OGC WMTS英汉对照版

（屏幕双联显示）

OpenGIS® 基于Web的地图图块服务实现标准

OpenGIS® Web Map Tile Service Implementation Standard

翻译：周　旭

查祝华

2010-10-31

目　　录

Open Geospatial Consortium Inc.

Date: 2010-04-06

Reference number of this document: OGC 07-057r7

Version: 1.0.0

Category: OpenGIS® Implementation Standard

Editors: Joan Masó, Keith Pomakis and Núria Julià

OpenGIS® Web Map Tile Service Implementation Standard

Warning

This document is an OGC Member approved international standard. This document is available on a royalty free, non-discriminatory basis. Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation..

Document type: OpenGIS® Standard

Document subtype: OGC Standard

Document stage: Approved for release

Document language: English

Open Geospatial Consortium Inc.

日期: 2010-04-06

本文档的引用编号：OGC 07-057r7

版本: 1.0.0

分类: OpenGIS® 实现标准

编者: Joan Masó, Keith Pomakis and Núria Julià

OpenGIS® 基于Web的地图图块服务实现标准

提醒信息

本文档是一个OGC成员一致认可的国际标准。本文档在免版税和一视同仁的基础上提供使用。邀请本文档的接收者提交其知晓的任何相关专利权的通知，并提供支持性的正式文件，同时附上其意见。

文档类型: OpenGIS® Standard

文档子类型: OGC Standard

文档阶段: Approved for release

文档语言: English

i. Preface

This document defines an OGC standard for a Web Map Tile Service (WMTS) interface standard. A WMTS enabled server application can servemap tiles of spatially referenced data using tile images with predefined content, extent, and resolution.

Suggested additions, changes and comments on this standard are welcome and encouraged. Such suggestions may be submitted using the online change request form on OGC web site: http://portal.opengeospatial.org/public_ogc/change_request.php

ii. Standard verbs and usages

This document uses the standard verbs defined in subclause 5.3 of [OGC 06-121r3], which is based on the ISO/IEC Directives, Part 2. Rules for the structure and drafting of International Standards that are also included in subclause 6.1 of [OGC 06-135r7] Policy Directives for Writing and Publishing OGC Standards. In particular, the word "SHALL" (in capital letters and not "must") is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

iii. Submitting organizations

The following organizations submitted this document to the Open Geospatial Consortium Inc.

Autonomous University of Barcelona

CREAF

CubeWerx Inc.

iv. Document contributor contact points

All questions regarding this document should be directed to the editor or the contributors:

Name	Organization	E-mail Address
Joan Masó	UAB-CREAF	joan.maso@uab.cat
Keith Pomakis	CubeWerx Inc.	pomakis@cubewerx.com
Núria Julià	UAB-CREAF	n.julia@creaf.uab.cat

i. 序言

本文档定义了基于Web的地图图块服务接口标准（WMTS）这一OGC标准。支持WMTS的服务器应用系统可以使用包含预生成内容、范围和分辨率图块的方式提供具有空间参考的数据的地图图块服务。

欢迎并鼓励提出针对本标准的增补、修改建议和意见。提出的建议可以使用OGC网站 http://portal.opengeospatial.org/public_ogc/change_request.php 上的在线修改请求单提交。

ii. 标准动词与用法

本文档使用基于ISO/IEC 导则第二部分的 [OGC 06-121r3]中5.3定义的标准动词。结构和起草轨迹标准的规则也包含在[OGC 06-135r7]的6.1，即编写与出版OGC标准的政策导则。具体来说，"SHALL"（文中译为“应”）一词（全部大写且不是"must"）是用来表明要符合本标准应严格遵循的要求。

iii. 提交机构

以下机构向Open Geospatial Consortium Inc提交了本文档：

Autonomous University of Barcelona

CREAF

CubeWerx Inc.

iv. 文档编写人员的联系信息

关于本文档的所有问题应提给编者或共同起草人员：

Name	Organization	E-mail Address
Joan Masó	UAB-CREAF	joan.maso@uab.cat
Keith Pomakis	CubeWerx Inc.	pomakis@cubewerx.com
Núria Julià	UAB-CREAF	n.julia@creaf.uab.cat

v. Revision History

Date	Release	Editor	Primary Clause edited	Description
2007-05-08	07-057	Joan Masó and Núria Julià		First draft as a discussion paper in the WMS.RWG
2007-06-15	07-057r1	Joan Masó		Corrections proposed mainly by Dimitri Monie and Benjamin Chartier.
2007-08-14	07-057r2	Keith Pomakis		Merged OGC 07-057r1 and OGC 07-085r1 into a unified document.
2007-11-13	07-057r3	Keith Pomakis		Now proposes Web Map Tiling Service as a separate service rather than as a profile of WMS; introduced GetLegendGraphic and GetAlternateSources operations.
2008-03-25	07-057r4	Keith Pomakis		Several modifications proposed mainly by Chuck Morris and Joan Masó.
2008-06-15	07-057r5	Joan Masó		Reformated as a Standards document. GetLegendGraphic and GetAlternateSources operations were removed. RESTful was introduced. Major revisions by Benjamin Chartier and Keith Pomakis.
2009-01-25		Joan Masó		Many considerations in Keith Pomakis OGC 09-006 *OWS-6-DSS Engineering Report – SOAPXML and REST in WMTS* have been included.
2009-02-26	07-057r6	Joan Masó		Added WSDL annex F and general minor corrections.
2009-08-11	07-057r7	Joan Masó		CRs in RFC period have been incorporated. URL template has been introduced. Name changes from "Web Map Tiling Service" to "Web Map Tile Service". Major revision by Nuke Goldstein, Herve Caumont, Satish Sankaran, Lacey Sharpe and Xavier Pons.
2010-02-01		Adrian Custer and Joan Masó		Adrian Custer deeply detailed revision on grammar and general consistency.
2010-02-14		Carl Reed		Additional edits in preparation for publication as an OGC standard.
2010-03-09		Adrian Custer and Joan Masó		Grammar and general consistency last details. MIME type correction following 09-144r1.

vi. Changes to the OGC Abstract Specification

The OpenGIS® Abstract Specification does not require changes to accommodate the technical contents of this standard.

vii. Changes to OpenGIS® Implementation Standards

This document defines an OGC implementation standard called "OpenGIS® Web Map Tile Service Implementation Standard". This standard references the OGC Web Services Common Specification, version 1.1.0 (with Corrigendum 1). No other implementation standard should be affected.

viii. Future work

Add support for OWS Common 1.2 (approved late 2009). Several additions planned in OWS Common 1.2 [06-121r8] could potentially impact this standard and need to be evaluated.

Follow future guidance for resource oriented architectural (ROA) style that is expected to be defined in future versions of OWS Common.

v. 修改历史

日期	版本	编辑人员	修改的主要章节	说明
2007-05-08	07-057	Joan Masó and Núria Julià		在WMS.RWG 里作为讨论论文提出第一个草案
2007-06-15	07-057r1	Joan Masó		主要由 Dimitri Monie 和 Benjamin Chartier提出的修正建议
2007-08-14	07-057r2	Keith Pomakis		把 OGC 07-057r1 和 OGC 07-085r1 合并为一个文档
2007-11-13	07-057r3	Keith Pomakis		建议把基于Web的地图分块服务作为一个单独的服务而非WMS Profile。引入了GetLegendGraphic和GetAlternateSources操作
2008-03-25	07-057r4	Keith Pomakis		主要由Chuck Morris 和 Joan Masó提出的几个修改建议.
2008-06-15	07-057r5	Joan Masó		按照标准文本的要求调整了文本格式。删除了GetLegendGraphic和GetAlternateSources操作。引入了REST风格。Benjamin Chartier 和 Keith Pomakis进行的一些重要修订。
2009-01-25		Joan Masó		包含了许多Keith Pomakis OGC 09-006 *OWS-6-DSS Engineering Report – SOAPXML and REST in WMTS*中提出的注意事项
2009-02-26	07-057r6	Joan Masó		增加了WSDL附录F，以及一些一般性的小修正
2009-08-11	07-057r7	Joan Masó		包含了还在RFC阶段的一些CR；引入了URL模板；名称由“基于Web的地图分块服务”改为“基于Web的地图图块服务”; Nuke Goldstein, Herve Caumont, Satish Sankaran, Lacey Sharpe 和 Xavier Pons进行的一些重要修订。
2010-02-01		Adrian Custer and Joan Masó		Adrian Custer对语法和总体一致性进行了深入细致的修订
2010-02-14		Carl Reed		为准备作为OGC标准发布而做了一些编辑
2010-03-09		Adrian Custer and Joan Masó		语法与总体一致性最后细节修改。根据09-144r1修正了MIME类型。

vi. 对OGC抽象规范的修改

OGC抽象规范不需要修改以适应本标准的技术内容。

vii. 对OpenGIS®实现标准的修改

本文档定义了一个名为"OpenGIS® 基于Web的地图图块服务实现标准" 的OGC实现规范。本标准引用了OGC Web服务通用规范，V1.1.0（包含勘误表1）.没有其他实现标准会受到影响。

viii. 将来的工作

增加对OWS Common 1.2 (2009年通过).的支持。几个在OWS Common 1.2 [06-121r8]中规划的新增内容可能会影响本标准并需要作出评估。

遵循将来的要求，期待在OWS Common的将来版本中定义面向资源架构（ROA）风格的接口。

Foreword

The Web Map Tile Service (WMTS) described in this standard builds on earlier efforts to develop scalable, high performance services for web based distribution of cartographic maps. WMTS is inspired by the OSGeo Tile Map Service Specification (available at http://wiki.osgeo.org/index.php/Tile_Map_Service_Specification). The team that worked on this standard also considered similar initiatives, such as Google maps and NASA OnEarth. This OGC standard includes both resource (RESTful approach) and procedure oriented architectural styles (KVP and SOAP encoding) in an effort to harmonize this interface standard with the OSGeo specification.

WMTS complements earlier efforts to develop services for the web based distribution of cartographic maps. The OGC WMTS provides a complementary approach to the OGC Web Map Service (WMS) for tiling maps. **WMS** focuses on rendering **custom maps** and is an ideal solution for dynamic data or custom styled maps (combined with the OGC Style Layer Descriptor (SLD) standard). **WMTS** trades the flexibility of custom map rendering for the scalability possible by serving of **static** data (base maps) where the bounding box and scales have been constrained to discrete **tiles**. The fixed set of tiles allows for the implementation of a WMTS service using a web server that simply returns existing files. The fixed set of tiles also enables the use of standard network mechanisms for scalability such as distributed cache systems.

This standard has been structured as a stand alone standard (relying on OpenGIS Web Service Common Implementation Specification OGC 06-121r3 as a base document) but shares many concepts with the WMS 1.3.0.

This document replaces any previous versions of OGC 07-057 that were released as OGC Discussion Papers.

This document includes 7 annexes; Annexes A, B and F are normative, and Annexes C, D E and G are informative.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium Inc. shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

前言

本标准描述的WMTS是在更早的一些旨在建立可伸缩、高性能、基于Web的地图分发服务的工作成果基础上完成的。WMTS受到了OSGeo的分块地图服务规范（OSGeo Tile Map Service Specification，http://wiki.osgeo.org/index.php/Tile_Map_Service_Specification）的启发。本标准的工作团队同时还研究了其他一些类似的工作成果，如Google地图和NASA的OnEarth。本OGC标准同时包含面向资源（REST风格的方法）和面向过程（KVP和SOAP编码）的架构风格，并努力使该接口标准与OSGeo的规范保持一致。

WMTS补充了之前开发基于Web的地图分发服务的一些成果。OGC WMTS弥补了WMS不能提供分块地图的不足。WMS针对提供可定制地图的服务，是一个动态数据或用户定制地图（需结合SLD标准）的理想解决办法。WMTS牺牲了提供定制地图的灵活性，代之以通过提供静态数据（基础地图）来增强伸缩性，这些静态数据的范围框和比例尺被限定在各个图块内。这些固定的图块集使得对WMTS服务的实现可以使用一个仅简单返回已有文件的Web服务器即可，同时使得可以利用一些标准的诸如分布式缓存的网络机制实现伸缩性。

本标准虽被列为一个单独的标准（基于OGC Web服务通用实现规范），但与WMS1.3.0共用了许多概念。

本标准替代OGC发布为讨论稿的OGC 07-057的所有历史版本。

本标准包括7个附录，附录A，B和F是规范性附录，其他为资料性附录。

请注意本文档的某些部分有可能会涉及专利权。Open Geospatial Consortium Inc.不会负责指出任何或所有此类的专利权。

请本文档的接收者若知晓本文档中的标准集可能违反任何相关专利权申明或其他知识产权，提交通知，并随附意见，并请提供支持性的正式文件。

Introduction

This Web Map Tile Service (WMTS) Implementation Standard provides a standard based solution to serve digital maps using predefined image tiles. The service advertises the tiles it has available through a standardized declaration in the ServiceMetadata document common to all OGC web services. This declaration defines the tiles available in each layer (*i.e.* each type of content), in each graphical representation style, in each format, in each coordinate reference system, at each scale, and over each geographic fragment of the total covered area. The ServiceMetadata document also declares the communication protocols and encodings through which clients can interact with the server. Clients can interpret the ServiceMetadata document to request specific tiles.

The WMTS standard complements the existing Web Map Service standard of the OGC. The WMS standard focuses on flexibility in the client request enabling clients to obtain exactly the final image they want. A WMS client can request that the server creates a map by overlaying an arbitrary number of the map layers offered by the server, over an arbitrary geographic bound, with an arbitrary background color at an arbitrary scale, in any supported coordinate reference system. The client may also request that the map layers be rendered using a specific server advertised style or even use a style provided by the client when the WMS server implements the OGC Styled Layers Descriptor (SLD) standard. However, all this flexibility comes at a price: server image processing must scale with the number of connected clients and there is only limited potential to cache images between the server and client since most images are different.

As web service clients have become more powerful, it has become possible to consider an alternative strategy which forces the clients to perform image overlays themselves and which limits the clients to requesting map images which are not at exactly the right position thereby forcing the clients to mosaic the tiles obtained from the server and clip the set of tiles into a final image. This restriction of image requests to a fixed, predefined set allows for servers to scale based on communication processing abilities rather than image processing abilities because servers can prerender some or all of their images and can use image caching strategies. The fixed set of images also enables network providers to cache images between the client and the server, reducing latency and bandwidth use. Popular, non standardized, commercial implementations of this approach, such as Google Maps, Microsoft Virtual Earth and Yahoo! Maps have already shown that there are clear performance benefits to adopting this methodology.

Some WMS servers have already embarked on this road, developing their own tiling structures built by constraining WMS GetMap requests to a fixed set and then advertising those constraints in their service metadata. Although this mechanism enables those servers to scale as just described, the tiling structure and the advertising and discovery mechanisms are not standardized. That unfortunately limits interoperability and forces developers to build, for each server, special clients that can understand the server advertised constraints and limit the WMS GetMap requests issued by the client to exactly the requests understood by the particular server. This WMTS standard offers a standardized approach to declaring the images which a client can request from a server, enabling a single type of client to be developed for all servers. While developing a profile of WMS was initially considered, limiting a WMS in the ways important to allow efficient access to cacheable tiles proved awkward while forcing implementors to read both a standard and a profile seemed less efficient than developing this stand alone specification.

引言

本WMTS实现标准提供了一种采用预定义图块方法发布数字地图服务的标准化解决方案。服务使用OGC Web服务通用的服务元数据文档中标准化的申明（declaration）机制来告知其潜在用户它所提供的图块的信息。这些申明定义了每个图层（如每种类型的内容）、每种图形表示式样、每种格式、每种坐标参照系统、每种比例尺以及整个覆盖区域中每块地理区域所具有的图块。服务元数据文档同时还说明客户端可以与服务器端进行交互所使用的通信协议和编码方法。客户端可以通过解译服务元数据文档来确定如何请求特定的图块。

WMTS标准与WMS标准实现了互补。WMS标准针对客户端请求的灵活性，可以让客户端获得刚好是其期望的最终图像。WMS客户端可以要求服务器通过对服务器上所具有的地图图层进行任意数量的叠加，指定任意地理范围，设定任意的背景颜色和任意比例尺，并以任何服务器支持的坐标参照系统生成一幅地图。客户端也可以请求指定的地图图层按照服务器告知的图式进行符号化，甚至如果服务器实现了OGC SLD，可以按照客户端提供的图式进行符号化。但是，这些灵活性是有代价的，服务器的图像处理能力必须随客户端连接数的增加而增加，而且由于生成的绝大多数地图图像都不同，对服务器和客户端之间交互产生的地图图像进行缓存的可能性大受限制。

由于许多Web服务的客户端已经变得非常强大，使得采用变通策略成为可能，即要求客户端自行完成图片叠加，并限制客户端请求不在准确位置上的地图图片而导致客户端需要拼接这些图块然后剪切成最后的图片。这种把图片请求限定为对固定的、预处理的图片集的方法使得服务器能力可以随通讯处理能力而非图像处理能力递增，因为服务器可以预先生成部分或全部地图图片，并利用图片缓存策略。固定的图片集使得网络提供商可以对服务器和客户端之间传输的图片进行缓存，以减少延迟和带宽占用。这一技术方法的一些流行的非标准商业实现，如Google Maps, Microsoft Virtual Earth and Yahoo! Maps，已经表明采用这一方法可以带来明显的性能优势。

一些WMS服务器已经采用了这一方法，通过把WMS GetMap请求限定为一组固定集合，并在其服务元数据中告知这些约束规则，定义了自己的分块结构。尽管这一机制使这些服务器象描述的那样具有伸缩性，但分块结构、公告和发现机制都没有标准化。这一点限制了互操作性，并迫使开发人员为每个服务器开发专门的客户端以理解该服务器告知的约束规则并据此把该客户端发出的请求限定在特定服务器可以理解的确切范围内。本WMTS标准提供了一个申明客户端可以从服务器上获得的图片的标准化方法，使得仅需开发一种客户端软件即可访问所有服务器。同时，开始也考虑过通过采用允许对可缓存图片进行高效访问的方式对WMS进行限定来制定一个WMS的定制标准，结果证实非常不便，因为要迫使实现端同时理解标准和定制标准似乎比仅单独制定本标准效率更低。

This standard specifies WMTS in two stages. First, an abstract specification describes the semantics of the resources offered by the servers and requested by the client. This abstract definition specifies the semantics of the **ServiceMetadata** document, of the **Tile** images or representations, and of the optional **FeatureInfo** documents providing descriptions of the maps at specific locations. Second, this standard specifies several different concrete exchange mechanisms between clients and servers in two different architectural styles. The standard defines the **GetCapabilities**, **GetTile** and optional **GetFeatureInfo** operations for procedure oriented architectural style based approaches using several different message encodings, including messages encoded using Key-Value Pairs (KVP), XML messages, or XML messages embedded in SOAP envelopes. The standard also defines the request mechanisms and endpoint publishing strategy to enable a resource oriented architectural style based on web based URL endpoints allowing clients to simply request the ServiceMetadata, Tile, and FeatureInfo resources as documents.

This resource oriented architecture style is new to the OGC but offers key advantages in ease of deployment, scalability and network effects of OGC web services. The RESTful pattern provides the ability to set up conformant WMTS servers simply. If all the images are prerendered, a WMTS server could even be created using no image processing logic at all but relying only on a normal web server to return the static ServiceMetadata XML document and provide the image tile files. This is important for deployment purposes as many Internet service providers (especially the free ones) allow web pages and static content hosting but do not allow using CGI, ASP, or more advanced applications for security reasons. The RESTful approach therefore enables small organizations to provide geographic data using readily available services or simple web server configurations. This approach also scales dramatically since the issues of serving fixed resources in high volumes have been continuously tackled over the past decades. Finally, this approach can benefit from network scaling effects since the images are considered by the HTTP protocol to be standard web resources and network providers can leverage their existing technologies to improve the flow of those resources to requesting clients.

