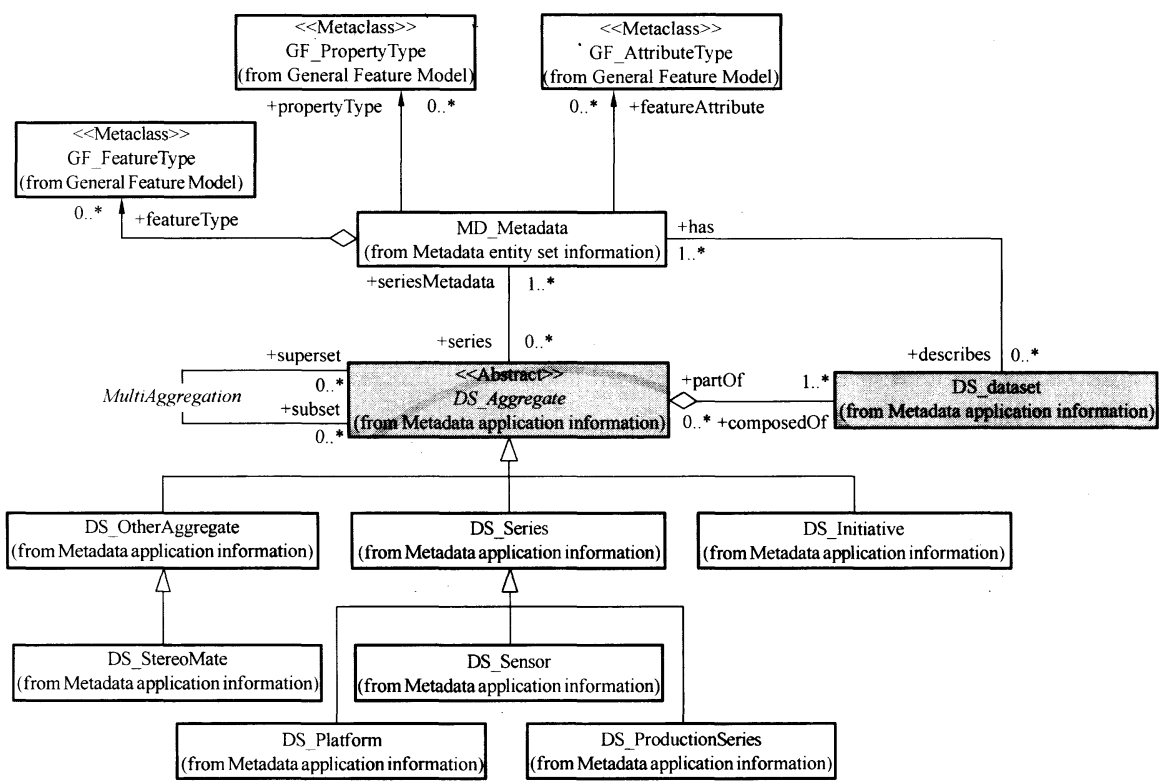


图 3 数据集、聚集及其元数据

基于元数据的地理信息传输需要对 GB/T 19710—2005 的扩展如图 4 所示。

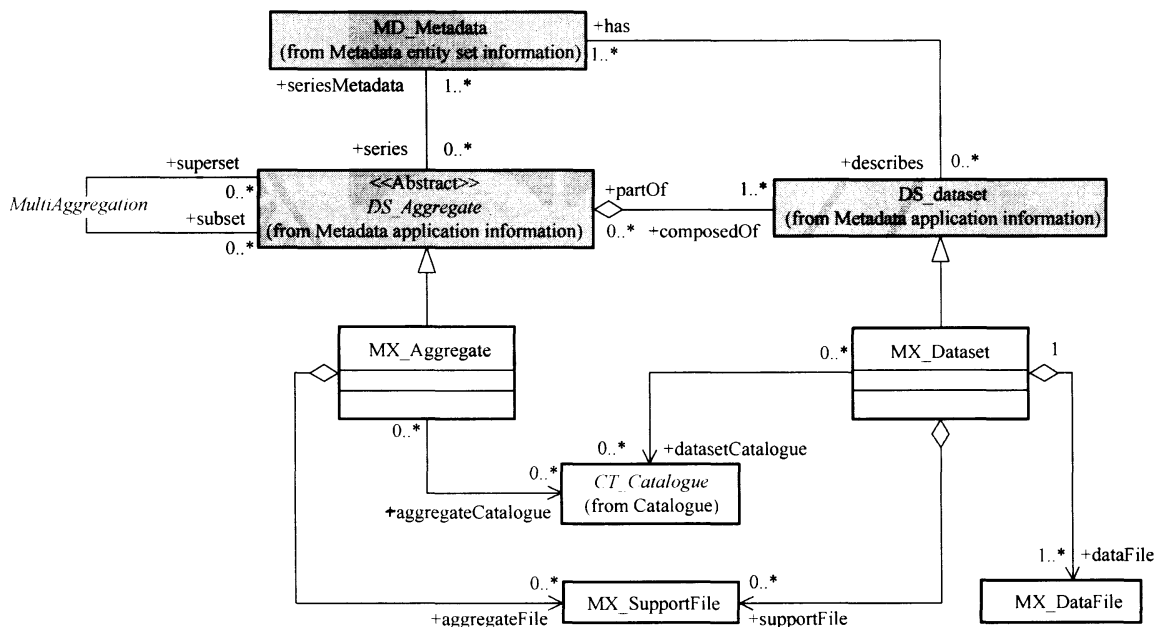


图 4 通用交换结构

为了提供一个通用的传输交换机制，引入两个新的概念：传输数据集（MX_Dataset）和传输聚集（MX_Aggregate）。

传输交换环境中，数据集数据按数据文件（MX_DataFile）组织。传输数据集和聚集可能伴随支持

文件(MX_SupportFile),这些支持文件可能包含利用前者时所需的资源或补充信息。数据文件及支持文件在 7.3.3 中描述。

- 实际上,使用数据集或聚集需要的信息不限于元数据,尤其是:
- 元数据引用了要素和图示目录,但并不嵌入这些信息;
 - 元数据实例引用的信息,如代码表、度量单位和坐标参照系等在数据使用时都需要。

所有这些信息都可以通过外部的在线注册表进行管理,但通常在传输交换模型下,与传输数据集和传输聚集中同时提供这些信息也是必要的。