本标准对WMTS分两部分进行了规定。第一部分是抽象规范，描述关于服务器所提供资源和客户端所请求资源的语义。该抽象定义规定了ServiceMetadata文档、图块或图式表达结果，以及提供描述特定位置地图的可选的FeatureInfo文档的语义。第二部分采用两种不同架构风格定义了客户端与服务器端几个不同的具体交换机制。本标准基于面向过程的架构风格采用包括KVP、XML消息或嵌入SOAP信报的XML消息在内的不同编码方法定义了GetCapabilities, GetTile以及GetFeatureInfo（可选）操作，同时基于建立在URL端点基础上的面向资源的架构风格定义了相应的请求机制和端点发布方法，使得客户端可以把ServiceMetadata, Tile以及FeatureInfo等资源简单地作为文档进行请求。

这种面向资源的架构风格对于OGC来说是新鲜事物，但在易于部署、伸缩性以及OGC Web服务的网络效率等方面具有显著优势。REST风格让建立兼容的WMTS服务器变得简单。如果所有的图片都是预先生成的，甚至无需任何图片处理逻辑而仅靠一个常规的Web服务器返回静态的ServiceMetadata XML文档并提供图块文件即可建立一个WMTS服务器。这对于系统部署非常重要，因为许多因特网服务提供商（特别是免费的）允许网页和静态内容寄存，但出于安全原因不允许使用CGI、ASP或更高级的应用。REST方法因此让小的机构可以使用容易获得的服务或简单的Web服务器配置即可提供地理信息服务。该方法同时具有显著的伸缩性，因为通过Web提供海量的固定资源服务所存在的问题已经在过去几十年中不断设法得到解决。最后，该方法可以受惠于网络伸缩效应（network scaling effects），因为图片被HTTP协议视为标准的Web资源，网络提供商可以充分利用他们已有的技术以提高此类资源流向客户端的效率。

OpenGIS® Web Map Tile Service Implementation Standard

1 Scope

This OGC® document specifies an interface standard called "OpenGIS® Web Map Tile Service Implementation Standard" (WMTS).

This OGC® document is applicable to servers and clients that can serve and consume rendered tile maps. It can be combined with other OGC standards and also integrated with the emerging RESTful applications and "mash-ups".

2 Compliance

Compliance with this standard SHALL be checked using all the relevant tests specified in Annex A (normative).

3 Normative references

The following normative documents contain provisions that, through reference in this text, constitute provisions of this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

ISO 19105:2000, *Geographic information — Conformance and Testing*

OGC 06-121r3, *OpenGIS® Web Services Common Specification, version 1.1.0 with Corrigendum 1, Arliss Whiteside, ed.,*

NOTEThis OWS Common Specification contains a list of normative references that are also applicable to this Implementation Specification.

W3C *SOAP Version 1.2 Part 1: Messaging Framework, W3C Recommendation* 24 June 2003,

W3C *SOAP 1.2 Attachment Feature*, W3C Working Group Note 8 June 2004,

W3C *Web Services Description Language (WSDL) 1.1*, W3C Note 15 March 2001,

In addition to this document, this standard includes, as normative, several XML Schema Document files as described in Annex B.

4 Terms and definitions

For the purposes of this standard, the definitions specified in clause 4 of the OWS Common Implementation Specification [OGC 06-121r3] and SHALL apply. In addition, the following terms and definitions apply.

4.1

coordinate reference system

coordinate system that is related to the real world by a datum

4.2

coordinate system

set of mathematical rules for specifying how coordinates are to be assigned to points

OpenGIS® 基于Web的地图图块服务实现标准

1 范围

本OGC文档规定了称为WMTS的接口标准。

本OGC文档适用于提供和访问预生成的图块地图的服务器和客户端软件。它可以与其他OGC标准结合使用，也可以与新兴的REST风格的应用系统和混搭系统集成到一起。

2 一致性

宣称与本标准一致，必须通过附录A中所有相关测试。

3 规范性引用

以下引用的规范性文件所包括的条款也构成本文档的条款。对于标明日期的引用文件，其后续增补、修订版本均不适用；对于未标明日期的引用文件，则适用其最新版本。

ISO 19105:2000, *Geographic information — Conformance and Testing*

OGC 06-121r3, *OpenGIS® Web Services Common Specification, version 1.1.0 with Corrigendum 1, Arliss Whiteside, ed.,*

NOTEThis OWS Common Specification contains a list of normative references that are also applicable to this Implementation Specification.

W3C *SOAP Version 1.2 Part 1: Messaging Framework, W3C Recommendation* 24 June 2003,

W3C *SOAP 1.2 Attachment Feature*, W3C Working Group Note 8 June 2004,

W3C *Web Services Description Language (WSDL) 1.1*, W3C Note 15 March 2001,

除本文档之外，本标准还包括在规范性附录B中给出的几个XML模式文件。

4 术语与定义

根据本标准的需要，OWS的通用实现规范[OGC 06-121r3]第4节给出的定义应适用本标准；此外，以下术语和定义也适用。

4.1

坐标参照系统 coordinate reference system

与现实世界通过某一基准进行关联的坐标系统。

4.2

坐标系统 coordinate system

规定如何给点位赋予坐标的数学规则集

4.3

feature

abstraction of a real world phenomenon

4.4

feature info

information related to a particular pixel of a map that refers to the geographic data portrayed on that area

4.5

layer

basic unit of **geographic information** that may be requested as a **map** from a **server**

4.6

map

portrayal of geographic information as a digital image file suitable for display on a computer screen

4.7

portrayal

graphical presentation of information to humans

4.8

procedure oriented architectural style

platform-independent design approach that is focused on operations, their parameters and their results, that can be defined in an abstract level specification. Concrete platform-dependent specifications can be derived from the abstract level, allowing, for example, KVP or SOAP messaging.

4.9

resource oriented architectural style

platform-independent design approach that is focused on resources, representations and actions, that can be defined in an abstract level specification. Concrete platform-dependent specifications can be derived form the abstract level, allowing, for example, a RESTful architecture.

4.10

theme

a group layers that can be nested hierarchically

4.3

要素feature

对现实世界各种现象的抽象表达

4.4

要素信息 feature info

某范围内地理数据图示表达形成的地图上与某一特定像素位置相关的信息

4.5

图层 layer

可以从服务器上作为地图请求的地理信息基本单元。

4.6

地图 map

地理信息以适合计算机屏幕显示的数字图像文件形式进行图示表达的结果。

4.7

图示表达 portrayal

为便于人理解，对信息进行的图形化表达。

4.8

面向过程的架构风格 procedure oriented architectural style

一种专注操作及其参数和结果、可通过抽象规范进行定义的与平台无关的设计方法。具体的与平台相关的规范可以从该抽象规范派生，如KVP或SOAP消息机制。

4.9

面向资源的架构风格 resource oriented architectural style

一种专注资源、表现和动作、可通过抽象规范进行定义的与平台无关的设计方法。具体的与平台相关的规范可以从该抽象规范派生，如REST架构。

4.10

专题 theme

一组可以层叠嵌套的图层。

4.11

tile

a rectangular pictorial representation of geographic data, often part of a set of such elements, covering a spatially contiguous extent and sharing similar information content and graphical styling, which can be uniquely defined by a pair of indices for the column and row along with an identifier for the tile matrix.

4.12

tile matrix

a collection of tiles for a fixed scale

4.13

tile matrix set

a collection of tile matrices defined at different scales

5 Conventions

5.1 Abbreviated terms

Most of the abbreviated terms listed in subclause 5.1 of the OWS Common Implementation Specification [OGC 06-121r3] apply to this document, plus the following abbreviated terms.

ASP Active Server Pages

CGI Common Gateway Interface

JPEG Joint Photographic Experts Group (image format)

JPIP JPEG 2000 Interactive Protocol

PNG Portable Network Graphics (image format)

REST Representational State Transfer

SLD Styled Layer Descriptor

SOAP Simple Object Access Protocol

WMTS Web Map Tile Service

WSDL Web Services Description Language

5.2 UML notation

Most diagrams that appear in this standard are presented using the Unified Modeling Language (UML) static structure diagram, as described in subclause 5.2 of OWS Common [OGC 06-121r3].

4.11

图块 tile

地理数据的矩形图片表达结果。通常各个图块覆盖空间上连片的范围，共用相同的信息内容和图式符号，可以通过成对行列号并随块阵标识符一道进行唯一标识。

4.12

块阵 tile matrix

针对某一固定比例尺的图块的集合

4.13

块阵集 tile matrix set

不同比例尺下定义的块阵的集合。

5 约定

5.1 缩略语

大部分OWS通用实现规范[OGC 06-121r3]5.1节中的和以下缩略语适用于本文档。

ASP Active Server Pages

CGI Common Gateway Interface

JPEG Joint Photographic Experts Group (image format)

JPIP JPEG 2000 Interactive Protocol

PNG Portable Network Graphics (image format)

REST Representational State Transfer

SLD Styled Layer Descriptor

SOAP Simple Object Access Protocol

WMTS Web Map Tile Service

WSDL Web Services Description Language

5.2 UML表示法

根据OWS通用实现规范[OGC 06-121r3]中5.2的要求，本标准中大部分插图使用UML静态结构图表示。

5.3 Used parts of other documents

This document uses significant parts of document [OGC 06-121r3], herein referred to as "OWS Common". To reduce the need to refer to that document, this document copies some of those parts with small modifications. In tables and figures, to indicate those parts to readers of this document, the largely copied parts are shown with a light grey 15% background.

5.4 Platform-neutral and platform-specific standards

As specified in clause 10 of OGC Abstract Specification Topic 12 ―OpenGIS Service Architecture‖ (which contains ISO 19119), this document includes both Distributed Computing Platform-neutral and platform-specific standards. This document first specifies the resources in each operation request and response in platform-neutral fashion. This is done using a table for each data structure, which lists and defines the parameters and other data structures contained. These tables serve as data dictionaries for the UML model in Annex C, and thus specify the UML model data type and multiplicity of each listed item.

NOTE 1 Platform-neutral standards are contained in clause 7.

The specified platform-neutral data could be encoded in many alternative ways, each appropriate to one or more specific Distributed Computing Platform. This document currently specifies encodings appropriate for HTTP GET transfer of operations requests (using KVP or RESTful encodings), and for HTTP POST transfer of operations requests (using XML or SOAP encodings). However, the same operation requests and responses (and other data) could be encoded for other specific computing platforms, including HTTP POST transfer of raw XML requests.

NOTE 2 Platform-specific standards for KVP, SOAP and RESTful are contained in clause 8, 9 and 10 respectively.

5.5 UML graphical and table representations

The UML model data is specified herein in a series of tables, called data dictionary tables. The contents of the columns in these tables are described in Table 1.

5.3 用到的其他文档内容

本标准基于[OGC 06-121r3]（以下称OWS通用实现规范）中各个重要部分。为了减少引用此文件，本文件复制了其中的部分内容并进行了些微修改，在各个图表中，为了向读者指出这些内容，主要拷贝内容以背景灰度为15%进行表示。

5.4 平台无关与平台相关的标准

正如OGC抽象规范的第10章中的主题12- OpenGIS服务架构（含ISO19119）所规定的那样，本文档同时包括分布式计算平台的平台无关与平台相关的标准。本文档首先规定平台无关模式下每个操作请求和响应的资源。这是通过为每个数据结构提供一个表格来实现的，表格中列出并定义各个参数和其他包含的数据结构。这些表格可以作为附录C中的UML模型的数据字典，从而说明了UML模型的数据类型和各个所列参数的出现次数。

注1 平台无关标准包括在第7章

规定的平台无关的数据可以用多种选择方式编码，每种方式适合于一或多种分布式计算平台。本文档目前规定了适用于操作请求的HTTP GET传输（使用KVP或REST风格的编码）的编码，和适用于操作请求的HTTP POST传输（使用XML 或 SOAP编码）的编码。但是，相同的操作请求和响应（以及其他数据）也可以为其他特定计算平台进行编码，包括原始XML请求的HTTP POST传输方式。

注2 针对KVP, SOAP 和 REST风格的平台相关标准，分别包括在第8，9和10章。

5.5 UML图形与表格表示法

UML模型数据在一系列称为数据字典表格的表格中说明。这些表格的各列在表1中说明。

Table 1 — Contents of data dictionary tables

Column title	Column contents
Names(left column)	Two names for each included parameter or association (or data structure). The first name is the UML model attribute or association role name. The second name is the XML encoding of the parameter name. It is shown in monospaced font. Some names in the tables may appear to contain spaces, but no names contain spaces.
Definition(second column)	Specifies the definition of this parameter (omitting un-necessary words such as ―a‖, ―the‖, and ―is‖). If the parameter value is the identifier of something, not a description or definition, the definition of this parameter reads something like ―Identifier of …‖.
Data type and values (third column)	The first item is the data type used for this parameter, using data types or a data structure appropriate in a UML model, in which this parameter is a named attribute of a UML class. Alternately, the first item can identify the data structure (or class) referenced by this association, or references a separate table used to specify the contents of that class (or data structure). The second item indicates the source of values for this parameter, the alternative values, or other value information, unless the values are quite clear from other listed information.
Multiplicity and use (right or fourth column)	The first item specifies the multiplicity and optionally of this parameter in this data structure, either ―One (mandatory)‖, ―One or more (mandatory)‖, ―Zero or one (optional)‖, or ―Zero or more (optional)‖. The second item specifies how any multiplicity other than ―One (mandatory)‖ will be used.

When the data type used for this parameter, specified in the third column of such a table is an enumeration or code list, all the values specified by this document are listed, together with the meaning of each value. When this information is extensive, these values and meanings are specified in a separate table that is referenced in the third column of the table row where the parameter is defined.

NOTESeveral parameters have their data type specified in the third table column as ―Character String type, not empty‖. In the XML Schema Documents specified herein, these parameters are encoded with the xsd:string type, an XML type which does not require that these strings not be empty. Nonetheless, the injunction of the table SHALL prevail and the element SHALL not be empty

表 1 — 数据字典表的内容

列标题	列的内容
名称(第1列)	每个包含的参数或关联（或数据结构）有两个名称 第一个名称是UML模型属性或关联角色名称 第二个名称是参数名称的XML编码。显示为等宽字体。 表中某些名称可能看起来包含空格，但没有名称包含空格。
定义(第2列	说明该参数的定义（省略不必要的词，如a，the和is）。如果参数值是某种事物的标识符，而不是描述或定义，该参数的定义应形似：是…的标识符
数据类型与取值(第3列)	第一项是该参数使用的数据类型，用数据类型或适用UML模型中的数据结构说明，UML模型中，该参数是UML类的一个属性名称。 第一项也可以指明该关联指向的数据结构（或类），或指向一个用来说明该类（或数据结构）的内容的单独表格。 第2项表明该参数的取值来源，替换值或其他有用信息，除非取值非常明确来自其他列出的信息。
出现次数与用法 (第4列)	第一项说明该数据结构中该参数的出现次数和选用条件， 既1（必选）、1或多（必选）、0或1（可选）或0或多（可选）中的一种。 第二项说明使用的次数（不是1次才需说明）

当该参数使用了数据结构，表格中的第3列指定的则是一个枚举或代码表，所有本文件规定的取值都列出来，同时还有每个值的含义。如果这些信息不包含两端的值，这些值和含义则在表格中定义该参数的相应行的第3列所引用的一个单独的表格中说明.

注 有几个参数的第3列指定的数据类型都是“字符串类型，非空”。这里规定的XML模式文档中，这些参数被编码为xsd:string类型，该类型是一种不要求这些字符串为非空的一种XML类型。但是，结合表格中的定义，可以确保该元素为非空。

The contents of these data dictionary tables are normative, including any table footnotes. Particularly, the ―Multiplicity and use‖ columns in Table 6 through Table 16 in OWS Common [OGC 06-121r3], and in Table 2, 3 and Table 5 through Table 16 of this document, specify the optionality of each listed parameter and data structure in the ServiceMetadata document. Also, ―Multiplicity and use‖ columns of this document in Table 22 specify the optionality of each listed parameter and data structure in the GetTile operation request and in Table 25 specify the optionality of each listed parameter and data structure in the GetFeatureInfo operation request. All the ―mandatory‖ parameters and data structures SHALL be implemented by all WMTS clients, using a specified value(s). Similarly, all the ―mandatory‖ parameters and data structures SHALL be implemented by all WMTS servers, checking that each request parameter or data structure is received with the specified value(s). All the ―optional‖ parameters and data structures in the operation requests SHOULD be implemented by all WMTS clients using specified values, for each implemented layer to which that parameter or data structure applies. Similarly, all the ―optional‖ parameters and data structures SHALL be implemented by all WMTS servers, for each implemented layer to which that parameter or data structure is declared to apply by the server in the ServiceMetadata document.

6 WMTS overview

The goal of providing a WMTS enabled service is to be performance oriented and scalable. Therefore, servers must be able to return tiles quickly. A good way to achieve that is to use locally stored pre-rendered tiles that will not require any image manipulation or geo-processing. Server developers will decide if pre-rendered tiles will be generated in a previous tile-preparation process or generated on the fly utilizing a caching mechanism. With tile-based mapping it is important that the server will be able to handle asynchronous access to tiles as most clients will simultaneously query for multiple tiles to fill a single view.

The purpose of a WMTS service is to serve maps divided in individual tiles.

The WMTS interface allows a client to receive three types of resources either in response to a resource request in the resource oriented architectural style or in response to an operation in the procedure oriented architectural style. Those resources and operations are:

a) A ServiceMetadata resource (in response to a GetCapabilities operation for the procedure oriented architectural style) (required implementation by servers) – It describes the abilities and information holdings of the specific server implementation. In procedure oriented architectural style this operation also supports negotiation of the standard version being used for client-server interactions.

b) A **tile** resource (in response to a GetTile operation for the procedure oriented architectural style) (required implementation by servers) – It shows a fragment of a map representation of a layer.

c) A **FeatureInfo** resource (in response to a GetFeatureInfo operation for the procedure oriented architectural style) (optional implementation by servers) – It provides information about the features located at a particular pixel of a tile map, in a similar way to the WMS GetFeatureInfo operation, by providing, for example, the thematic attribute name and value pairs in textual form.

这些数据字典表格的内容是规范性的，包括所有表格脚注。特别是OWS Common [OGC 06-121r3]的表6至表16，以及本文档的表2，3和表5至表16中出现次数和用法一列规定了每个所列参数的选用条件和服务元数据文档中的数据结构。本文档中表22的出现次数和用法一列规定了GetTile操作请求中每个参数和数据结构的选用条件。表25的出现次数和用法一列规定了GetFeatureInfo操作请求中每个参数和数据结构的选用条件。对于所有必选参数和数据结构，所有WMTS客户端都应实现。相似，对于所有必选参数和数据结构，所有WMTS服务器都应实现，并检查每个请求参数或数据结构是按照指定值接受的。对于每个该参数或数据结构适用的已实现的图层，对于操作请求中所有的可选参数和数据结构，所有使用规定值的WMTS客户端都应实现。相似，对于在服务元数据文档中声明的该参数或数据结构适用的每个已实现的图层，对于所有的可选参数和数据结构，任何WMTS服务器都应实现。

6 WMTS概述

提供支持WMTS的服务是出于性能、伸缩性的需要。因此，服务器必须能够快速返回请求的图块。实现这一目的的一个好办法是使用本地存储的预先生成的图块，不需要任何图像操作或地理信息处理过程。将由服务器端开发人员决定是通过一个图块准备过程预先生成图块还是利用缓存机制实时生成图块。在这种基于图块的地图服务中，重要的是服务器要能够处理对各个图块的异步访问，因为大多数客户端将同时请求多个图块以布满一屏地图。

WMTS服务的目的是把地图以若干图块的形式提供服务。

WMTS接口可以通过响应客户端请求使其接收3类资源，包括以面向资源的架构风格的资源请求或以面向过程的架构风格的操作请求。这些资源和操作包括：

a) 一个服务元数据（ServiceMetadata）资源（面向过程架构风格下对GetCapabilities操作的响应）（服务器方必须实现）。ServiceMetadata资源描述指定服务器实现的能力和包含的信息。在面向过程的架构风格中该操作也支持客户端与服务器间的标准版本协商。

b) 图块资源（对面向过程架构风格下GetTile操作的响应）（服务器方必须实现）。图块资源表示一个图层的地图表达结果的一小块。

c) 要素信息（**FeatureInfo**）资源（对面向过程架构风格下GetFeatureInfo操作的响应）（服务器方可选择实现）。该资源提供了图块地图中某一特定像素位置处地物要素的信息，与WMS中GetFeatureInfo操作的行为相似，以文本形式通过提供比如专题属性名称及其取值的方式返回相关信息。

These operations have many similarities to other OGC Web Services (OWS), including the Web Map Service (WMS), Web Feature Service (WFS), and Web Coverage Service (WCS). Many of the aspects of this WMTS interface that are shared in common with other OWSs are specified in the OpenGIS® Web Services Common Implementation Specification [OGC 06-121r3]. Many of these common aspects are included normatively by reference to that document, instead of being repeated in this standard.

Figure 1 is a simple UML diagram summarizing the WMTS interface for the procedure oriented architectural style. This class diagram shows that the WMTS interface class inherits the getCapabilities operation from the OGCWebService interface class, and adds the getTile and getFeatureInfo operations. (This capitalization of names uses the OGC/ISO profile of UML.) A more complete UML model of the WMTS interface is provided in Annex C (informative).

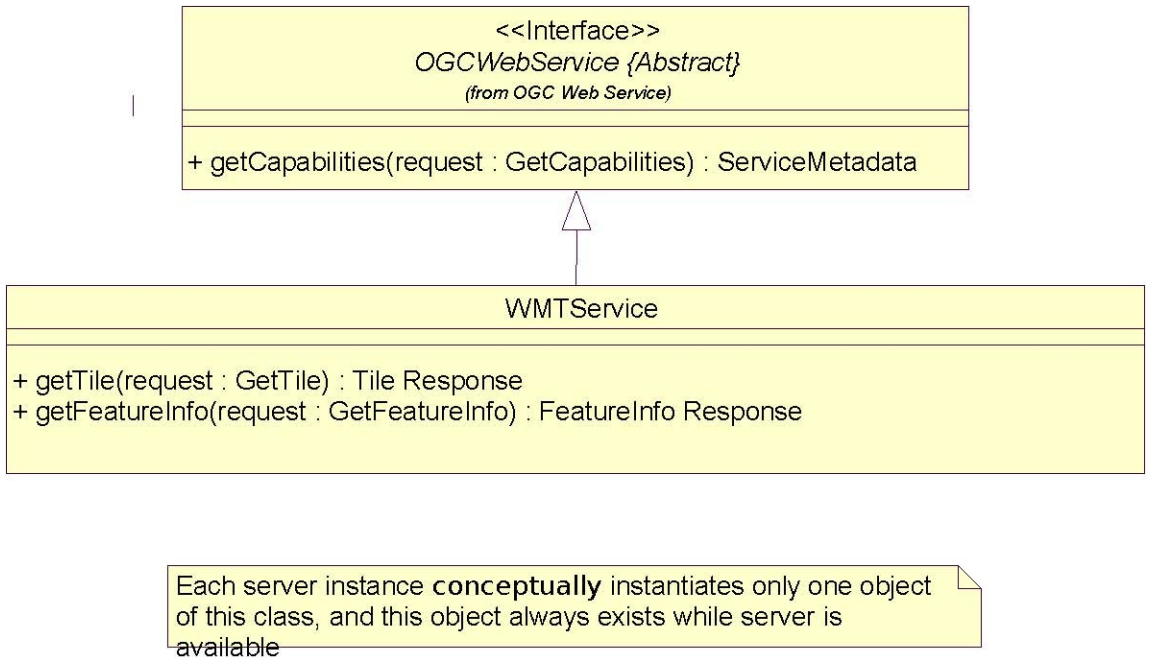

Figure 1 — WMTS interface UML diagram

NOTE In this UML diagram, the request and response for each operation is shown as a single parameter that is a data structure containing multiple lower-level parameters, which are discussed in subsequent clauses. The UML classes modeling these data structures are included in the complete UML model in Annex C.

The WMTS serves a single tile of a single layer of a map. Unlike WMS, there is no specified way to request a server to combine and return a map tile with information coming from more than one layer in a single fetching process. WMTS clients that want to show a combination of layers must make independent requests for the layer tiles and then combine or overlay the responses. Also bounding boxes and scales of these WMTS tiles are constrained to a discrete set of values.

6.1 Tile matrix set – the geometry of the tiled space

In a tiled map layer, the representation of the space is constrained in a discrete set of parameters. A tile matrix set defines these parameters. Each tile matrix set contains one or more "tile matrices" defining the tiles that are available for that coordinate reference system. Each tile matrix specifies:

这些操作与其他许多OWS相似，包括WMS、WFS以及WCS。许多WMTS接口与OWS通用实现规范中定义的其他OWS具有共同特征。许多这些通用内容都作为规范性引用包含到本标准中，这些内容不再重复。

图1是一个简单的UML图，概括了面向过程架构风格的WMTS接口。该类图表示WMTS接口类继承了OGC Web服务接口类的getCapabilities操作，并增加了getTile和getFeatureInfo操作（这里名称大写方式使用了UML的OGC/ISO专用规范）。一个更加完整的WMTS接口的UML模型在附录C（资料性附录）中给出。

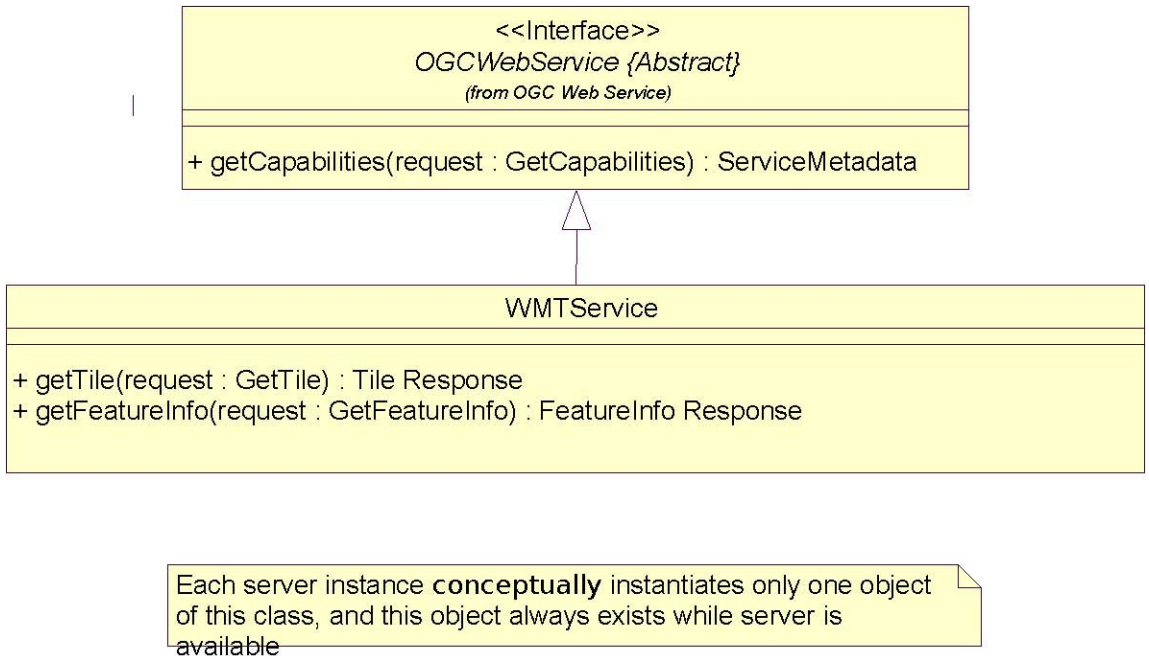

图1 — WMTS接口的UML图

注：在本UML图中，每个操作的请求与响应显示为一个参数，该参数又是一个包含多个更低层级参数的数据结构，这些数据结构将在后续章节中讨论。这些数据结构的UML类包含在在附录C中的完整UML模型中。

WMTS每次提供地图单个图层的单个图块。不像WMS，没有一个特定方法通过单个获取过程要求服务器合并并返回具有来自多个图层信息的图块。WMTS客户端要想组合多个图层，必须分别请求各个图层的图块然后组合或叠加显示返回的结果。同样这些WMTS图块的范围框和比例尺也被限定为一组离散值。

6.1 块阵集-图块空间的结构

在一个分块的地图图层中，空间的表达通过离散参数集进行限定。块阵集对这些参数进行定义。每个块阵集包含一或多个块阵，块阵定义了在相应坐标参照系统下可访问的图块：

a) The scale of the tiles as a scale denominator.

The scale denominator is defined with respect to a "standardized rendering pixel size" of 0.28 mm × 0.28 mm (millimeters). The definition is the same used in WMS 1.3.0 [OGC 06-042] and in Symbology Encoding Implementation Specification 1.1.0 [05-077r4]. Frequently, the true pixel size is unknown and 0.28 mm is a common actual size for current displays.

b) The width and height of each tile in pixels.

c) The top left (minimum x, maximum y) corner of the bounding box of the tile matrix (*i.e.*, the CRS coordinates of the top left corner of the top left pixel of the top left tile).

d) The width and height of the tile matrix in tile units (*i.e.*, number of tiles).

The number of tile matrix sets that a WMTS server serves for a particular layer is:

nTileMatrices × *nTiledStyles* × *nTiledFormats*

if no dimensions are defined or:

nTileMatrices × *nTiledStyles* × *nTiledFormats* × *nTiledDimensions*

The number of distinct tiles within each tile matrix of a tile matrix set (*i.e.*, for a particular scale within a tile-matrix set) is a product of:

matrixWidth × *matrixHeight*

Each tile matrix set defines its own set of scale levels corresponding with the contained tile matrices. Each layer references one or more tile matrix sets. Although each layer could reference a different tile matrix set, it is likely that a server will offer many layers with the same tile matrix set reference.

A tile matrix set is composed of a collection of tile matrices, each one with a resolution optimized for a particular scale and identified by a tile matrix identifier (see figure 3). Each tile matrix set has an optional approximated bounding box but each tile matrix has an exact bounding box that is deduced indirectly from other parameters. Tile matrix bounding boxes at each scale will usually vary slightly due to pixel alignment, and it is important for the client and server to take this variation into account. Given the top left point of the tile matrix in CRS coordinates (tileMatrixMinX, tileMatrixMaxY), the width and height of the tile matrix in tile units (matrixWidth, matrixHeight), the width and height of a tile in pixels (tileWidth, tileHeight), the coefficient to convert the coordinate reference system (CRS) units into meters (metersPerUnit) and the scale (1:scaleDenominator), the bottom right corner of the bounding box of a tile matrix (tileMatrixMaxX, tileMatrixMinY) can be calculated as follows:

pixelSpan = scaleDenominator × *0.28 10*$^{-3}$ */ metersPerUnit(crs);*

tileSpanX = tileWidth × *pixelSpan;*

tileSpanY = tileHeight × *pixelSpan;*

tileMatrixMaxX = tileMatrixMinX + tileSpanX × *matrixWidth;*

tileMatrixMinY = tileMatrixMaxY - tileSpanY × *matrixHeight;*

The tile space therefore looks like this:

a) 比例尺分母给定的图块比例尺

比例尺分母定义时采用的标准显示像素尺寸为0.28mm×0.28mm，与WMS1.3.0 [OGC 06-042]和符号编码实现规范1.1.0 [05-077r4]的定义相同。通常，实际的像素大小并不可知，但0.28mm对于当前的显示器是较为普遍的实际尺寸。

b) 每个图块以像素为单位的宽和高

c) 块阵范围框左上角坐标（即坐标参照系统下块阵的左上角图块的左上角像素的左上角坐标）

d) 以图块为单位的块阵宽和高(即图块数)

如果没有定义任何维度（dimension），WMTS服务器上某一特定图层所具有的块阵集的数量为：

$$nTileMatrices \times nTiledStyles \times nTiledFormats$$

否则如果定义了维度（dimension），块阵集的数量为：

$$nTileMatrices \times nTiledStyles \times nTiledFormats \times nTiledDimensions$$

块阵集中每个块阵中图块的数量为（如对于块阵集中某一特定比例尺）：

$$matrixWidth \times matrixHeight$$

每个块阵集规定自己的比例尺系列，分别对应所包含的块阵。每个图层涉及一或多个块阵集。虽然每个图层可以引用不同的块阵集，服务器很可能会让多个图层引用相同的块阵集。

块阵集由一组块阵组成，每个块阵具有一个由块阵标识符标识并根据特定比例尺优化过的分辨率（如图3）。每个块阵集有一个可选的大致的范围框，但每个块阵都有一个间接通过其他参数推算而来的准确的范围框。块阵各级比例尺所对应的范围框由于像素对齐的需要通常会有轻微变化，客户端和服务器必须把这些变化考虑在内。给定块阵左上角点的坐标参照系下的坐标（ileMatrixMinX, tileMatrixMaxY）、块阵的以图块为单位的宽和高(matrixWidth, matrixHeight)、图块的以像素为单位的宽和高(tileWidth, tileHeight)、坐标参照系下长度单位与米之间的转换因子（metersPerUnit）以及比例尺（1:scaleDenominator），则块阵范围框右下角点的坐标(tileMatrixMaxX, tileMatrixMinY)可以由下面的公式计算得出：

$$pixelSpan = scaleDenominator \times 0.28\ 10^{-3} / metersPerUnit(crs);$$

$$tileSpanX = tileWidth \times pixelSpan;$$

$$tileSpanY = tileHeight \times pixelSpan;$$

$$tileMatrixMaxX = tileMatrixMinX + tileSpanX \times matrixWidth;$$

$$tileMatrixMinY = tileMatrixMaxY - tileSpanY \times matrixHeight;$$

由此得到的图块空间如下图所示：

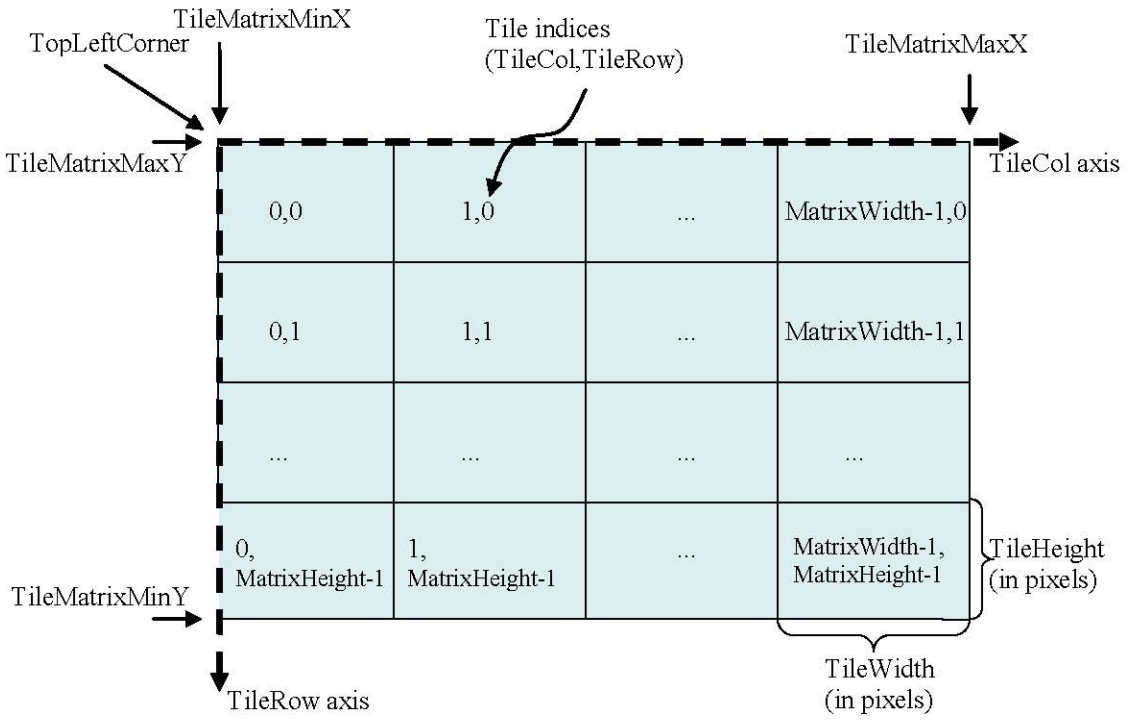

Figure 2 — Tile Space

Each tile in a tile matrix is identified by its TileCol and TileRow indices that have their 0,0 origin in the tile next to the top left corner of the tile matrix and that increases towards the right and towards the bottom respectively, as shown in figure 2. Annex H in this document includes pseudo code that illustrates the process for obtaining the tile indices that cover a bounding box rectangle and also the computation to get the CRS coordinates that bound a tile.

NOTE 1 Non-square pixels are not supported. This is different from WMS, which does allow non-square pixels (although many implementations fail to support this properly).

A tiled layer links to its tile matrix set through a tileMatrixSet URI that points to a TileMatrixSet section that completely defines it as previously explained. A layer can use a specific TileMatrixSet that describes a region adjusted to the actual content of this layer. In this case, the optional tileMatrixSetLimits section will not be used and changes in spatial extension of the layer can affect the minimum bounding box of the layer forcing to redefine the TopLeftCorner of each TileMatrix and that will end up changing the TileCol,TileRow indices thereby invalidating any previously cached tile. To overcome this problem a layer can optionally use a more generic TileMatrixSet that covers a bigger (or even global) area. In fact, that TileMatrixSet will define an area that could be covered by the layer in a future and could easily be shared for many layers in this server. To inform the client about the valid range of the TileCol and Tile Row indices a layer definition can optionally use the tileMatrixSetLimits section that specifies a minimum and a maximum that are limits of these indices for each TileMatrix. Any request outside these limits will result in a server exception (see Figure 6).