目录(catalogue)(CT_Catalogue,在 7.3.4 中详细描述)的抽象概念直接对应利用数据集和聚集及其元数据中需要的资源。因此,目录用于传输交换,并与传输数据集(MX_Dataset)和传输聚集(MX_aggregate)相关联。

7.3.2 对聚集的管理

图 3 所示的 DS_Aggregate 的分类完全是概念性的。在数据传输模型下,数据集聚集所遵循的限制(如传输介质的容量)和需求要超出地理空间聚合的理想设计。传输交换机制是建立在传输聚集的基础上的,但地理空间聚合的基本特性可以通过扩展图 5 所示的 MD_ScopeCode 代码表利用 MD_Metadata 的 hierarchyLevel 属性来表达。

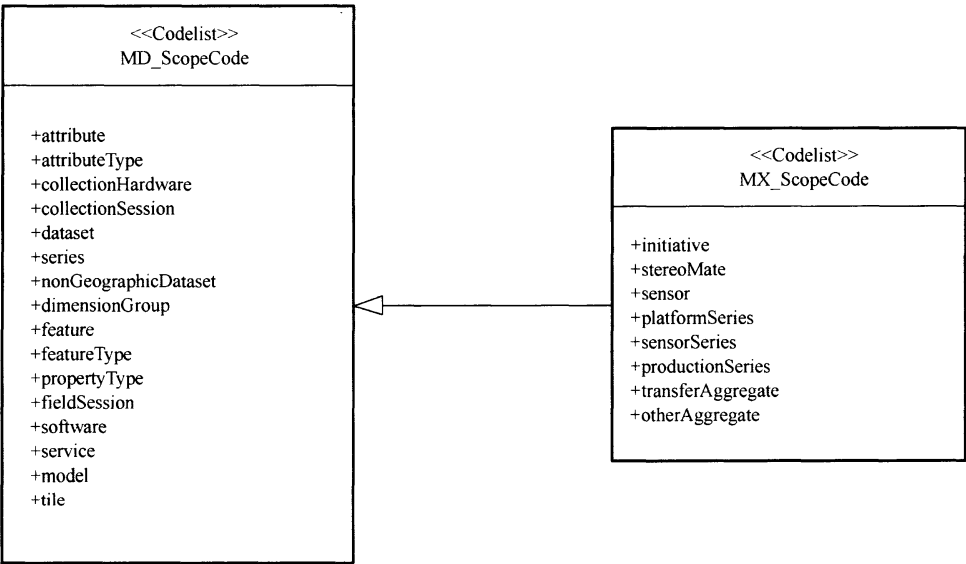


图 5 扩展的范围代码

注意: transferAggregate 值表示聚集只有在实际的传输交换环境下才具有意义。

7.3.3 传输文件

图 6 描述了支持文件和数据文件的概念。数据文件可能与一个或多个要素类型相关。每一个数据文件有其自身的格式,格式的信息会在数据集的元数据中说明。(也可以在聚合体的元数据中说明格式,但只适用于整个聚合体的格式都相同的情况。)

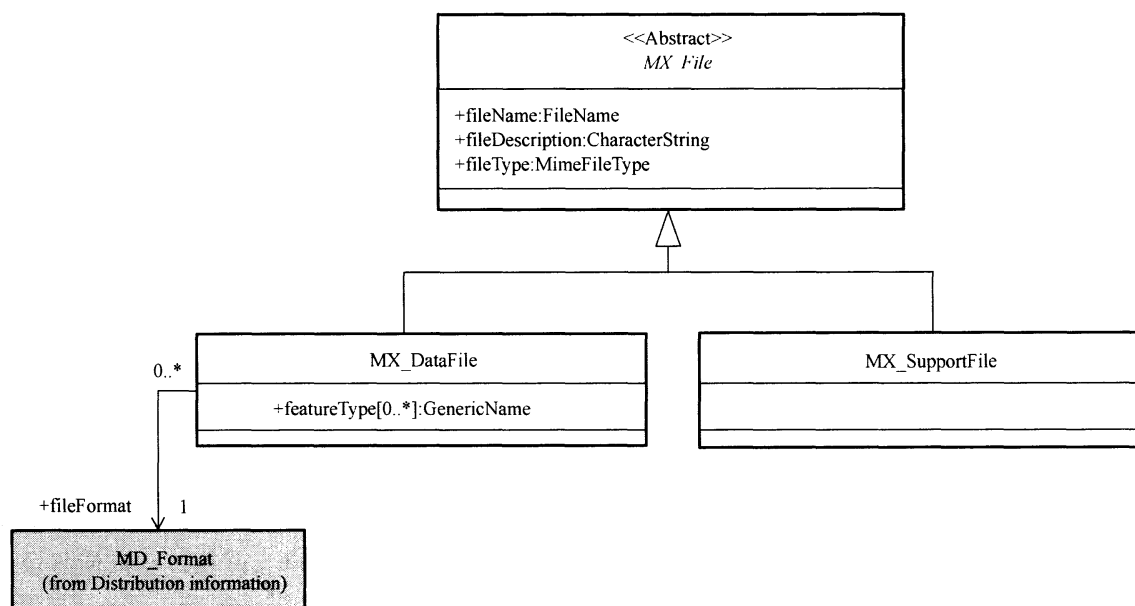


图 6 传输文件

7.3.4 目录

7.3.4.1 一般概念

图 7 描述了目录 (CT_Catalogue) 的概念,就本指导性技术文件而言,目录是地理空间信息标准传输交换的一个关键组成部分。其设计思想来自于 ISO19110 定义的要目目录。

- 增加了几个可选属性 (包括语种、字符集和语区) 以满足文化和语种适应性方面的需求;
- 考虑了针对要素目录的 FC_FeatureCatalog 类的几个属性。

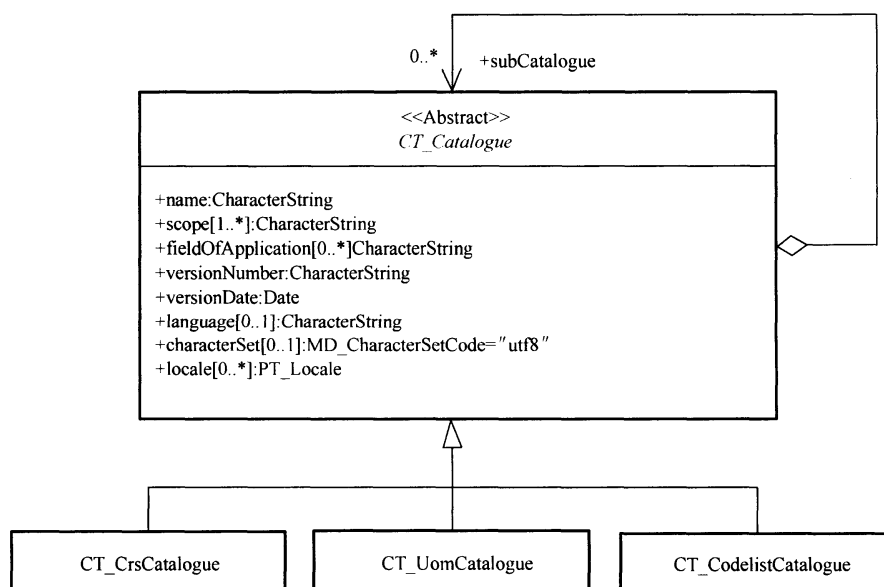


图 7 各种目录

本指导性技术文件提供 3 个具体目录的概念模式,包括:

- 坐标参照系(CRS)目录(CT_CrsCatalogue)包含坐标参照系及其组成部分 (如基准、坐标系统等)的说明。在传输交换环境中(详见 7.3.1),一个 CRS 目录包含传输数据集或传输数据聚

集中使用的具体坐标参照系的说明。

- 度量单位目录(UoM)目录(CT_UomCatalogue)包含度量单位的说明。在传输交换环境中(详见 7.3.1),UoM 目录包括传输数据集或传输数据聚集中使用的具体度量单位的说明。
- 代码表目录(CT_CodelistCatalogue)包含代码表的描述,包括其名称、定义及其值的说明。在传输交换环境中(详见 7.3.1),代码表目录包括传输数据集或传输数据聚集中使用的具体代码表的说明。