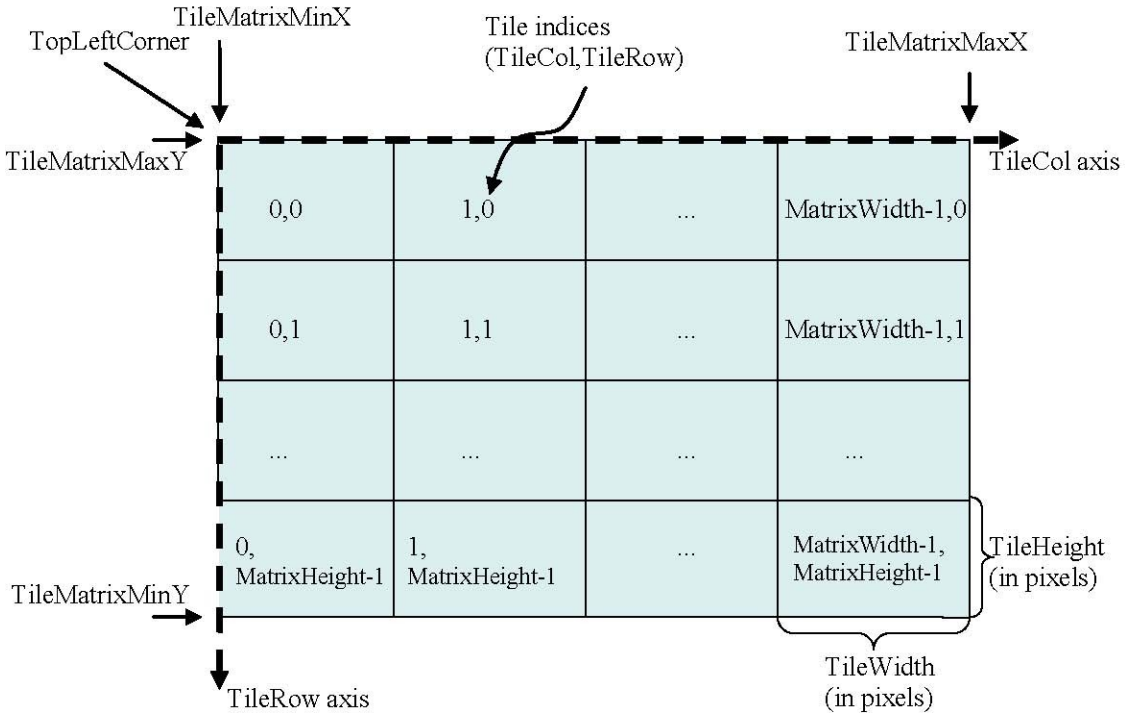

图2 — 图块空间

块阵中每个图块由其列和行的值(TileCol 、TileRow)标识。紧挨块阵左上角的图块的列和行的值为（0，0），分别向右向下递增，如图2所示。本文件的附录H包括了说明获取覆盖某一范围框矩形的行列号的计算过程和某一图块范围框的坐标范围计算方法的伪码。

注1 不支持非正方的像素。这一点不同于WMS，WMS确实支持非正方的像素（但许多实现并不能正确支持）。

分块后的图层通过tileMatrixSet URI与其块阵链接，tileMatrixSet指向一个TileMatrixSet子集，该子集如前面说明的那样对块阵集给出了完整的定义。一个图层可以使用一个适合该图层实际内容对应区域的特定TileMatrixSet。这种情况下，不会用到可选的tileMatrixSetLimits子集，图层空间范围的变化可能会影响图层的最小范围框，从而迫使重新定义每个TileMatrix的TopLeftCorner，以致改变TileCol,TileRow的值，并由此导致之前缓存图块失效。要避免这一问题，图层可以使用比该范围稍大（甚至覆盖全球）的更为通用的TileMatrixSet。实际上，TileMatrixSet可以定义图层将来可能覆盖的范围并可以很容易与该服务器上的许多图层共用。为了告知客户端TileCol和TileRow的有效取值范围，图层的定义可以选用tileMatrixSetLimits部分说明每个TileMatrix的行列最大最小值。任何超出范围的请求都会得到服务器端的一个出错响应（见图6）

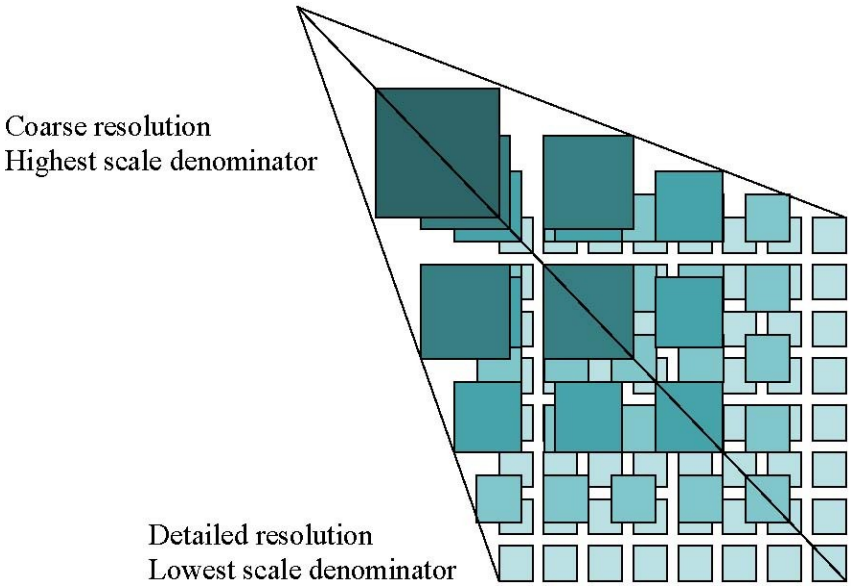

Figure 3 — Tile Matrix Set representation

In some other standards, this way of dividing the space is called *image pyramid* like in clause 11.6 of the KML 2.2 [OGC 07-147r2]. JPEG2000 (ISO/IEC 15444-1) and JPIP (ISO/IEC 15444-9) also use a similar division of the space called *resolution levels*. Nevertheless, in those cases the pyramid is self defined starting from the more detailed tile matrix that uses square tiles, and constructing tiles of the next scales by successively aggregating 4 tiles of the previous scale and so on. That approach involves a more rigid structure which has scales related by powers of two and tiles that perfectly overlap tiles on the inferior scale denominators. Since WMTS is more flexible, KML *superoverlays* or JPEG2000 based implementations can still use WMTS to describe their tile matrix sets and to serve tiles. Annex E.3 and E.4 describe scale sets related by powers of two.

Each of the WMTS procedure oriented architectural style operations and resource oriented architectural style resources are described in more detail in subsequent clauses.

NOTE 2Clients and servers have to be careful when comparing floating numbers with tolerance (double precision, 16 digit numbers, has to be used).

6.2 Well-known scale sets

Since a WMTS server will serve its data in a limited number of coordinate systems and scales (because, unlike a WMS, it serves only pre-defined tiles), and since some simple WMTS client will be unable to perform coordinate-system transformations or rescaling of tiles, the ability for a WMTS client to overlay tiles from one server on top of tiles from other servers will be limited unless there are some general agreements among WMTS servers as to what constitutes a common coordinate reference system and a common set of scales. Thus, this standard defines the concept of well-known scale sets. In order to increase interoperability between clients and servers it is recommended that many layers use a common set of scales in the same CRS that the target community agree to use.

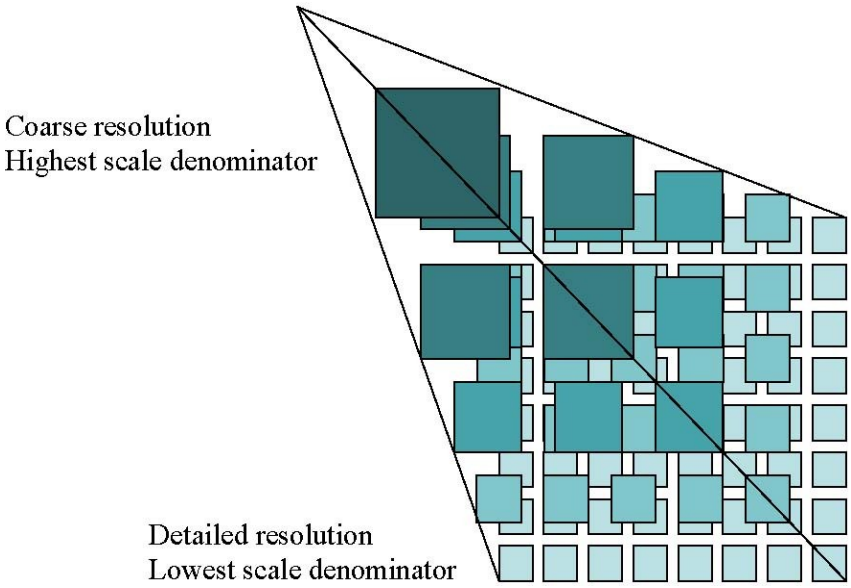

图3 — 块阵集示意

在某些其他标准中，这种划分空间的方法称为图像金字塔，如 KML 2.2 [OGC 07-147r2]的11.6 中。JPEG2000 (ISO/IEC 15444-1) 和 JPIP (ISO/IEC 15444-9)也使用了相似的空间划分方法，称为分辨率级别。但是，这些方法中的金字塔都是自定义的，始于使用方块的最详尽的块阵，之后通过聚合前一级四个图块依次构建下一级比例尺的图块。该方法形成了一个更为严格的结构，各级比例尺形成倍增系列，各级图块精确覆盖下一级比例尺的图块。由于WMTS更为灵活，KML应用程序或基于JPEG2000的实现依然可以利用WMTS来描述他们的块阵集并提供图块服务。附录E.3和E.4给出了一些有关倍乘关系的比例尺系列。

每个面向过程架构风格的WMTS操作和面向资源架构风格的WMTS资源将在后面的章节中详细描述。

注2 客户端和服务器端在比较具有误差的浮点数时必须留意（必须使用双精度16位数）。

6.2 知名比例尺系列

由于WMTS只能以有限的坐标系统和比例尺提供服务（因为只提供预生成的图块，不如WMS那样灵活），而且一些WMTS客户端不能够进行坐标系统变换或图块缩放，WMTS客户端把来自一个服务器的图块叠加到来自另一个服务器的图块之上的能力是有限的，除非在WMTS服务器之间有共同的协议约定一个共同的坐标参照系统和一个共同的比例尺系列。因此，本标准定义了知名比例尺系列的概念。为了增加客户端与服务器端的互操作性，建议多个图层使用在目标应用领域约定使用的通用比例尺系列和相同的坐标参照系统。

A well-known scale set is a well-known combination of a coordinate reference system and a set of scales that a tile matrix set declares support for. Each tile matrix set references one well-known scale set. A client application can confirm that tiles from one WMTS server are compatible with tiles from another WMTS server merely by verifying that they declare a common well-known scale set. It may also be the case that a client application is limited to supporting a particular coordinate system and set of scales (*e.g.*, an application that overlays WMTS tiles on top of Google Maps tiles). In this situation, a client application can accept or reject a WMTS as being compatible merely by verifying the declared well-known scale set. Furthermore, the existence of well-known scale sets provides incentive for WMTS servers to support a well-known scale set, increasing the odds of compatibility with other WMTS sources. The informative Annex E provides several well-known scale sets and others could be incorporated in the future.

A tile matrix set conforms to a particular well-known scale set when it uses the same CRS and defines all scale denominators ranging from the largest scale denominator in the well-known scale set to some low scale denominator (in other words, it is not necessary to define all the lower scale denominators to conform to a well-known scale set).

NOTE 1 Well-known scale sets are technically not necessary for interoperability, since a client application can always examine the actual list of coordinate systems and scales available for each layer of a WMTS server in order to determine its level of compatibility with other WMTS servers. Well-known scale sets are merely a convenience mechanism.

7 WMTS Implementation model

This clause describes the WMTS resources that can be requested by a client from a server in either the procedure oriented architectural style or in the resource oriented architectural style. It also describes the procedure oriented architectural style operations in an abstract way; for KVP encoding, see clause 8 and for SOAP encoding, see clause 9. Resource oriented architectural style description and a RESTful implementation can be found in clause 10.

7.1 Service metadata

This subclause describes the ServiceMetadata document and the way in which it may be obtained using either a procedure oriented architectural style or a resource oriented architectural style.

7.1.1 ServiceMetadata document

The ServiceMetadata document is the response document of a GetCapabilities request in procedure oriented architectural style or of a standard request to the right endpoint in a resource oriented architectural style. It is the entry point resource that represents the resources available on the service and communication requirements for the service.

7.1.1.1 ServiceMetadata document description

The ServiceMetadata document contains all the sections specified in Table 2, but partial documents can be requested containing only a subset of these sections. Depending on the values in the Sections parameter of the GetCapabilities operation request in the procedure oriented architectural style (see subclause 7.1.2.1), any combination of these sections can be requested and SHALL be returned when requested except if the service does not support requests for sub-sections of the ServiceMetadata document.

知名比例尺系列是一个坐标参照系统和若干比例尺集合构成的广为所知的组合，也是块阵集申明支持的对象。每个块阵集描述一个知名比例尺系列。客户端应用程序可只需通过检查各个WMTS服务器是否申明使用的同一个知名比例尺系列即可确认来自它们的图块是否兼容。也可能客户端应用程序只支持限定的特定坐标系统以及比例尺集合（如把WMTS图块与Google Maps图块进行叠加的应用程序）。这种情况下，客户端应用程序可以检查申明的知名比例尺系列来确定是否兼容，并决定接受或拒绝WMTS。知名比例尺系列的存在可以鼓励WMTS服务器对知名比例尺系列的支持，增加了与其他WMTS服务兼容的可能性。资料性附录E提供了几个知名的和其他将来可能加入进来的比例尺系列。

当使用相同的坐标参照系统时，块阵集与特定知名比例尺系列相关联，块阵集定义了知名比例尺系列中从最大比例尺分母到某个较小比例尺分母的所有比例尺分母（换句话说，如果与知名比例尺系列保持一致，就不必再规定所有更低比例尺分母）。

注1 技术上来说，针对互操作并不需要知名比例尺系列，因为客户端程序可以始终检验WMTS服务器的每个图层支持的坐标系统和比例尺，以确定与其他WMTS服务器的兼容程度。知名比例尺系列仅是一个便利机制。

7 WMTS实现模型

本章描述客户端可以从服务器请求到的WMTS资源，请求可以是面向过程的架构风格也可以是面向资源的架构风格。本章也对面向过程架构风格的操作进行了抽象描述；对于KVP编码，见第8章，对于SOAP编码的，见第9章。面向资源的架构风格的描述以及REST方式的实现见第10章。

7.1 服务元数据

本节对ServiceMetadata文档以及获取该文档的方式进行了说明，获取方式包括面向过程的架构风格和面向资源的架构风格。

7.1.1 服务元数据文档

ServiceMetadata文档是面向过程架构风格下GetCapabilities请求的响应文档，也是面向资源架构风格下对正确端点发起标准请求的响应文档。它是入口资源，描述了服务中可用的资源以及连接该服务的必备条件。

7.1.1.1服务元数据文档说明

ServiceMetadata文档包括表2中所有的子集，但是可以请求只包括部分这些子集的文档。视GetCapabilities操作中Sections参数的取值（见7.1.2.1），可以请求这些子集的任意组合并返回相应的结果，除非服务不支持对ServiceMetadata文档子集内容的请求。

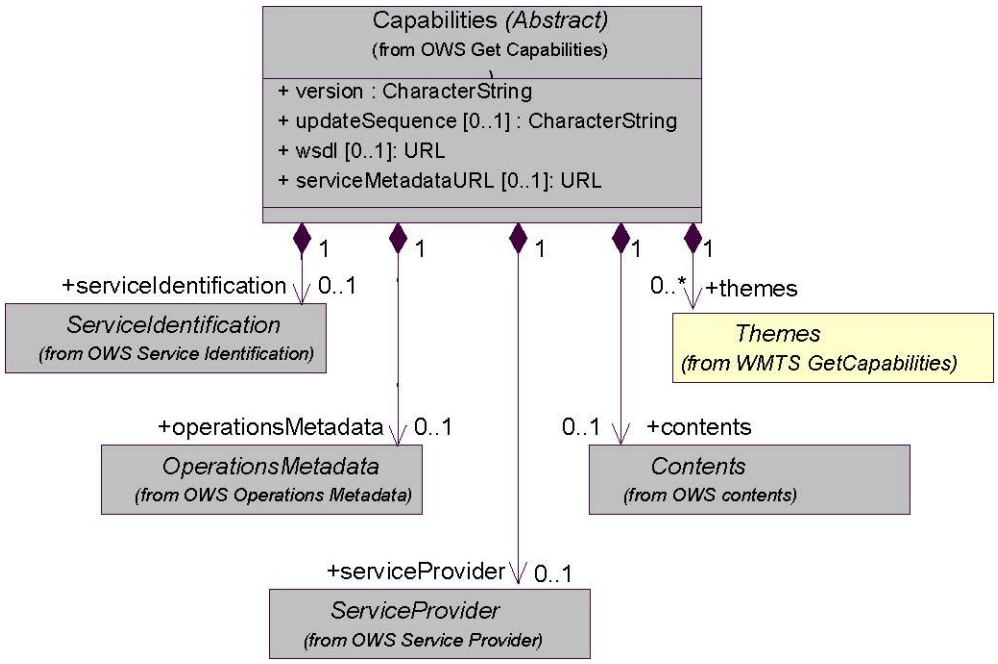

Figure 4 — ServiceMetadata UML model

Table 2 —ServiceMetadata sections

Section name	Contents
ServiceIdentification	Metadata about this specific server. The schema of this section SHALL be the same as for all OWSs, as specified in subclause 7.4.3 and owsServiceIdentification.xsd of OWS Common [OGC 06-121r3].
ServiceProvider	Metadata about the organization operating this server. The schema of this section SHALL be the same for all OWSs, as specified in subclause 7.4.4 and owsServiceProvider.xsd of OWS Common [OGC 06-121r3].
OperationsMetadata	Metadata about the operations specified by this service in procedure oriented architectural style and implemented by this server, including the URLs for operation requests. The basic contents and organization of this section is the same as for all OWSs, as specified in subclause 7.4.5 and owsOperationsMetadata.xsd of OWS Common [OGC 06-121r3].
Contents	Metadata about the data served by this server. For the WMTS, this section SHALL contain data about layers and TileMatrixSets, as specified in Tables 5 through 14 below.
Themes	Metadata describing a theme hierarchy for the layers, as specified in Tables 15 and 16 below.

The ServiceIdentification and ServiceProvider sections are described on subclause 7.4.4 and 7.4.5 of OWS Common [OGC 06-121r3]. The OperationsMetadata, Contents and Themes sections are described in subclauses 7.1.2.1, 7.1.2.2 and 7.1.2.3 of this document.

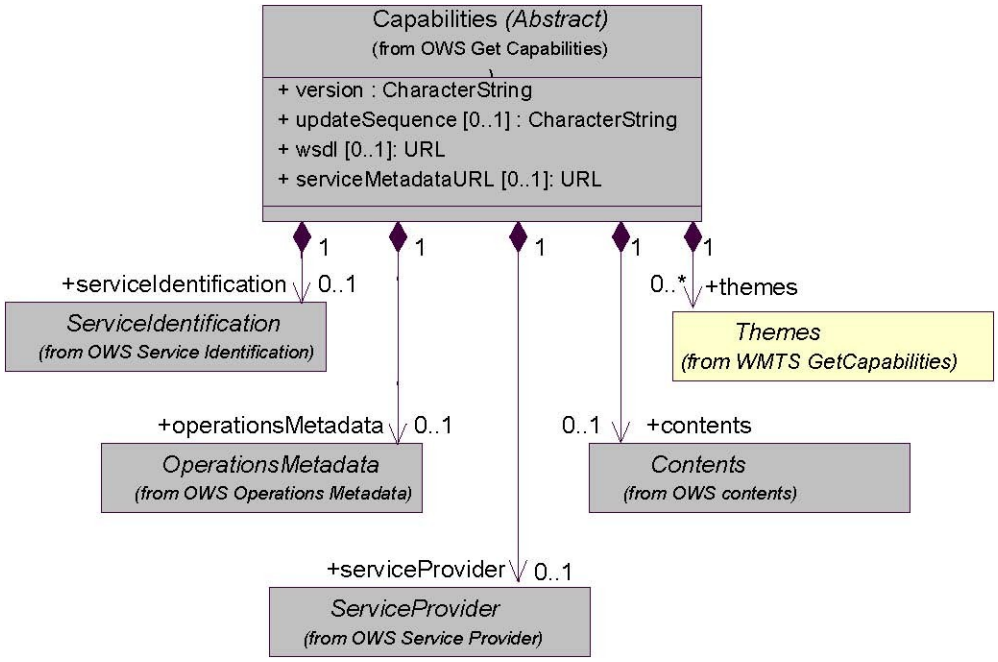

图4 — ServiceMetadata的UML模型

表2 —ServiceMetadata子集

子集名称	内容
ServiceIdentification	关于指定服务器的元数据。该子集的模式与其他所有OWS的相同，在OWS通用实现规范[OGC 06-121r3]的7.4.3和owsServiceIdentification.xsd中进行了定义。
ServiceProvider	说明运作该服务器的机构的元数据。该子集的模式与其他所有OWS的相同，在OWS通用实现规范[OGC 06-121r3]的7.4.4和owsServiceProvider.xsd中进行了定义。
OperationsMetadata	关于该服务器实现的面向过程架构风格下服务所支持操作以及操作请求所用的URL的元数据。该子集的基本内容和组织方式与所有OWS相同，在OWS通用实现规范[OGC 06-121r3]的7.4.5和owsOperationsMetadata.xsd中进行了定义。
Contents	关于该服务器提供服务的数据的元数据。对于WMTS，该子集应包含关于图层和块阵集的信息，在后面的表5至表14进行说明。
Themes	描述各图层的专题分级体系的元数据，在下面表15和表16说明。

ServiceIdentification和ServiceProvider两个子集在OWS通用实现规范[OGC 06-121r3]的7.4.4和7.4.5中说明，OperationsMetadata, Contents 和Themes三个子集在本文件中的7.1.2.1, 7.1.2.2 和7.1.2.3中说明。

In addition to these sections, each service metadata document SHALL include the mandatory "version" parameter and can optionally include "updateSequence" parameter specified in Table 9 in subclause 7.4.2 of OWS Common [OGC 06-121r3] and copied below. Finally, "WSDL" and "serviceMetadataURL" parameters are only needed for servers using specific encodings.