这些目录的详细说明分别在 7.3.4.2 和 7.3.4.4 中给出。

作为协调地理信息国家标准中目录概念的基础,目录的抽象概念已经有定义,如 PF_PortrayalCatalogue(ISO 19117)和 FC_FeatureCatalogue(ISO 19110)。ISO 19110 和 ISO 19117 的 XML 模式实现虽不在本指导性技术文件的范围之内,但希望要素和图示的 XML 目录是可传递的,就像其概念(PF_PortrayalCatalogue 和 FC_FeatureCatalogue)是从 CT_Catalogue 继承而来一样。

目录的抽象概念也可以作为建立 ISO 19138 数据质量度量目录的基础。这可以使一个质量度量目录作为这里定义的传输数据集和聚集的一部分进行传输。总的来说,对于任何以在线或可以通过在线注册表的方式管理的信息集都建议扩展目录概念和使用目录,这样目录可以支持对传输数据和聚集的利用。

7.3.4.2 UoM 目录

图 8 所示的度量单位目录(CT_UomCatalogue)聚集一组度量单位,这些单位在 GB/T 23708—2009 中定义。

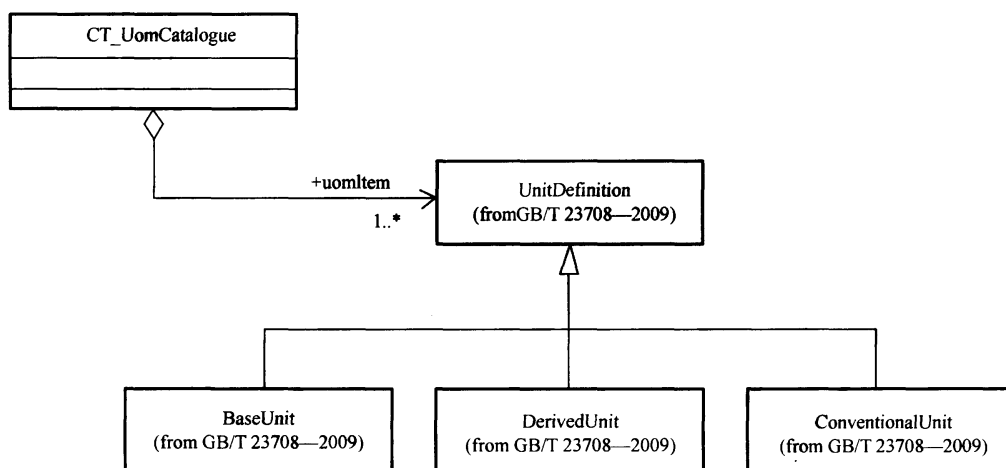


图 8 UoM 目录

7.3.4.3 CRS 目录

CRS 目录的概念(CT_CrsCatalogue)如图 9 所示。CRS 目录没有专门的属性,但聚集不同类型的基准,都是继承自 ISO 19111:2007 定义的基础类。