Table 3 — Parts of the Capabilities data structure

Names	Definition	Data type and values	Multiplicity and use
version `version`	Standard version for operation, in this case for GetCapabilities operation response	Character String type, not empty. Value is list of x.y.z ―version‖ values. SHALL be "1.0.0"	One (mandatory)
updateSequence `updateSequence`	Service metadata document version, value is ―increased‖ whenever any change is made in complete service metadata document	Character String type, not empty Values are selected by each server, and are always opaque to clients	Zero or one (optional) Omitted when parameter not supported by server
WSDL `WSDL`	Reference to a WSDL resource	URL type	Zero or more (optional) Only for SOAP encoding
serviceMetadataURL `ServiceMetadataURL`	Reference to a ServiceMetadata resource	URL type	Zero or more (optional) Mandatory in resource oriented architectural style
serviceIdentification `ServiceIdentification`	Metadata about this specific server	ServiceIdentification section, see Table 11 of OWS Common [OGC 06-121r3]	Zero or one (optional)
serviceProvider `ServiceProvider`	Metadata about the organization operating this server.	ServiceProvider section, see Table 12 of OWS Common [OGC 06-121r3]	Zero or one (optional)
operationsMetadata `OperationsMetadata`	Metadata about the operations specified by this service	OperationsMetadata section, see Table 13 of OWS Common [OGC 06-121r3]	Zero or one (optional)
contents `Contents`	Metadata about the data served by this server.	Contents section, see Table 5	Zero or one (optional)
themes `Themes`	Metadata describing a theme hierarchy for the layers	Themes section, see Table 15	Zero or more (optional)

Parameters "version", and "updateSequence" are described in subclause 7.4.2 of OWS Common [OGC 06-121r3]. Parameters "WSDL" and "serviceMetadataURL" are described in Annex F2 and subclause 010.1.1 of this document.

7.1.1.1.1 *OperationsMetadata* section contents

The OperationsMetadata section is the same as for all OGC Web Services, as specified in subclause 7.4.6 and owsOperationsMetadata.xsd of OWS Common [OGC 06-121r3]. It is only relevant in the procedure oriented architectural style. The parameters are specified in Table 4 bellow. In Table 4, the ―Name‖ column uses dot-separator notation to identify parts of a parent item. The ―Value‖ column references an operation parameter, in this case an operation name, and the meaning of including that value is listed in the right column.

除这些子集之外，每个元数据文档应包含一个必选的version参数，可以包含一个在OWS通用实现规范[OGC 06-121r3]的表9和7.4.2说明、下面也做了拷贝的updateSequence参数。最后，WSDL和serviceMetadataURL参数只是在使用特定编码时才需要。

表3 —Capabilities数据结构的组成

名称	定义	数据类型与取值	出现次数与用法
version version	GetCapabilities操作响应的标准版本	字符串，非空 取值为x.y.z的值列表 如1.0.0	1(必选)
updateSequence updateSequence	服务元数据文档的版本，一旦对服务的元数据文档进行修改，其值就应增加	字符串，非空 由每个服务器确定，对客户端视不透明的	0或1（可选） 如果服务器不支持则忽略该参数
WSDL WSDL	对WSDL资源的引用	URL类型	0或1（可选） 仅适用于SOAP编码
serviceMetadataURL ServiceMetadataURL	对ServiceMetadata资源的引用	URL类型	0或1（可选） 在面向资源架构风格下是必选的
serviceIdentification ServiceIdentification	关于该指定服务器的元数据	ServiceIdentification子集，见OWS通用实现规范[OGC 06-121r3]的表11	0或1（可选）
serviceProvider ServiceProvider	关于运作该服务器的机构的元数据	ServiceProvider子集，见OWS通用实现规范[OGC 06-121r3]的表12	0或1（可选）
operationsMetadata OperationsMetadata	关于该服务提供的操作的元数据	OperationsMetadata子集，见OWS通用实现规范[OGC 06-121r3]的表13	0 或 1（可选）
contents Contents	关于该服务器提供服务的数据的元数据	Contents子集，见表 5	0 或 1（可选）
themes Themes	关于图层专题层级的元数据	Themes 子集，见表15	0 或 1（可选）

version和updateSequence参数在OWS通用实现规范[OGC 06-121r3]的7.4.2中进行了说明。参数WSDL和serviceMetadataURL在本文件的附录F2和10.1.1中进行了说明。

7.1.1.1.1 *OperationsMetadata*子集的内容

OperationsMetadata子集与其他所有OGC Web服务的一样，在OWS通用实现规范[OGC 06-121r3]的7.4.6和owsOperationsMetadata.xsd进行了详细说明，只与面向过程架构风格相关。相关参数在下面表4中说明。表4中，名称一列使用点号“.”分开的符号来标识其上级项。取值一列引用一个操作参数，此处为操作的名称，以及取该值时的含义。

Table 4 — Values of OperationsMetadata section parameters

Name	Value	Meaning of parameter value
Operation.name	GetCapabilities	The GetCapabilities operation is implemented by this server.
	GetTile	The GetTile operation is implemented by this server.
	GetFeatureInfo	The GetFeatureInfo operation is implemented by this server.

In addition to the values listed in Table 4, there are many optional values of the ―Name‖ attributes and ―Value‖ parameters in the OperationsMetadata section, which MAY be included when considered useful. Most of these attributes and parameters are for recording the domains of various operation parameters and quantities.

The Operation data type allows specifying distributed computing platform (DCP) parameters and the encoding of this DCP as a Constraint within the DCP parameter.

All WMTS servers operating in a procedure oriented architecture style and using HTTP SHALL specify with an ows:Constraint parameter the encodings that MAY be sent using HTTP GET or HTTP POST. Each operation can support more than one encoding and the set of supported encodings CAN be different for each operation (but this is discouraged since it is not usually expected).

All WMTS servers operating in a procedure oriented architecture style and using HTTP SHALL specify the message encodings that MAY be sent using HTTP GET transfer of operation requests. Specifically, an ows:Constraint parameter SHOULD be included, with ―GetEncoding‖ as the value of the ―name‖ attribute and specifying the values allowed:

a) The value ―KVP‖ indicates that KVP encoding is allowed, when using HTTP GET transfer as specified in clause 8.

Also, all WMTS servers operating in a procedure oriented architecture style and using HTTP SHALL specify the message encodings that MAY be sent using HTTP POST transfer of operation requests. Specifically, an ows:Constraint parameter SHALL be included, with ―PostEncoding‖ as the value of the ―name‖ attribute and specifying the values allowed:

a) The value ―SOAP‖ shall indicate that SOAP encoding is allowed, as specified in clause 9.

b) The value ―XML‖ shall indicate that XML encoding is allowed (without SOAP message encapsulation).

c) The value ―KVP‖ shall indicate that KVP encoding is allowed, when using HTTP POST transfer.

If the HTTP connection point URL is different for different encodings of the operation requests, the URL SHALL be specified in an ows:Constraint parameter in each Get or Post section. If the connection point URL is the same for all encodings of all operation requests, this ows:Constraint parameter SHALL be included in the OperationsMetadata section. The constraint names and values presented in this subclause are the actual exact names and values that SHALL be used for each encoding explained and are not just examples.

表4 —OperationsMetadata 子集中个参数的取值

名称	取值	参数值的含义
Operation.name	GetCapabilities	该服务器实现了GetCapabilities操作
	GetTile	该服务器实现了GetTile操作
	GetFeatureInfo	该服务器实现了GetFeatureInfo操作

除表4列出的取值，OperationsMetadata子集中还有许多可选名称属性和取值参数，可视情况包含进来。大部分这些属性和参数记录了各个操作参数和数量的值域范围。

操作的数据类型允许指定分布式计算平台（DCP）参数，以及DCP参数内Constraint指定的该DCP的编码。

所有采用面向过程架构风格和使用HTTP的WMTS服务器应采用一个ows:Constraint参数指出可以使用HTTP Get和POST方法发送的编码方法。每个操作可支持多于1个编码方法，每个操作支持的编码方法集可以互不相同（但通常不期望这样，因此并不鼓励这样）。

所有采用面向过程架构风格和使用HTTP的WMTS服务器应指明使用HTTP Get传输操作请求方式中可以发送的消息编码方式。明确地说，应包含一个名称（name）属性取值为GetEncoding的ows:Constraint参数，以说明允许的取值：

a) 取值为KVP，表示当使用HTTP GET传输方法（见第8章）时，允许使用KVP编码；

同时，所有面向过程架构风格和使用HTTP的WMTS服务器应指明使用HTTP POST方式传输操作请求时使用的消息编码方法。明确地说，应包含一个名称（name）属性取值为PostEncoding的ows:Constraint参数，以说明允许的取值：

a) 取值为SOAP，表示允许使用SOAP编码方法，在第9章说明；

b) 取值为XML，表示允许使用XML编码方法（没有SOAP消息封包）；

c) 取值为KVP，表示使用HTTP POST传输时，允许使用KVP编码方法 。

如果HTTP连接点的URL会因操作请求的编码方式不同而不一样，则应在GET或POST子集的ows:Constraint参数中指明其URL。如果连接点URL对于操作请求的所有编码形式都一样，该ows:Constraint参数应包含在OperationsMetadata子集中。本节给出的Constraint的名称与取值，在实际使用时应严格与此保持一致，在此列出并不是仅作为举例。

Resource oriented architecture style HTTP encodings SHALL not be described in the OperationsMetadata section. Instead, the service metadata document provided by servers operating in a resource oriented architectural style SHALL use ResourceURL and ServiceMetadataURL to indicate support for that architectural style, as is explained in clause 10.

7.1.1.1.2 *Contents* section contents *Contents* Section

The Contents section of a ServiceMetadata document contains metadata about the data served by this server. For the WMTS, this Contents section contains a general description of the layers available and descriptions of the extra dimensions, styles, image formats and tile matrix sets that apply to each layer. The Contents section SHALL include parameters as specified in Table 5 through Table 14.

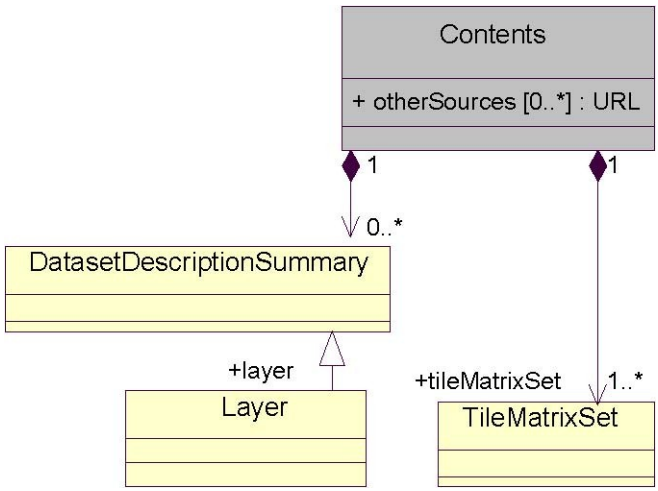

图5 — Contents子集的UML模型

Table 5 through Table 12 define the components of the layer section and Table 13 through Table 14 define the components of the tile matrix set section of the ServiceMetadata document. The UML class diagram in Figure 5 provides a useful graphical view of the composition of the Contents section.

The Contents section, described in Table 5, contains a list of layers available on the server and a list of tileMatrixSets. Each layer links to a particular tileMatrixSet using a reference to a tileMatrixSet identifier. Layers are described in Table 6 with a name, a title, an abstract description, keywords, a WGS84BoundingBox, a tileMatrixSet reference, a supported image format list, an infoFormat list, a metadata URL document link, and an optional dimensions list. In addition, a layer has one or more map portrayal representations that are called styles. Each style is described with a Style section as detailed in Table 7 with a name, a title, an abstract and a list of legendURLs described in Table 8. Each legendURL provides an iconic representation of the layer in its style, suitable for display in a legend; it specifies the URL of the icon image, and optionally the width and height of the icon image and the range of scales for which the icon is appropriate, as described in Table 8. Optional dimensions of the layer are described in Table 9. Dimensions are described by an identifier, a title, an abstract, units and unit symbols, a list of possible values and a default value. Typical examples of dimension identifiers are "time", "elevation" and "band", but the service can define any other dimension property that exists in the multidimensional layer collection being served.

面向资源架构风格的HTTP编码不在OperationsMetadata子集中说明，而是使用ResourceURL和ServiceMetadataURL来表明对这种架构风格的支持，第10章将做详细说明。

7.1.1.1.2 *Contents* section contents *Contents*子集

ServiceMetadata文档的Contents子集包含关于该服务器提供服务的数据的元数据。对于WMTS，该Contents子集包含对可访问图层的一般性描述和对附加的维度、图式符号、图像格式以及用于每个图层的块阵集的说明。Contents子集应包含表5到表14规定的所有参数。

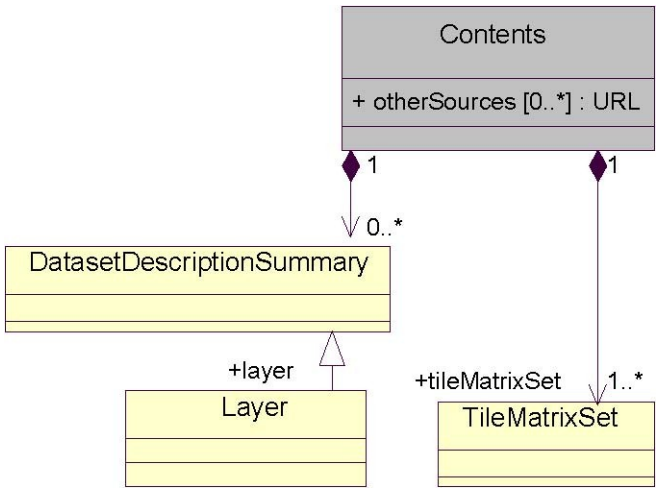

图5 — Contents子集的UML模型

表5到表12定义layer子集的各个组成部分，表13和表14定义ServiceMetadata文档的块阵集子集的各组成部分。图5的UML类图对Contents子集的组成提供了一个很有用的图形视图。

表5描述的Contents子集包含服务器上可访问的图层和tileMatrixSet（块阵集）列表。每个图层通过一个对tileMatrixSet标识符的引用链接到一个特定的tileMatrixSet。Layer子集在表6中说明，包括名称（name）、标题（title）、摘要(abstract)、关键词（keyword）、基于WGS84坐标系的范围框（WGS84BoundingBox）、tileMatrixSet引用、支持的图像格式列表、返回信息的格式（infoFormat）、元数据文件的URL链接以及一个可选的维度列表。此外，一个图层可以有一或多个地图图示表达方式，称为图式。每一个图式使用一个Style子集进行描述，如表7所示，包括名称（name）、标题（title）、摘要（abstract）和表8中说明的图例URL地址（legendURL）的列表。每个legendURL提供一个图层采用该图式符号化后的一个图标，适用于在图例中进行显示；legendURL指定该图标文件的URL地址，并可指定其宽度和高度以及该图标适合的比例尺范围，具体在表8中说明。图层的其他可选维度在表9中说明。维度通过标识符、标题（title）、摘要（abstract）、单位（units）和单位符号，以及一个可能取值的列表和默认值。维度标识符的典型例子如"time"、"elevation" 和"band"（分别表示时间、高程和波段），但服务可以定义在服务器上的多维图层集合中存在的任何其他维度特性。

Table 5 — Parts of the Contents section

Names	Definition	Data type and values	Multiplicity and use
layer `Layer`	Metadata describing one top-level dataset available from this server	Layer data structure, see Table 6	Zero or more (optional) [a] One for each dataset available
otherSource `OtherSource`	Reference to another source of contents metadata	See CI_OnlineResource class in ISO 19115	Zero or more (optional)
tileMatrixSet `TileMatrixSet`	A description of the geometry of a tile cut	TileMatrixSet data structure, see Table 13	Zero or more (optional)
a SHALL be included unless Other Source parameter(s) are included and all this metadata is available from those sources.			

The OtherSource parameter may reference one or more catalogue servers from which dataset metadata is available. This ability is expected to be used by servers with a very large number of datasets, for which searching a catalogue is more feasible than retrieving and then searching a very large ServiceMetadata XML document. When there is no Layer section in the Contents section of the ServiceMetadata document, the otherSource parameter SHALL reference one or more catalogue servers that contain current metadata summaries for all the datasets currently available from this WMTS server, with the metadata for each dataset referencing this WMTS server.

The UML class diagram in Figure 6 provides a useful graphical view of the Layer section with its properties, complex data types and dependencies.