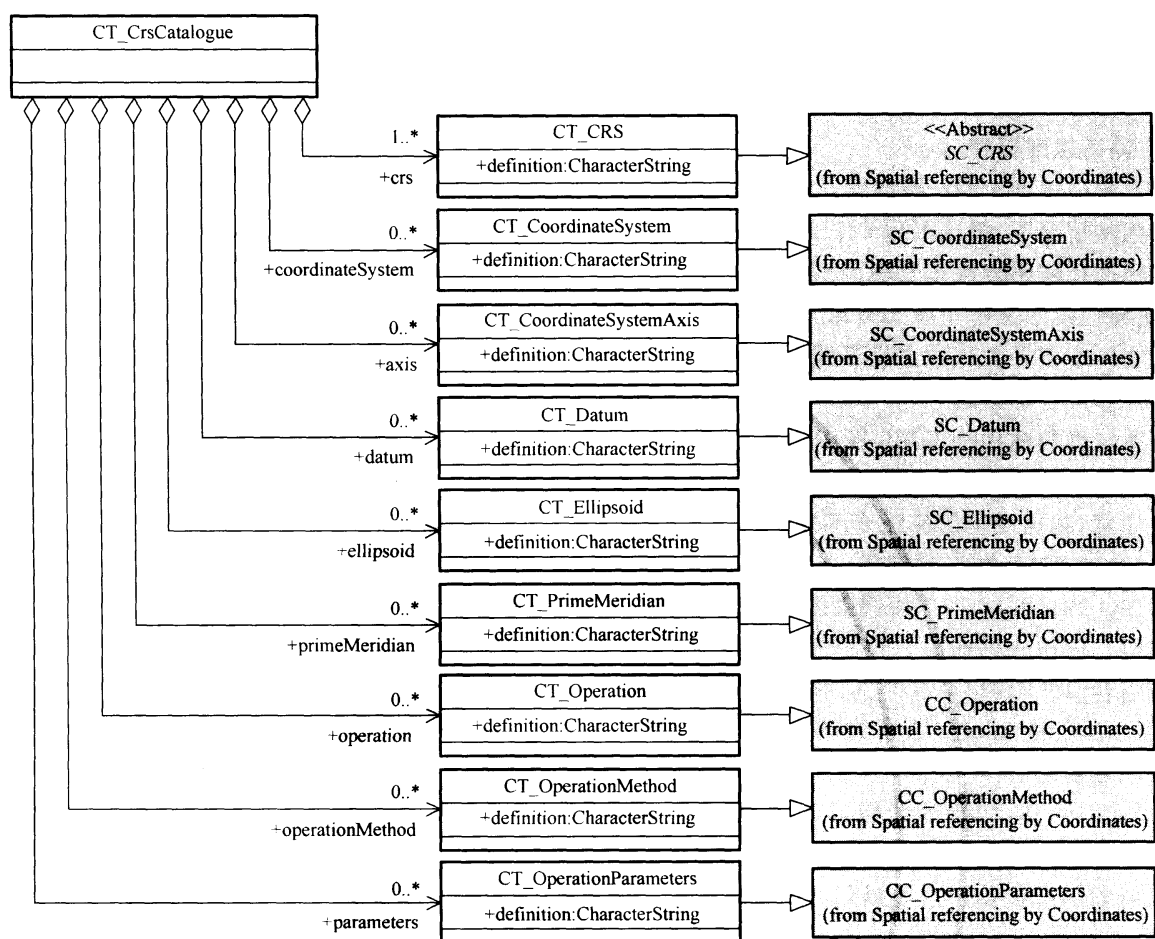


图 9 CRS 目录

7.3.4.4 代码表目录

代码表目录(**CT_CodelistCatalogue**)的概念如图 10 所示。代码表目录没有专门的属性,但聚集一种类型的基于通用的目录项概念的代码表(**CT_Codelist**)。代码表也聚集一个与代码表值一致的简单类型的项(**CT_CodelistValue**)。

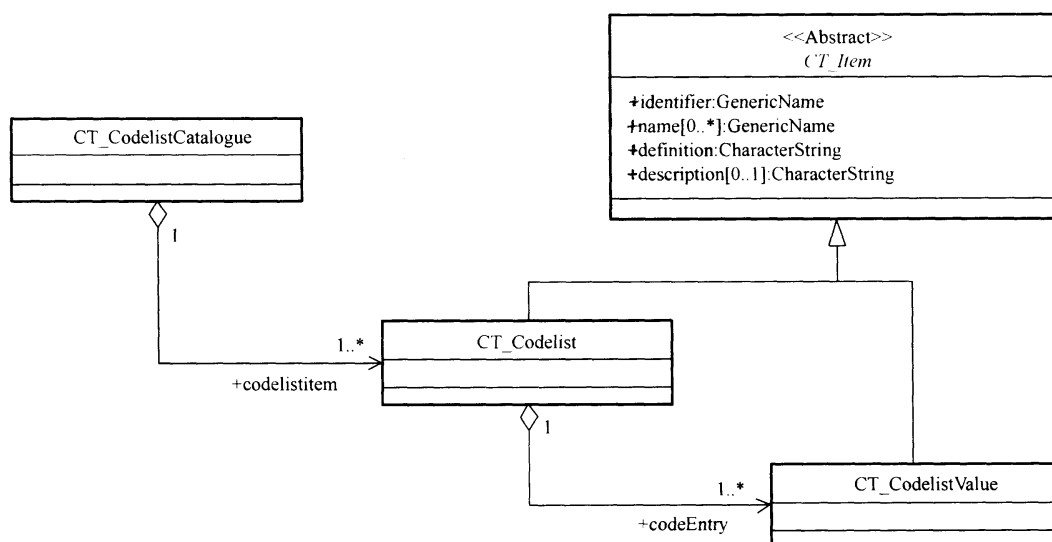


图 10 代码表目录

引入目录项概念(**CT_Item**)可简化代码表目录中的组成部分,但强烈建议用来定义新类型的目录

组成部分。

8 编码规则

8.1 编码规则概述

下面描述把 UML 转换成 XML 模式的通用规则,这些规则与 ISO 19118:2005 定义的规则是一致的。某些情况下,根据 ISO 19118:2005 可以通过多种方法把 UML 转换为 XML 模式,本指导性技术文件阐明选择何种方法来实现该规范。本指导性技术文件不说明各个类的背景情况以及用于数据交换(本指导性技术文件中指元数据交换)的这些类的编码块,因为这些内容在 ISO 19118:2005 中进行了详细说明。本指导性技术文件是基于 ISO 19118:2005 的编码规则,如果熟悉 ISO 19118:2005,可以增强对第 8 章内容的理解。

默认规则适用于 UML 图中指定类型的每一个元素,除非在第 9 章有其他针对该元素的专门规则。第 8 章也说明 gmd 的 XML 打包情况以及相关的命名空间,通用的打包规则在 8.6 和 8.7 中说明。