表5 —Contents 子集的组成部分

名称	定义	数据类型与取值	重复次数与用法
layer Layer	关于该服务器上一个顶层数据集的元数据	Layer的数据结构，见表6	0或多（可选）a 每一个可访问数据集对应一个
otherSource OtherSource	对contents元数据的另一个来源的引用	见ISO 19115中CI_OnlineResource类的定义	0或多（可选）
tileMatrixSet TileMatrixSet	关于地图切块几何规则的说明	TileMatrixSet数据结构，见表13	0或多（可选）
a 如果包括其他来源才使用该参数，包括的话则所有这些元数据都由这些来源提供。			

OtherSource参数可以引用1或多个包含数据集元数据的目录服务器。该功能有望被具有大量数据集的服务器用到，因为这种情况下，搜索一个目录比访问并搜索一个非常大的ServiceMetadata的XML文档更具可行性。当ServiceMetadata文档中的Contents子集中没有Layers子集时，otherSource参数将引用1或多个目录服务器，这些目录服务器应包含关于该WMTS服务器上当前所有数据集的元数据概要，且这些数据集元数据同时引用了该WMTS服务器。

图6的UML类图给出了一个关于Layer子集的有用的图形视图，包括了它的特性、复杂数据类型以及依赖关系。

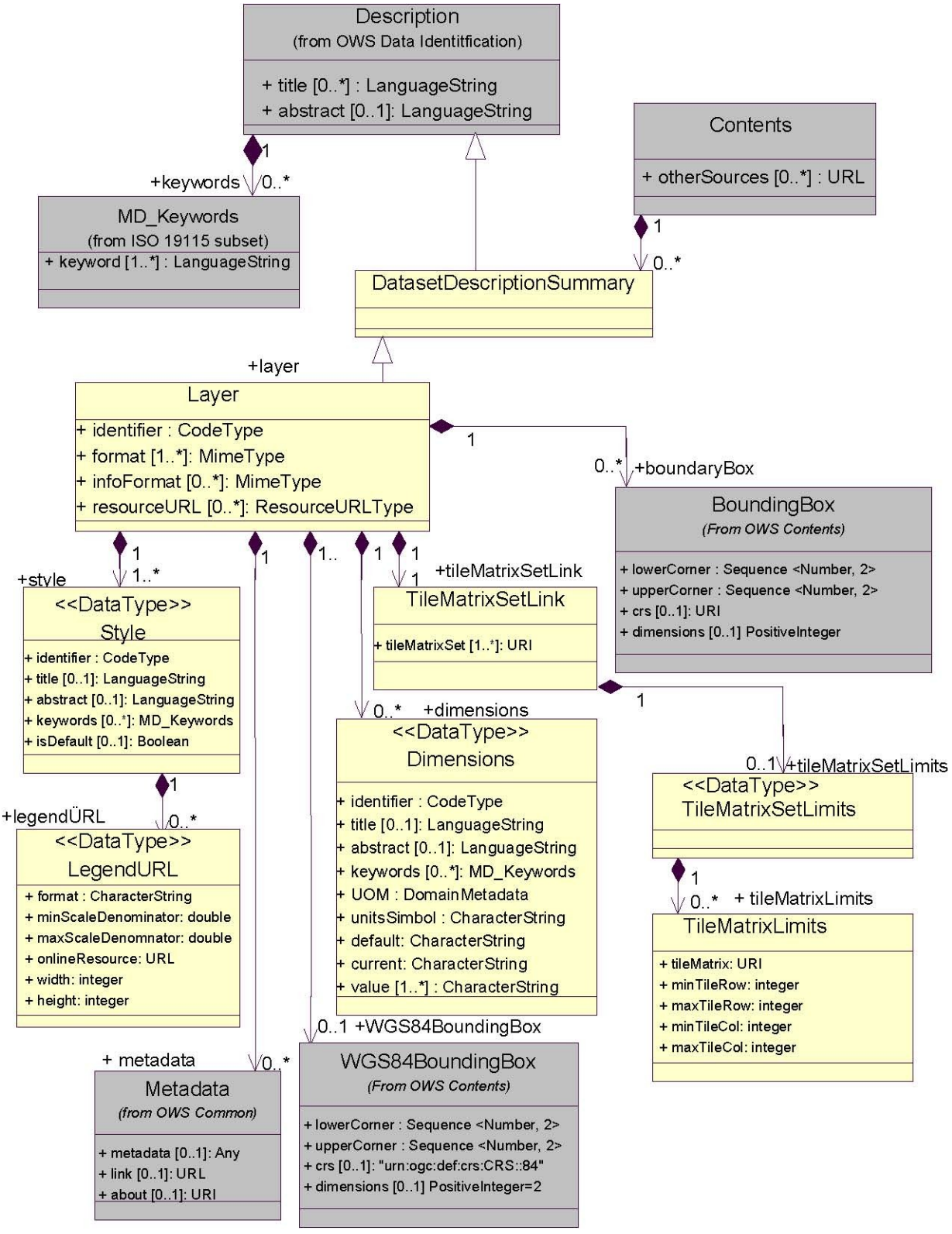

Figure 6 — Layer UML Modal

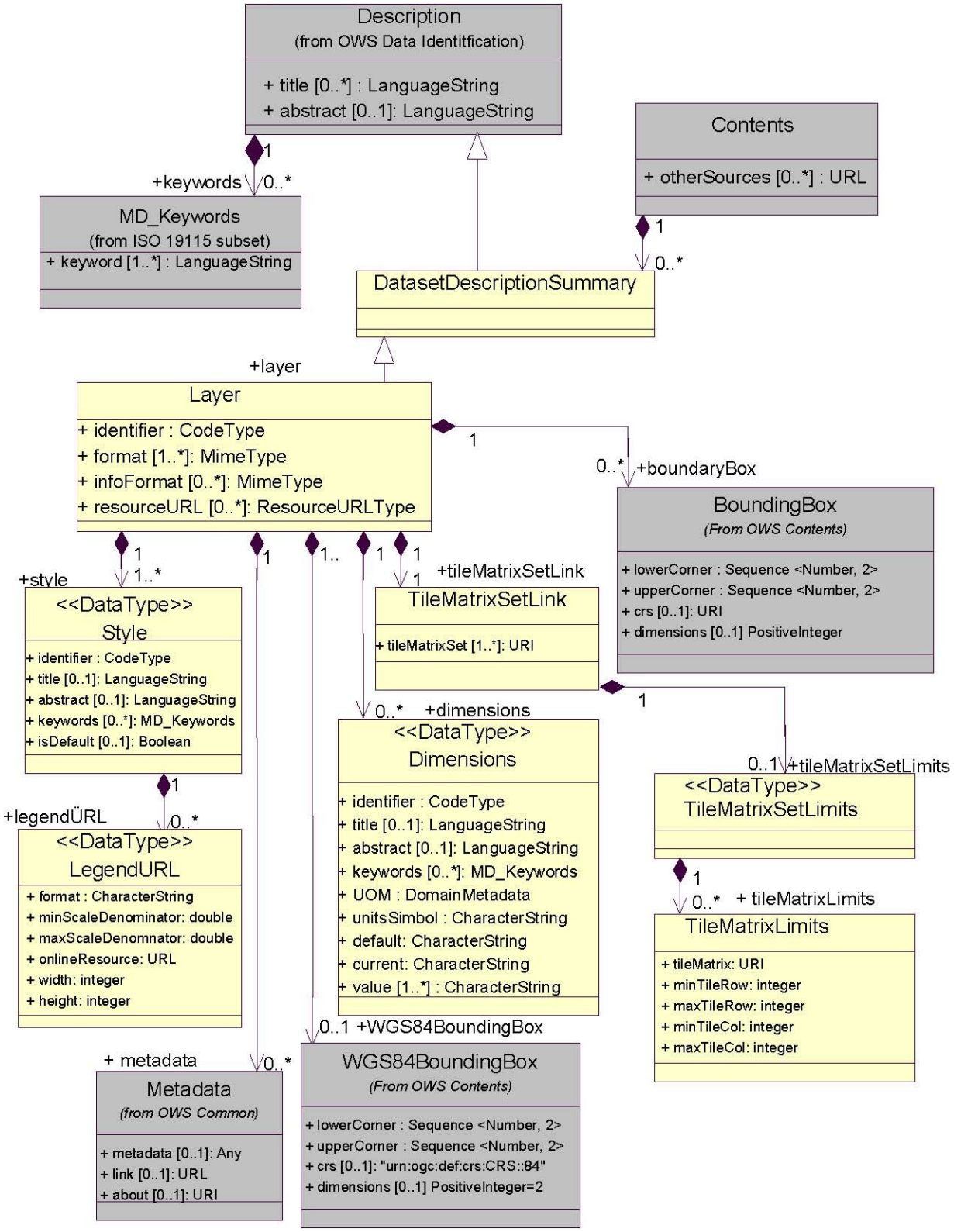

图6 — Layer的UML模型

Table 6 — Parts of Layer data structure

Names	Definition	Data type and values	Multiplicity and use
identifier [a] `Identifier`	An unambiguous reference to this layer,normally used by software [f]	ows:CodeType, as adaptation of MD_Identifier class ISO 19115	One (mandatory)
title [c] `Title`	Title of this layer, normally used for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented [e]
abstract[c] `Abstract`	Brief narrative description of this layer, normally available for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented
keywords [c] `Keywords`	Unordered list of one or more commonly used or formalised word(s) or phrase(s) used to describe this dataset	MD_Keywords class in ISO 19115	Zero or more (optional) One for each keyword authority used
wgs84BoundingBox `WGS84BoundingBox`	Minimum bounding rectangle surrounding dataset, using WGS 84 CRS with decimal degrees and longitude before latitude [b]	WGS84Bounding Box data structure see subclause 10.2 of OWS Common [OGC 06-121r3]	Zero or one (optional)
boundingBox `BoundingBox`	Minimum bounding rectangle surrounding the layer, in the supported CRS [g]	BoundingBox data structure, see subclause 10.2 of OWS Common [OGC 06-121r3]	Zero or more (optional)
style `Style`	Description of the style that has been applied to this layer	Style data structure, see Table 7	One or more (mandatory)
format `Format`	Supported valid output formats for a tile request	ows:MimeType	One or more (mandatory)
infoFormat `InfoFormat`	Supported valid output formats for a FeatureInfo document request	ows:MimeType	Zero or more (optional) [d]
dimension `Dimension`	Extra dimensions for a tile and FeatureInfo resource requests	Dimension data structure, see Table 9	Zero or more (optional) One for each extra dimension available.
metadata `Metadata`	Additional metadata about this dataset	ows:Metadata	Zero or more (optional) One for each useful metadata object
tileMatrixSetLink `TileMatrixSetLink`	Reference to a tileMatrixSet and limits	TileMatrixSetLink data structure, see Table 10	One or more (mandatory)
resourceURL `ResourceURL`	URL template to a tile or a FeatureInfo resource	URLTemplate data structure, see Table 31	0 or more (optional) Include one or more in resource oriented architectural style

a This has the same meaning as "name" in WMS but has been replaced by "identifier" to harmonize with OWS Common
b This WGS84BoundingBox can be approximate, but SHOULD be as precise as practical.
c The multilingual scoping rules in subclause 10.7.3 of OWS Common [OGC 06-121r3] SHALL apply.
d If no infoFormats are specified, then the layer is not queryable (*i.e.*, a request for a FeatureInfo is not permitted for this layer)
e If no Title is specified, client may display the Identifier value instead.
f Layer identifies SHALL be unique (different) for each layer of this server
gIt represents the area where this layer is represented. It could seem redundant with the bounding box of the tile matrix set but in complex cases that limits the area with data using tileMatrixLimits it is not so easy to calculate form the tile matrix set parameters.

表6 —Layer 数据结构的各组成部分

名称	定义	数据类型与取值	出现次数与用法
identifier [a] Identifier	对该图层无二义性的引用，通常由软件使用[f]	ows:CodeType，由ISO 19115中MD_Identifier类演变而来	1(必选)
title [c] Title	图层的标题，常用来显示给人看	LanguageString，见OWS通用实现规范[OGC 06-121r3]中的图15	0或多次 (可选) 每种语言一个[e]
abstract[c] Abstract	该图层的简短描述，常用来显示给人看	LanguageString，见OWS通用实现规范[OGC 06-121r3]中的图15	0或多次 (可选) 每种语言一
keywords [c] Keywords	用来描述该数据集的1或多个通用规范化的词语或短语	ISO 19115 定义的MD_Keywords类	0或多次 (可选) 每个用到的关键词定义机构采用一个
wgs84BoundingBox WGS84BoundingBox	数据集的最小包围矩形，使用WGS84坐标参照系统，十进制度，经度在前纬度在后[b]	WGS84Bounding Box data structure see subclause 10.2 of OWS Common [OGC 06-121r3]	0或多次（可选）
boundingBox BoundingBox	图层的的最小包围矩形，使用图层支持的坐标参照系统[g]	BoundingBox数据结构，见OWS通用实现规范[OGC 06-121r3]的10.2	0或多次（可选）
style Style	对使用本图层的图式的说明	Style数据结构，见表7	1或多次（必选）
format Format	图块请求所支持的有效输出格式	ows:MimeType	1或多次（必选）
infoFormat InfoFormat	FeatureInfo请求所支持的有效输出格式	ows:MimeType	0或多次（可选） [d]
dimension Dimension	图块和FeatureInfo资源请求支持的其他维度	Dimension，见表9	0或多次（可选）每一个维度一个
metadata Metadata	关于该数据集的其他元数据	ows:Metadata	0或多次（可选）每个Metadata对象一个
tileMatrixSetLink TileMatrixSetLink	对tileMatrixSet和限定的引用	TileMatrixSetLink，见表10	1或多次（必选）
resourceURL ResourceURL	图块或FeatureInfo资源的URL模板	URLTemplate，见表31	0或多次（可选）面向资源架构风格下包含1或多个。

a 与WMS中的“name”具有相同的含义，但为了与OWS通用实现规范一致，换成了"identifier"。
b 这里的WGS84BoundingBox可能是大概的，但应尽可能准确.
c 应符合OWS通用实现规范[OGC 06-121r3] 中10.7.3的多语言限定规则的要求。
d 如果没有指定infoFormat，则该图层不支持查询（即不允许对该图层发送FeatureInfo请求）。
e 如果Title没有指定，客户端可以使用identifier替代显示。.
f 服务器上每个图层的标识符应是唯一的
g它代表了该图层所代表的区域。块阵集的范围框可能会有冗余，但在复杂情形下，使用tileMatrixLimits 限定相应的区域，要通过块阵集的参数来计算覆盖范围并非易事。

The list of output formats SHOULD be chosen carefully. A long list of formats will improve interoperability with clients but will reduce the effectiveness of caching mechanisms. As a general rule, servers should use a short list, should avoid including redundant formats in the list and should use the formats recommended in subclause 11.3.

NOTE 1 In WMTS the list of supported output formats can be different for each layer, in contrast with WMS which specifies a shared single list of supported formats for all layers. WMTS layers have been given this ability because different layers may have different optimal formats. The use of a shared single list would force layers to be offered in all declared formats, reducing scalability and performance.

NOTE 2 The UML class diagram contained in the Annex C4 provides a useful graphical view of the contents of the Contents section listed in Tables 6 - 16.

Table 7 — Parts of Style data structure

Names	Definition	Data type and values	Multiplicity and use
identifier[a] `Identifier`	An unambiguous reference to this style, identifying a specific version when needed, normally used by software [d]	ows:CodeType, as adaptation of MD_Identifier class ISO 19115	One (mandatory)
title [b] `Title`	Title of this style, normally used for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented [c]
abstract[b] `Abstract`	Brief narrative description of this style, normally available for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented
keywords [b] `Keywords`	Unordered list of one or more commonly used or formalised word(s) or phrase(s) used to describe this dataset	MD_Keywords class in ISO 19115	Zero or more (optional) One for each keyword authority used
legendURL `LegendURL`	Description of an image that represents the legend of the map	LegendURL data structure, see Table 8	Zero or more (optional) Include when available and useful
isDefault `isDefault`	The style that a client SHOULD select as the first choice (default style)	Boolean	Zero or one (optional) Default is "false" e

a This has the same meaning as "name" in WMS but has been replaced by "identifier" to harmonize with OWS Common.
b The multilingual scoping rules in subclause 10.7.3 of OWS Common [OGC 06-121r3] SHALL apply.
c If no Title is specified, client may display the Identifier value instead.
d Style identifies SHALL be unique (different) for each style of a particular layer
e Only one style per layer can have a "true" value

输出格式的多少应仔细选择，较多的格式可以提高与客户端的兼容性但同时会降低缓存系统的效率。常用规则是，服务器端应支持较少的格式，应避免在列表中包含冗余的格式，应使用11.3中建议的格式。

注1 在WMTS中，每个图层支持的输出格式的列表可以互不相同，WMS则不然，所有图层只支持同一个格式列表。WMTS图层被赋予了这一能力是因为不同的图层可以具有不同的优化格式。使用单一共用的格式列表将迫使所有图层通过所有宣称的格式进行服务，这样会降低系统伸缩性和性能。

注2 附录C4中的UML类图提供了表6至表16中列出的Contents子集内容的图形视图。

表 7 —Style数据结构的各组成部分

名称	定义	数据类型与取值	出现次数与用法
identifier[a] Identifier	对该图式的无二义性引用，必要时指明特定的版本，通常由软件使用[d]	ows:CodeType，为ISO 19115中MD_Identifier 类的衍生	1(必选)
title [b] Title	该图式的标题，通常用于显示给人看	LanguageString数据结构，见OWS 通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 需要时包含 每种语言包含一个[c]
abstract[b] Abstract	该图式的简短描述，通常用于显示给人看	LanguageString数据结构，见OWS 通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 需要时包含 每种语言包含一个
keywords [b] Keywords	用于描述该数据集的1或多个词语或短语的无序列表	ISO 19115中的MD_Keywords类	0或多 (可选) 每个定义关键词的权威机构一个
legendURL LegendURL	代表地图图例的一个图片的说明	LegendURL数据结构，见表8	0或多 (可选) 需要时包含
isDefault isDefault	客户端应选做默认图式的图式	Boolean	0或1 (可选) 默认值为"false"[e]

a 与WMS中的“name”具有相同的含义，但为了与OWS通用实现规范一致，换成了"identifier"。
b 应符合OWS通用实现规范[OGC 06-121r3] 中10.7.3的多语言限定规则的要求。
c 如果Title没有指定，客户端可以使用identifier替代显示。.
d 图层的每个图式的标识符应是唯一的。
e 每个图层只能有一个图式可以取值为“true”。

A WMTS ServiceMetadata document may include zero or more LegendURL elements to provide an image(s) of a legend relevant to each style of a layer. Clients can show this image to the user as a visual summary of the information rendered in the tiles. The legend image should clearly represent the symbols, lines and colors used in the portrayal of the tiles and their meanings. The legend image should not contain text that duplicates the title of the layer, because that information is known to the client and may be shown to the user by other means.

Table 8 — Parts of LegendURL data structure

Names	**Definition**	**Data type and values**	**Multiplicity and use**
format `format`	A supported output format for the legend image	ows:MimeType	One (mandatory)
minScaleDenominator `minScaleDenominator`	Minimum scale denominator (inclusive) for which this legend image is valid	Double type, not empty	Zero or one (optional) Include when available and useful [a]
maxScaleDenominator `maxScaleDenominator`	Maximum scale denominator (exclusive) for which this legend image is valid	Double type, not empty	Zero or one (optional) Include when available and useful [a]
href `href`	The URL from which the legend image can be retrieved	URL type	One (mandatory)
width `width`	Width (in pixels) of the legend image	Positive integer type not empty	Zero or one (optional) Include when available and useful
height `height`	Height (in pixels) of the legend image	Positive integer type not empty	Zero or one (optional) Include when available and useful
a The minScaleDenominator and maxScaleDenominator define the range of scales where this legend is valid. The absence of a minScaleDenominator parameter means there is no minimum scale denominator to the condition or logically that the default value is 0. The absence of a maxScaleDenominator parameter means that there is no maximum scale denominator to the condition or logically that the default value is infinity. The absence of both scale parameters in LegendURL metadata means that there is no scale constraint and that the LegendURL is applicable to the style at all scales. General considerations about the meaning of minScaleDenominator and maxScaleDenominator values and their pixel size equivalences as explained in subclause 6.1 also apply here.			

In case of multi-dimensional data, the service metadata can describe their multi-dimensionality and tiles can be requested at specific values in these dimensions. Examples of dimensions are Time, Elevation and Band. Optional parameters in WMTS service metadata declare available values along one or more dimensional axes applicable to a Layer. GetTile and GetFeatureInfo requests for that layer should include parameters specifying dimensional value(s).

WMTS的ServiceMetadata文档可包含0或多个LegendURL元素以提供与图层的各个图式相关的图例图片。客户端可以以此作为图块中表达信息的可视化样例向用户显示该图例。图例图片应清楚表示图块图示表达中使用的符号、线划和颜色及其含义。图例图片不应包含与图层名称一样的文本，因为这样的信息客户端已经知道，且可以通过其它方式显示给用户。

表 8 —LegendURL数据结构的组成部分

名称	定义	数据类型与取值	出现次数与用法
format format	支持的图例图片输出格式	ows:MimeType	1(必选)
minScaleDenominator minScaleDenominator	该图例图片有效的最小比例尺分母（含）	Double，非空	0或1 (可选) 需要时包含 a
maxScaleDenominator maxScaleDenominator	该图例图片有效的最大比例尺分母（不含）	Double，非空	0或1 (可选) 需要时包含 a
href href	可以获取到图例图片的URL	URL 类型	1(必选)
width width	图例图片的宽（像素为单位）	正整数，非空	0或1 (可选) 需要时包含
height height	图例图片的高（像素为单位）	正整数，非空	0或1 (可选) 需要时包含
a minScaleDenominator 和 maxScaleDenominator定义了该图例图片有效的比例尺范围。如果没有minScaleDenominator，表示取默认值0.如果没有maxScaleDenominator，表示取值为无穷大。如果都没有，表示没有比例尺约束，该图例图片适用所有比例尺。6.1中给出的关于minScaleDenominator和maxScaleDenominator的取值含义及其对应像素大小的一般性说明也适用于此。			

对于具有多个维度的数据，服务元数据可以描述其支持的维度以及可以通过指定这些维度值请求的图块。维度可以包括时间、高程以及波段。WMTS服务元数据中的可选参数申明了某一图层在一或多个维度轴下的可用取值。针对该图层的GetTile和GetFeatureInfo请求应包含指定维度值的参数。

Table 9 — Parts of Dimension data structure

Names	Definition	Data type and values	Multiplicity and use
identifier[a] `Identifier`	A name of dimensional axis [e]	ows:CodeType, as adaptation of MD_Identifier class ISO 19115	One (mandatory)
title [b] `Title`	Title of this dimension, normally used for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented [d]
abstract[b] `Abstract`	Brief narrative description of this dimension, normally available for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented
keywords[b] `Keywords`	Unordered list of one or more commonly used or formalised word(s) or phrase(s) used to describe this dataset	MD_Keywords class in ISO 19115	Zero or more (optional) One for each keyword authority used
UOM `UOM`	Units of measure of dimensional axis	DomainMetadata data structure, see Table 43 in OWS Common [OGC 06-121r3]	Zero or one (optional) Include when available and useful
unitSymbol UnitSymbol	Symbol of the units	Character String type, not empty	Zero or one (optional) Include when available and useful
default `Default`	Default value that will be used if a tile request does not specify a value or uses the keyword 'default'	Character String type, not empty (with the exception of 'default' or 'current')	One (mandatory)
current `Current`	A value of 'true' indicates (a) that temporal data are normally kept current and (b) that the request value of this dimension accepts the keyword 'current'	Boolean	Zero or one (optional) Include only for temporal extents. Default is 'false'
value `Value`	Indicates an available value for this dimension	Character String type, not empty	one or more (mandatory)[c] One of each dimension value

a This has the same meaning as "name" in WMS but has been replaced by "identifier" to harmonize with OWS Common.
b The multilingual scoping rules in subclause 10.7.3 of OWS Common [OGC 06-121r3] SHALL apply.
c Repeat this parameter for each available value for this dimension.
d If no Title is specified, client may display the Identifier value instead
e Dimension identifies SHALL be unique (different) for each layer of this server

NOTE 3The WMS content of a dimension section has a property called multiValues to inform the client that server supports the requesting of multiple values at the same time. The WMTS request for a tile does not support this as a possible property for the dimension data type in this standard.

表9 —Dimension数据结构各组成部分

名称	定义	数据类型与取值	出现次数与用法
identifier[a] Identifier	维度轴的名称[e]	ows:CodeType，为ISO 19115中MD_Identifier 类的衍生	1(必选)
title [b] Title	该维度的标题，通常显示给人看	LanguageString数据结构，见OWS 通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 必要时才包含 每个语言包含一个[d]
abstract[b] Abstract	该维度的简短说明，通常显示给人看	LanguageString数据结构，见OWS 通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 必要时才包含 每个语言包含一个
keywords[b] Keywords	描述该数据集的词语或短语的无序列表	ISO 19115中的MD_Keywords类	0或多 (可选) 每个用到的关键词定义机构一个
UOM UOM	维度轴的度量单位	DomainMetadata，见OWS 通用实现规范[OGC 06-121r3]的表43。	0或1(可选) 必要时才包含
unitSymbol UnitSymbol	单位的符号	字符串类型，非空	0或1(可选) 必要时才包含
default Default	如果图块请求中没有指定取值或使用了关键词“default”是所使用的默认值	字符串类型，非空(除'default' 或'current')	1(必选)
current Current	取值为‘true’表示（a）时态数据正常为实况（b）该维度的请求值接受关键词‘'current'’	Boolean	0或1 (可选) 只有时间维度才包含，默认值为‘false’
value Value	表示该维度可用的取值	字符串类型，非空	1或多(必选)[c] 每个维度值一个
a 与WMS中的“name”具有相同的含义，但为了与OWS通用实现规范一致，换成了"identifier"。 b 应符合OWS通用实现规范[OGC 06-121r3] 中10.7.3的多语言限定规则的要求。 c 对每一个该维度的可用取值重复该参数。 d 如果Title没有指定，客户端可以使用identifier替代显示。. e 该服务器上每个图层的维度的标识符应是唯一的。			

注3 WMS中维度部分有一个叫做multiValues的特性，用来告知客户端服务器支持同时请求多个值。本标准中对图块的WMTS请求中的维度数据类型不支持该特性。

Table 10 — Parts of TileMatrixSetLink data structure

Names	Definition	Data type and values	Multiplicity and use
tileMatrixSet `TileMatrixSet`	Reference to a tileMatrixSet	URI type Values SHALL be an tileMatrixSet identifieir in service metadata document	One (mandatory)
tileMatrixSetLimits `TileMatrixSetLimits`	Index limits for this tileMatrixSet	TileMatrixSetLimits data structure, see Table 11	Zero or one (optional) Should be include when the boundary of the data is a fragment of the boundary of the tileMatrixSet [a]
a The absence of this parameter means that tile row and tile column indices are only limited by 0 and the corresponding matrixWidth and matrixHeight for each tileMatrix of the tileMatrixSet definition.			

Table 11 — Parts of TileMatrixSetLimits data structure

Names	Definition	Data type and values	Multiplicity and use
tileMatrixLimits `TileMatrixLimits`	Indices limits for this tileMatrix	TileMatrixLimits data structure, see Table 12	one or more (mandatory) [a]
a Multiplicity SHALL be the multiplicity of tileMatrix this tileMatrixSet.			

Table 12 — Parts of TileMatrixLimits data structure

Names	Definition	Data type and values	Multiplicity and use
tileMatrix `TileMatrix`	Reference to a tileMatrix identifier	URI type Values defined in service metadata [a]	One (mandatory)
minTileRow `MinTileRow`	Minimum tile row index valid for this layer.	Positive integer type b	One (mandatory)
maxTileRow `MaxTileRow`	Maximim tile row index valid for this layer.	Positive integer type c	One (mandatory)
minTileCol `MinTileCol`	Minimum tile column index valid for this layer.	Positive integer type d	One (mandatory)
maxTileCol `MaxTileCol`	Maximim tile column index valid for this layer.	Positive integer type e	One (mandatory)
a URI SHALL be an identifier to a tileMatrix section of this tileMatrixSet for this layer. b From 0 to maxTileRow c From minTileRow to matrixWidth-1 of the tileMatrix section of this tileMatrixSet d From 0 to maxTileCol e From minTileCol to tileHeight-1 of the tileMatrix section of this tileMatrixSet			

表10 —TileMatrixSetLink 数据结构的各组成部分

名称	定义	数据类型与取值	出现次数与用法
tileMatrixSet TileMatrixSet	对tileMatrixSet的引用	URI类型 取值应为服务元数据文档中的一个tileMatrixSet的标识符	1 (必选)
tileMatrixSetLimits TileMatrixSetLimits	针对该tileMatrixSet的下标限定范围	TileMatrixSetLimits数据结构，见表11	0或1 (可选) 当数据的边界只是tileMatrixSet的一部分时应当包含a
a 如果该参数缺失则意味着图块的行列下标介于0和tileMatrixSet的各个相应tileMatrix的高（Height）和宽（Width）之间。			

表 11 —TileMatrixSetLimits数据结构的组成部分

名称	定义	数据类型与取值	出现次数与用法
tileMatrixLimits TileMatrixLimits	该tileMatrix的下标范围	TileMatrixLimits数据结构，见表 12	1或多(必选) a
a 出现次数与tileMatrixSet的tileMatrix的出现次数一样。			

表 12 —TileMatrixLimits数据结构的组成部分

名称	定义	数据类型与取值	出现次数与用法
tileMatrix TileMatrix	对一个tileMatrix标识符的引用	URI 类型 取服务元数据中定义的值a	1 (必选)
minTileRow MinTileRow	对该图层有效的最小行号	正整型b	1 (必选)
maxTileRow MaxTileRow	对该图层有效的最大行号	正整型 c	1 (必选)
minTileCol MinTileCol	对该图层有效的最小列号	正整型d	1 (必选)
maxTileCol MaxTileCol	对该图层有效的最大列号	正整型e	1 (必选)
a URI应该是一个针对该图层的块阵集（tileMatrixSet）的一个块阵（tileMatrix）的标识符 b 从0到maxTileRow c 从该块阵集（tileMatrixSet）的块阵（tileMatrix）的minTileRow到matrixWidth-1 d 从0到maxTileCol e从该块阵集（tileMatrixSet）的块阵（tileMatrix）的minTileCol到matrixHeight-1			

A tileMatrixSet defines a generic tiled space bounding box through a TopLeftCorner and MatrixWidth and MatrixHeight as explained in clause 6. For practical reasons some layers that point to this tiled space might not have data covering the entire bounding box but have data covering only some rectangular subset. The optional TileMatrixSetLimits should be included in the description of the layer section to reflect this fact. A request for a tile outside the area marked on Figure 7 SHOULD result in an exception response.

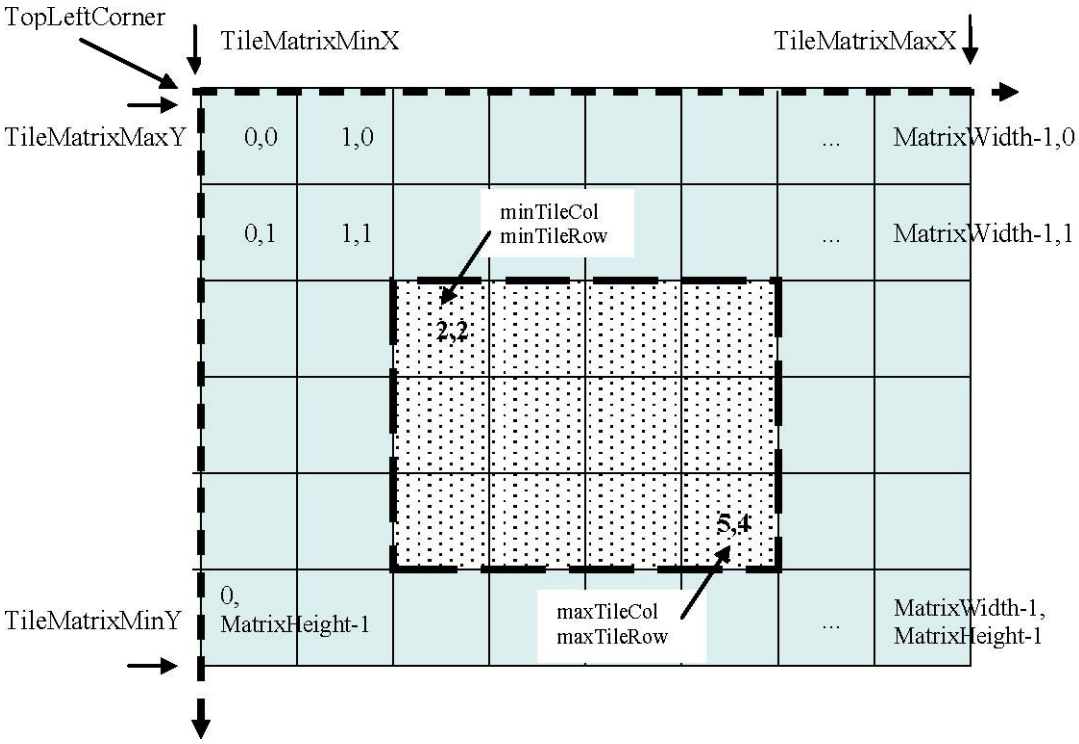

Figure 7 — Optional TileMatrix Limits

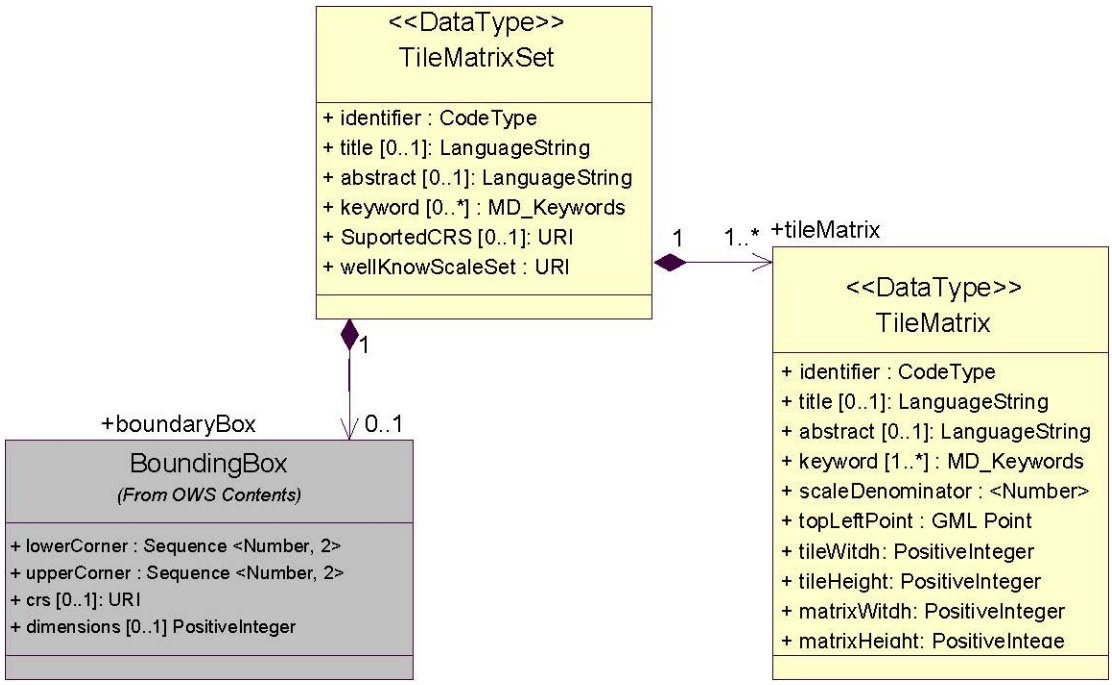

Figure 8 — TileMatrixSet UML model

tileMatrixSet通过第6章说明的TopLeftCorner、MatrixWidth和MatrixHeight定义了一个通用的空间范围框。实际上一些指向该图块空间的图层可能并不覆盖整个范围框，而只是覆盖局部范围，这时应在图层描述中包含可选的TileMatrixSetLimits来反映这种情况。如果请求图7标明区域外的图块则服务器应给出异常响应。

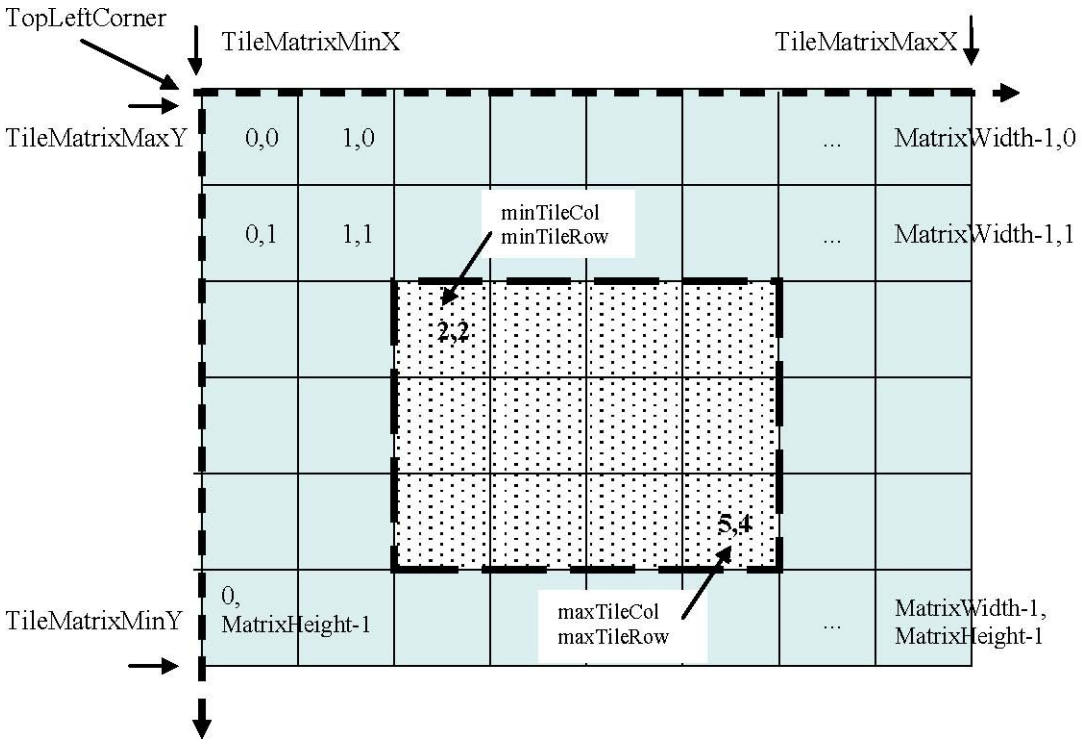

图7 — 可选的块阵范围限定

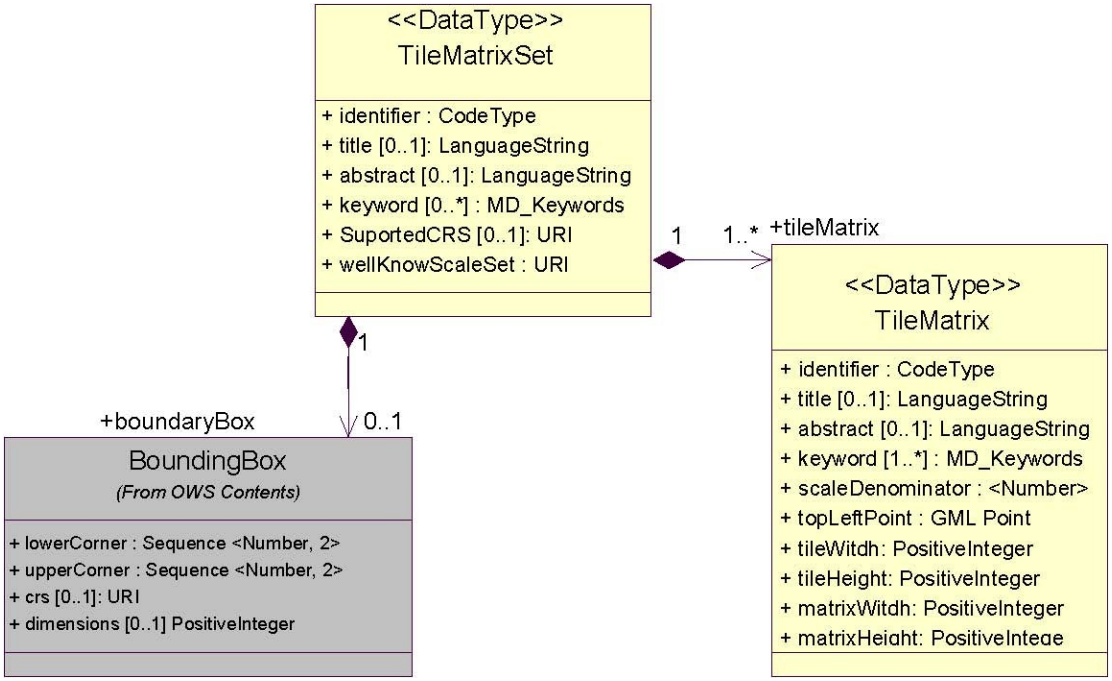

图8 — TileMatrixSet的UML模型

Each layer has one or more references to a TileMatrixSet identifier. The structure in Table 13 defines the structure of the TileMatrixSet sections in the Content section.

NOTE 4If a client requires all tiles to be aligned to a specific TileMatrixSet, it could choose to only display layers that share the same TileMatrixSet identifier. Alternatively, it could compare TileMatrixSet definitions for an equivalency (a simple calculation can be performed to verify whether or not two given tile matrices are aligned).