注意:编码规则的描述方式并不影响采用最好的 XML 模式生成方法。

8.2 缺省 XML 类类型编码

ISO 19118:2005 中使用类来表达 UML 中的基本模型概念,因此基本编码规则重点在于 UML 类的编码。值得一提的是 ISO 19118:2005 中,一个特性代表一个名值对,可以表示为一个属性、关联、聚合或组合[ISO 19118:2005]。类由一或多个特性组成。例如,如图 11 所示,类 Class1 有三个特性:attr1, attr2 和 role1。为了编码成 XML 模式,理解 UML 属性、关联、聚合或组合等特性之间没有区别是非常重要的。

ISO 19118:2005 也说明了 XML 模式中需要包含标识符,并规定使用标识符和全局唯一标识符来实现。在 gco 命名空间中有一个 XML 模式类型,即 gco:AbstractObject_Type (在 9.7 中详细说明)提供了所需的标识符。这里提到标识符是因为它是 XML 类型缺省编码的一部分。

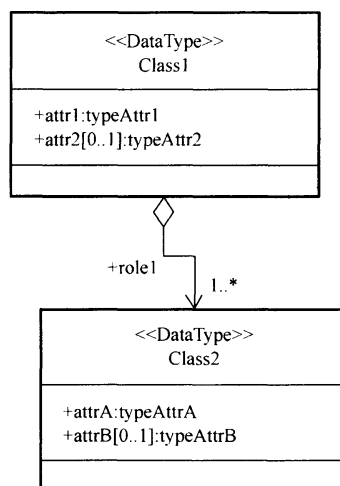


图 11 UML 示例

以下详细列出对缺省类的编码步骤:

- 1) UML 类编码为 XML 模式中的 XML 复杂类型:xs:complexType。XML 复杂类型以下称为 XML 类类型(XCT)。XML 模式中,每个 XCT 都有一个名称属性,其值为类的名称加上后缀"_Type"。

例 1: 图 11 所示的编码 Class1 类的第 1 步为:

```

<xs:complexType name = "Class1_Type">
  (...)
</xs:complexType>
  
```

- 2) 所有遵循缺省编码规则的 UML 类都包含复杂内容,可通过使用 xs:complexType 元素包含的

xs:complexContent 元素来提供这种功能。

例 2：图 11 所示的编码 Class1 类的第 2 步为：

```
<xs:complexType name = "Class1_Type">
  <xs:complexContent>
    (...)
  </xs:complexContent>
</xs:complexType>
```

3) 所有遵循缺省编码规则的 UML 类通过添加包含和 gco:AbstractObject_Type 等效的 base 属性的 xs:extension 元素来扩展 gco:AbstractObject_Type。

例 3：图 11 所示的编码 Class1 类的第 3 步为：

```
<xs:complexType name = "Class1_Type">
  <xs:complexContent>
    <xs:extension base = "gco:AbstractObject_Type">
      (...)
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

4) 所有遵循缺省编码规则的 UML 类具有一个包含该类所有特性的序列。这通过添加一个 xs:sequence 元素来实现,该 xs:sequence 元素包含对应于该类所有特性的 xs:element 元素。xs:element 元素包括以下属性：

- i. 属性的 name 等同于特性的名称。
- ii. 属性 type 等同于 XCPT 的名称,对应于特定特性指定类型对应的 UML 类。8.4 中说明了默认情况下该属性的取值为其指定 UML 类名加上 "_PropertyType",8.5.7 和 8.5.8.2 定义了例外情况。作为 type 属性值的 XCPT 名称也要加上 8.6 说明的合适的命名空间前缀。
- iii. minOccurs 和 maxOccurs 属性,其取值包含在 ISO 19118:2005 的表 A.5 中。另外,类的特性如果碰巧使用的是“set”和“sequence”结构的属性,则如果为可选属性,minOccurs 取值为 0,若为必选,其值取 1,且 maxOccurs 取值为“unbounded”。

例 4：图 11 所示的编码 Class1 类的第 4 步为：

```
<xs:complexType name = "Class1_Type">
  <xs:complexContent>
    <xs:extension base = "gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name = "attr1"
          type = "ns1:typeAttr1_PropertyType"/>
        <xs:element name = "attr2"
          type = "ns1:typeAttr2_PropertyType"
          minOccurs = "0" />
        <xs:element name = "role1"
          type = "ns1:Class2_PropertyType"
          minOccurs = "1" maxOccurs = "unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

注：本指导性技术文件的所有例子都有一个命名空间前缀“ns1”,在 UML 中不一定会表示。这是一个用来说明何时需要提供一个命名空间前缀的假想命名空间,但不对应本指导性技术文件中定义的任何命名空间或前缀。

xs:sequence 中的多个 xs:element 的次序是基于 GB/T 19710—2005 的数据字典(附录 B)中相应特性的顺序。序列中特性从其最小行号(GB/T 19710—2005 附录 B 的所有表格的第 1 列)到最高行号