Table 13 — Parts of TileMatrixSet data structure

Names	Definition	Data type and values	Multiplicity and use
identifier `Identifier`	Tile matrix set identifier [g]	ows:CodeType, as adaptation of MD_Identifier class ISO 19115	One (mandatory)
title [a] `Title`	Title of this tile matrix set, normally used for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented [f]
abstract[a] `Abstract`	Brief narrative description of this tile matrix set, normally available for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented
keywords[a] `Keywords`	Unordered list of one or more commonly used or formalised word(s) or phrase(s) used to describe this dataset	MD_Keywords class in ISO 19115	Zero or more (optional) One for each keyword authority used
boundingBox `BoundingBox`	Minimum bounding rectangle surrounding the tile matrix set, in the supported CRS [b]	BoundingBox data structure, see subclause 10.2 of OWS Common [OGC 06-121r3]	Zero or one (optional)
supportedCRS `SupportedCRS`	Reference to one coordinate reference system (CRS)	URI type	One (mandatory)
wellKnownScaleSet `WellKnownScaleSet`	Reference to a well known scale sete	URI type	Zero or one (optional) [c]
tileMatrix `TileMatrix`	Describes a scale level and its tile matrix	TileMatrix data structure, see Table 14	One or more (mandatory) [d]

a The multilingual scoping rules in subclause 10.7.3 of OWS Common [OGC 06-121r3] SHALL apply.
b If available, it represents the area where the data is expected to be represented. This does not necessarily indicate a complete tile boundary (and therefore does not necessarily include the TopLeftCorner of the tile matrices).
c When a tile matrix set conforms to a well-known scale set it SHOULD reference it by its URI. The well-known scale set SHALL be consistent with the supportedCRS and with the scaleDenominators of the tileMatrix parameters.
d Commonly more than one. Each tileMatrix of a tileMatrixSet SHALL have a unique (different) scaleDenominator
e Some possible values are defined the in Annex E
f If no Title is specified, client may display the Identifier value instead
g TileMatrixSet identifies SHALL be unique (different) for each TileMatrixSet of this server
h The content of this parameter follows subclause 11.3 and annex D.14 of OWS Common [OGC 06-121r3]

每个图层有一或多个对TileMatrixSet标识符的引用。表13定义了Content子集中的TileMatrixSet子集的结构。

注4 如果客户端要求所有图块与某一个指定的TileMatrixSet一致，可以选择只显示那些具有相同TileMatrixSet标识符的图层。或者，可以比较各个TileMatrixSet的定义以判断是否一致（通过简单的计算即可检验两个给定块阵是否一致）。

表13 —TileMatrixSet数据结构各组成部分

名称	定义	数据类型与取值	出现次数与用法
identifier Identifier	块阵集的标识符[g]	ows:CodeType，ISO 19115定义的MD_Identifier类的衍生类	1 (必选)
title [a] Title	该块阵集的标题，通常显示给人看	LanguageString数据结构，见OWS 通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 必要时包含 每一种语言一个[f]
abstract[a] Abstract	该块阵集的简短说明，通常显示给人看	LanguageString数据结构，见OWS 通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 必要时包含 每一种语言一个
keywords[a] Keywords	描述该数据集的词语或短语的无序列表	ISO 19115的MD_Keywords类	0或多 (可选) 每一个关键词定义机构一个
boundingBox BoundingBox	包围块阵集的最小范围矩形，采用所支持的坐标参照系统[b]	BoundingBox数据结构，见OWS 通用实现规范[OGC 06-121r3]的10.2	0或1 (可选)
supportedCRS SupportedCRS	对一个坐标参照系统（CRS）的引用	URI 类型	1 (必选)
wellKnownScaleSet WellKnownScaleSet	对一个知名比例尺系列的引用	URI 类型	0或1 (可选) [c]
tileMatrix TileMatrix	对比例尺级别及其块阵的说明	TileMatrix数据结构，见表14	1或多(必选) [d]

a 应符合OWS通用实现规范[OGC 06-121r3] 中10.7.3的多语言限定规则的要求；
b 如果有，则表示期望数据表达的区域，这并不意味着一个完整的图块边界（因此不必一定包含块阵的TopLeftCorner）；
c 当块阵集与一个知名比例尺系列一致时，应使用其URI对其进行引用。该知名比例尺系列应与supportedCRS以及tileMatrix参数的scaleDenominators一致；
d 通常会超过一个。tileMatrixSet的每个tileMatrix应有一个唯一的scaleDenominator；
e 附录E定义了一些可能取值；
f 如果没有指定Title，客户端可以显示Identifier；
g 该服务器上每一个TileMatrixSe的标识符应该是唯一的；
h 该参数的内容遵循OWS通用实现规范[OGC 06-121r3]的11.3和附录D.14的要求。

Table 14 — Parts of TileMatrix data structure

Names	Definition	Data type and values	Multiplicity and use
identifier `Identifier`	Tile matrix identifier [c]	ows:CodeType, as adaptation of MD_Identifier class ISO 19115	One (mandatory)
title [a] `Title`	Title of this style, normally used for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented [d]
abstract[a] `Abstract`	Brief narrative description of this style, normally available for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented
keywords[c] `Keywords`	Unordered list of one or more commonly used or formalised word(s) or phrase(s) used to describe this dataset	MD_Keywords class in ISO 19115	Zero or more (optional) One for each keyword authority used
scaleDenominator `ScaleDenominator`	Scale denominator level of this tile matrix	Double type	One (mandatory)
topLeftCorner `TopLeftCorner`	Position in CRS coordinates of the top-left corner of this tile matrix	Ordered sequence of double values [b]	One (mandatory)
tileWidth `TileWidth`	Width of each tile of this tile matrix in pixels	Positive integer type	One (mandatory)
tileHeight `TileHeight`	Height of each tile of this tile matrix in pixels	Positive integer type	One (mandatory)
matrixWidth `MatrixWidth`	Width of the matrix (number of tiles in width)	Positive integer type	One (mandatory)
matrixHeight `MatrixHeight`	Height of the matrix (number of tiles in height)	Positive integer type	One (mandatory)

a The multilingual scoping rules in subclause 10.7.3 of OWS Common [OGC 06-121r3] SHALL apply.
b CRS will be inherited from the supportedCRS parameter of the parent TileMatrixSet. The order of these axes, shall be as specified by the supportedCRS. These are the precise coordinates of the top left corner of top left pixel of the 0,0 tile. See Figure 2.
c This TileMatrix identifiers SHALL be unique (different) within the context of the parent TileMatrixSet. Consider using a rounded scale denominator or a rounded pixel size as a value.
d If no Title is specified, client may display the Identifier value instead.
e In XML schemas ows:PositionType data type is used. See OWS 1.1 schemas (owsCommon.xsd)

NOTE 5It may be desirable to define a tile matrix set with some general-scale tile matrices in one CRS (*e.g.*, CRS:84) and with detailed-scale tile matrices in a different CRS (*e.g.*, LCC projection). However, this standard does not allow this. Each tile matrix set SHALL declare a single CRS. You could define two tile matrix sets for the same layer instead.

表14 —TileMatrix数据结构的各组成部分

名称	定义	数据类型与取值	出现次数与用法
identifier Identifier	块阵的标识符c	ows:CodeType, ISO 19115的MD_Identifier 类的衍生类	1 (必选)
title a Title	该块阵的标题，通常显示给人看	LanguageString数据结构，见OWS通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 必要时包含 每一种语言一个f
abstracta Abstract	该块阵的简短说明，通常显示给人看	LanguageString数据结构，见OWS通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 必要时包含 每一种语言一个
keywordsc Keywords	描述该数据集的词语或短语的无序列表	ISO 19115的MD_Keywords类	0或多 (可选) 每一个关键词定义机构一个
scaleDenominator ScaleDenominator	该块阵的比例尺分母级别	双精度型	1 (必选)
topLeftCorner TopLeftCorner	块阵的左上角点在CRS下的坐标	双精度值的有序序列b	1 (必选)
tileWidth TileWidth	块阵个图块以像素为单位的宽度	正整数	1 (必选)
tileHeight TileHeight	块阵个图块以像素为单位的高度	正整数	1 (必选)
matrixWidth MatrixWidth	块阵的宽度，以图块为单位	正整数	1 (必选)
matrixHeight MatrixHeight	块阵的高度，以图块为单位	正整数	1 (必选)

a 应符合OWS通用实现规范[OGC 06-121r3] 中10.7.3的多语言限定规则的要求；
b CRS将从上级TileMatrixSet的supportedCRS继承而来，各个轴的顺序与supportedCRS指定的一致。此为0,0处左上角像素的左上角点的精确坐标，见图2；
c 该TileMatrix的标识符在上级TileMatrixSet的范围内是唯一的。可以考虑使用对比例尺分母或像素大小取整后的值作为标识符的值。
d 如果没有指定Title，客户端可以显示Identifier；
e 在XML模式中，使用ows:PositionType数据类型。见OWS 1.1的模式 (owsCommon.xsd)

注5 最好是能够定义一个块阵集，既可以包含使用同一个CRS（如CRS:84）、具有通用比例尺的一些块阵，也可以包含使用不同CRS（如LCC投影）的、具有细节比例尺的块阵。但是，本标准不支持这一点。每个块阵集只申明一个CRS。不过，你可以为同一个图层定义两个块阵集。

NOTE 6The width and height in tiles of each tile matrix is explicitly given, so the range of relevant tile indexes does not have to be calculated by the client application.

NOTE 7The bounding box of a tile matrix is not supplied explicitly because it can be calculated from topLeftCorner, tileWidth, tileHeight and scaleDenominator.

7.1.1.1.3 *Themes* section contents

The optional Themes section of a WMTS service metadata document SHALL contain data about how layers are organized thematically.

The WMTS standard proposes a different approach from WMS for layer organization, an approach based on the idea of themes. In the Contents section of WMTS, layers are represented as a linear list without hierarchy, and a hierarchy of themes is specified separately in the Themes section, removing the need to specify complex inheritance rules for layer properties. This separates both concepts and makes it easy for a client to ignore the theme hierarchy or even to force another layer organization. Also it allows servers to offer more than one layer organization (in more than one themes section).

Each theme has a human-readable description (*i.e.*, a title) and a list of layer references and child themes. It is possible for a layer to be a member of more than one theme, and for a layer to exist without being a member of any theme.

The Themes sections SHALL include the parameters specified in Table 15 and Table 16.

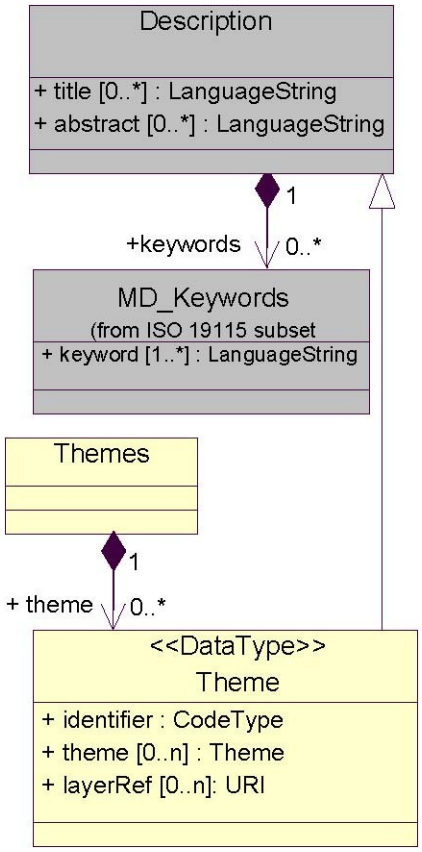

Figure 9 — Themes UML model

注6 每个块阵的宽和高需明确给定，这样客户端不需要再计算图块行列号的范围；

注7 块阵的范围框没有明确给定，是因为可以根据topLeftCorner, tileWidth, tileHeight 和scaleDenominator计算得出。

7.1.1.1.3 *Themes*子集的内容

WMTS服务元数据文档中可选的Themes子集应包含图层如何按专题组织的信息。

WMTS标准提出了不同于WMS的一种图层组织方法，即基于专题思想的方法。在WMTS的Contents子集中，图层表示成为没有层级的线性列表，以及单独在Themes子集中说明的专题层级，避免了为图层特性规定复杂的继承规则。这样分离了两个概念并且很容易让客户端忽略专题层级关系或者构建其他图层组织结构。同时可以让服务器提供1种以上的图层组织方式（通过提供多个Themes子集）。

每个专题有一个人可以理解的说明（如标题）以及一串图层引用和下层专题。一个图层可以属于1个以上的专题，也可以不属于任何一个专题。

Themes子集应包括表15和表16中规定的参数。

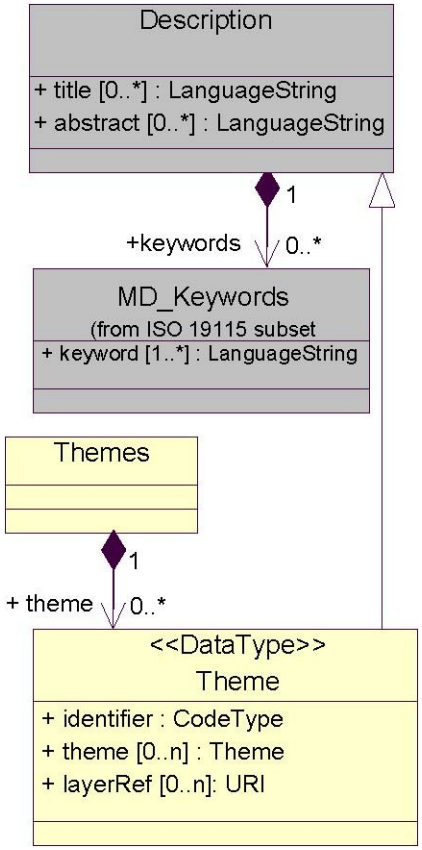

图 9 — Themes的UML模型

Table 15 — Parts of the Themes section

Names	Definition	Data type and values	Multiplicity and use
theme Theme	Metadata describing the top-level themes where layers available on this server can be classified	Theme data structure, see Table 16	Zero or more (optional) [a] One for each top-level theme available

Table 16 — Parts of Theme data structure

Names	Definition	Data type and values	Multiplicity and use
identifier Identifier	Name of the theme	ows:CodeType, as adaptation of MD_Identifier class ISO 19115	One (mandatory)
title [a] Title	Title of this theme, normally used for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented [c]
abstract[a] Abstract	Brief narrative description of this theme, normally available for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r3]	Zero or more (optional) Include when available and useful Include one for each language represented
keywords[a] Keywords	Unordered list of one or more commonly used or formalised word(s) or phrase(s) used to describe this dataset	MD_Keywords class in ISO 19115	Zero or more (optional) One for each keyword authority used
theme Theme	Metadata describing the child (subordinate) themes of this theme where layers available on this server can be classified	Theme data structure, see this Table	Zero or more (optional) [a] One for each theme available
layerRef LayerRef	Reference to layer	URI type Values defined in service metadata [b]	Zero or more (optional)
a The multilingual scoping rules in subclause 10.7.3 of OWS Common [OGC 06-121r3] SHALL apply. b A layer identifier on this ServiceMetadata document. c If no Title is specified, client may display the Identifier value instead.			

7.1.1.2 ServiceMetadata document XML schema

ServiceMetadata documents can be encoded in XML. This standard provides XML schemas to encode XML Service metadata documents as described in Annex B. The following XML schema fragment for a WMTS service metadata document shows how WMTS extends ows:CapabilitiesBaseType in owsCommon.xsd of OWS Common [OGC 06-121r3] to include other parameters as previously described in Table 3:

表 15 —Themes 子集的组成部分

名称	定义	数据类型与取值	出现次数与用法
theme Theme	说明针对该服务器上可分类图层的顶层专题的元数据	Theme数据结构，见表16	0或多 (可选) a 每一个顶层专题一个

Table 16 — Theme数据结构的组成部分

名称	定义	数据类型与取值	出现次数与用法
identifier Identifier	专题的名称	ows:CodeType, as adaptation of MD_Identifier class ISO 19115	One (mandatory)
title a Title	该块阵的标题，通常显示给人看	LanguageString数据结构，见OWS通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 必要时包含 每一种语言一个f
abstracta Abstract	该块阵的简短说明，通常显示给人看	LanguageString数据结构，见OWS通用实现规范[OGC 06-121r3]的图15	0或多 (可选) 必要时包含 每一种语言一个
keywordsa Keywords	描述该数据集的词语或短语的无序列表	ISO 19115的MD_Keywords类	0或多 (可选) 每一个关键词定义机构一个
theme Theme	描述针对该服务器上可分类图层的该专题下层专题的元数据	Theme数据结构，见本表	0或多 (可选) a 每一个专题一个
layerRef LayerRef	对图层的引用	URI类型 取服务元数据中定义的值b	0或多(可选)
a 应符合OWS通用实现规范[OGC 06-121r3] 中10.7.3的多语言限定规则的要求； b ServiceMetadata文档中的一个图层标识符 c d 如果没有指定Title，客户端可以显示Identifier的值；			

7.1.1.2 ServiceMetadata文档的XML模式

ServiceMetadata文档可以编码为XML。本标准在附录B中提供XML模式以用于编码XML的服务元数据文档。下面的WMTS服务元数据的XML模式片段说明WMTS是如何扩展OWS通用实现规范[OGC 06-121r3]的owsCommon.xsd中的ows:CapabilitiesBaseType以包含前面表3中说明的其他参数。

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:wmts="http://www.opengis.net/wmts/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
targetNamespace="http://www.opengis.net/wmts/1.0"
elementFormDefault="qualified" xml:lang="en">
 <annotation>
<documentation>
 This XML Schema Document defines the ServiceMetadata documentnamespace.
</documentation>
 </annotation>
 <!-- ==========================================================
includes and imports
 ========================================================== -->
 <import namespace="http://www.opengis.net/ows/1.1"
 schemaLocation="../../ows/1.1.0/owsAll.xsd"/>
 <!--==========================================================
elements and types
 ========================================================== -->
 <element name="Capabilities">
<complexType>
 <complexContent>
<extension base="ows:CapabilitiesBaseType">
 <sequence>
<element name="Contents" type="wmts:ContentsType"
minOccurs="0" />
<element ref="wmts:Themes" minOccurs="0"
maxOccurs="unbounded" />
<element name="WSDL" type="ows:OnlineResourceType"
minOccurs="0" maxOccurs="unbounded" />
<element name="ServiceMetadataURL"
type="ows:OnlineResourceType" minOccurs="0"
maxOccurs="unbounded" />
 </sequence>
</extension>
 </complexContent>
</complexType>
 </element>
...
</schema>
```

As indicated above, this XML Schema document uses the owsServiceIdentification.xsd, owsServiceProvider.xsd, and owsOperationsMetadata.xsd schemas specified in OWS Common [OGC 06-121r3]. It also uses an XML Schema document for the ―Contents‖ sections of the WMTS ServiceMetadata XML document, which is included in the wmtsGetCapabilities_response.xsd file. All these XML Schema documents contain documentation of the meaning of each element, attribute and type, and this documentation SHALL be considered normative as specified in subclause 11.6.3 of OWS Common [OGC 06-121r3].

Annex B contains more details on this normative set of XML Schema documents.

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:wmts="http://www.opengis.net/wmts/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
targetNamespace="http://www.opengis.net/wmts/1.0"
elementFormDefault="qualified" xml:lang="en">
 <annotation>
<documentation>
 This XML Schema Document defines the ServiceMetadata documentnamespace.
</documentation>
 </annotation>
 <!-- ==========================================================
includes and imports
 ========================================================== -->
 <import namespace="http://www.opengis.net/ows/1.1"
 schemaLocation="../../ows/1.1.0/owsAll.xsd"/>
 <!--==========================================================
elements and types
 ========================================================== -->
 <element name="Capabilities">
<complexType>
 <complexContent>
<extension base="ows:CapabilitiesBaseType">
 <sequence>
<element name="Contents" type="wmts:ContentsType"
minOccurs="0" />
<element ref="wmts:Themes" minOccurs="0"
maxOccurs="unbounded" />
<element name="WSDL" type="ows:OnlineResourceType"
minOccurs="0" maxOccurs="unbounded" />
<element name="ServiceMetadataURL"
type="ows:OnlineResourceType" minOccurs="0"
maxOccurs="unbounded" />
 </sequence>
</extension>
 </complexContent>
</complexType>
 </element>
...
</schema>
```

如上所示，该XML模式文档使用了OWS通用实现规范[OGC 06-121r3]中规定的owsServiceIdentification.xsd、owsServiceProvider.xsd和owsOperationsMetadata.xsd模式。也使用了WMTS服务元数据XML文档的Contents子集的XML模式文档，是在wmtsGetCapabilities_response.xsd文件中包含进来的。所有XML模式文档都包含每个元素、属性和