的顺序列出。如果类的某个特性在 GB/T 19710—2005 附录 B 中没有,则跟在 GB/T 19710—2005 附录 B 中已有特性的后面,并符合本指导性技术文件(附录 B)中说明的该特性出现的次序。这一排序规则适用于缺省编码方法,也适用于所有 XCT 的特殊情况编码。

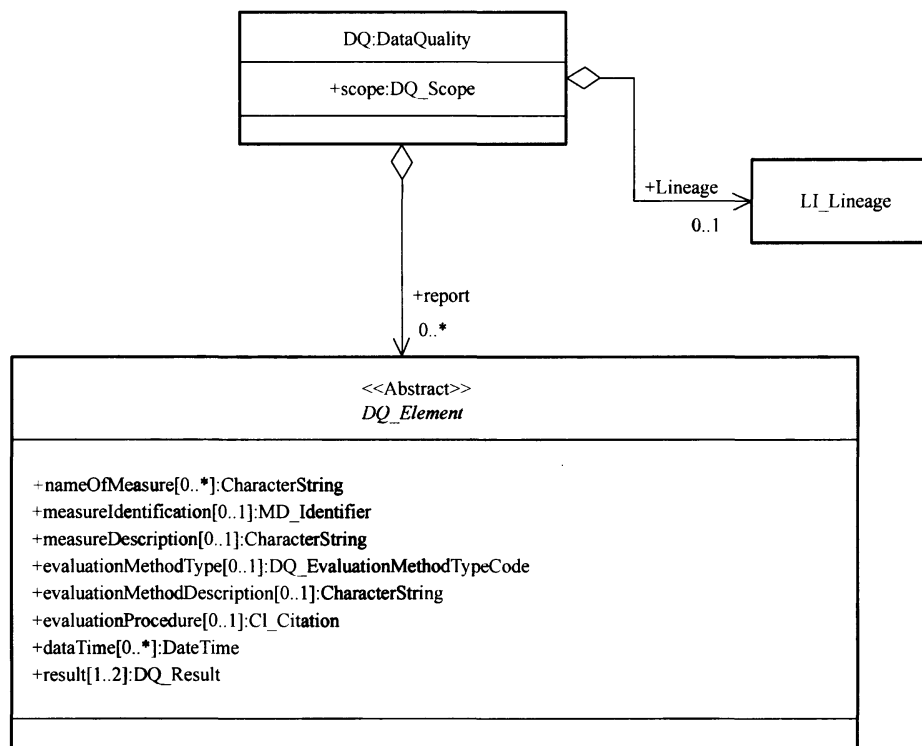


图 12 引自 GB/T 19710—2005 中的 A.2.4.1 的数据质量信息 UML 图局部

对应图 12 的 XCT:

```

<xs:complexType name="DQ_DataQuality_Type">
  <xs:complexContent>
    <xs:extension base="gco:AbstractObject_Type">
      <xs:sequence>
        <xs:element name="scope" type="gmd:DQ_Scope_PropertyType"/>
        <xs:element name="lineage"
          type="gmd:LI_Lineage_PropertyType" minOccurs="0"/>
        <xs:element name="report"
          type="gmd:DQ_Element_PropertyType" minOccurs="0"
          maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
  
```

8.3 XML 类全局元素的编码

UML 类也编码为全局元素,包含一个名称属性,等于 UML 类的名称,其 type 属性值为 8.2 说明的 XCT 的名称(包含相应的命名空间前缀)。下面全局元素称为 XML 类全局元素(XCGE)。

例 1: 类 Class1 的 XCGE 如图 12 所示:

```

<!-- ..... -->
<xs:element name="Class1" type="ns1:Class1_Type"/>
<!-- ..... -->
  
```

例 2: 图 12 所示的 DQ_DataQuality 的 XCGE 为:

```

<!-- ..... -->
<xs:element name="DQ_DataQuality" type="gmd:DQ_DataQuality_Type"/>
<!-- ..... -->
  
```

8.4 XML 类特性类型的编码

8.4.1 复杂类型 XCT 的特性类型编码

为了支持 8.2 描述的特性概念,以及给定类可以作为一组容器类的特性类型,每个 UML 类也定义为另外一个 XSD `xs:complexType`(以下称 XML 类特性类型,XCPT)。特性的包含是通过其数据类型的 XCPT 来进行管理的。默认情况下,XCPT 允许通过值和通过引用进行包含。以下列出一个默认类的 XCPT 编码的详细步骤:

1) 为了把 XCPT 和 XCT 区分开,`xs:complexType` 元素有一个等于类名加 `_PropertyType` 后缀的名称属性。

例 1: 编码图 11 所示的类 `Class1` 的 XCPT 的第 1 步为:

```
<xs:complexType name="Class1_PropertyType">
  (...)
</xs:complexType>
```

2) 为了以某种支持 6.6 说明的多态的方式来实现“通过值”的包含,XCPT 包含一个可选的 `xs:sequence` 元素,该元素包含一个带有 `ref` 属性的 `xs:element` 元素,该 `ref` 属性等于 UML 的 XCT。`xs:sequence` 元素规定为可选,是为了允许该特性以“通过引用”的方式实现。

例 2: 编码图 11 所示的类 `Class1` 的 XCPT 的第 2 步为:

```
<xs:complexType name="Class1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:Class1"/>
  </xs:sequence>
</xs:complexType>
```

3) 为了提供“通过引用”方式的包含,XCPT 有一个 `xs:attributeGroup` 元素,该元素带一个等于 `"gco:ObjectReference"` 的 `ref` 属性。在 ISO 19118:2005 中引入了 `IM_ObjectReference` 属性组,进一步说明在 9.7 中进行。

例 3: 图 11 所示的编码类 `Class1` 的 XCPT 的第 3 步为:

```
<xs:complexType name="Class1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:Class1"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
</xs:complexType>
```

4) 为了必要时说明值为 `Null` 的原因,XCPT 有一个带有 `ref` 属性的 `xs:attribute` 元素,该 `ref` 属性等于 `"gco:nilReason"`,`nilReason` 属性在 9.7.3.4 详细说明。

例 4: 图 11 所示的编码类 `Class1` 的 XCPT 的第 4 步为:

```
<xs:complexType name="Class1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:Class1"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

例 5: 图 12 所示的 `DQ_DataQuality` 类的 XCPT 实例:

```
<xs:complexType name="DQ_DataQuality_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="gmd:DQ_DataQuality"/>
  </xs:sequence>
  <xs:attributeGroup ref="gco:ObjectReference"/>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
```

注意: XCPT 中没有特定的限制来约束 UML 中定义的特性的包含信息。UML 包含说明应通过

适用于 XML 元数据的外部测试进行管理。这将在附录 A 中详细讨论。

8.4.2 简单类型 XCT 的特性类型编码

8.5.4 中给出了几个特定 XCT 编码的案例,第 9 章中说明 `xs:simpleType` 的 XCT,而不是 8.2 说明的 `xs:complexType`。如果 XCT 为简单类型,则相应的 XCPT 不包含 8.4.1 中描述的“通过引用”的功能。对简单类型不使用引用方法避免了可能的复杂性,在属性类型通过引用实现时会存在这种复杂性。

为了对构造型为枚举 (Enumeration) 的类进行编码,8.5.4 中引入了第一个简单 XCT。对应 `simpleType` 的 XCT 的 XCPT,如枚举,不包含 `gco:ObjectReference` 属性组。

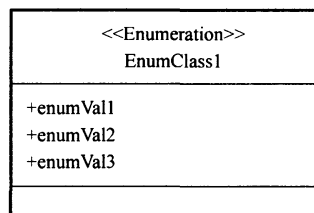


图 13 构造型为枚举类型(一种简单类型)的类的 UML 样例

例: 图 13 所示的 EnumClass1 类的 XCPT 为:

```

<xs:complexType name="EnumClass1_PropertyType">
  <xs:sequence minOccurs="0">
    <xs:element ref="ns1:EnumClass1"/>
  </xs:sequence>
  <xs:attribute ref="gco:nilReason"/>
</xs:complexType>
  
```

注意: 即使是简单类型, XCPT 中也没有专门的限制来强制某一特性使用 ByVal 或 nilReason。该包含应通过适用于 XML 元数据的外部测试来管理,这将在附录 A 中详细讨论。

8.5 特殊情况的编码

8.5.1 特殊情况编码的概述

尽管 8.2 和 8.4 描述的编码规则覆盖了地理信息国家标准中的 UML 中的多数类,但还有一些特例需要使用不同的编码规则。这些特例一般是基于各种类的构造型。所有没有构造型的类或者构造型为 Type 或 DataType 的类都遵循缺省编码规则。其他类则需遵循 8.5.2 和 8.5.6 中说明的编码规则。

8.5.2 抽象类

为了便于实现,值得更好地标识出抽象类类型,这样实现者会意识到不能出现这些类型的元素。地理信息国家标准的 UML 中的抽象类通过以下方式编码:

- 在相应 XCT 抽象类名称前加上“Abstract”;
- 在 XCT 的 `xs:complexType` 中添加 `abstract` 属性,其值设为“true”;
- 在 XCGE 的抽象类名称前加上“Abstract”;
- 在 XCGE 的 `xs:element` 中添加 `abstract` 属性,其值设为“true”;
- 在相应 XCPT 中的 `xs:element` 的 `ref` 属性中使用加了适当前缀的类名(注意在 XCPT 的名称属性前不加单词‘Abstract’,因为 XML 模式中特性类型不会是抽象的)。

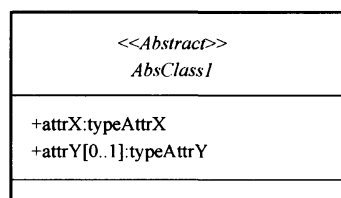


图 14 抽象类的 UML 样例

例 1: 图 14 所示的 AbsClass1 类的 XCP 为: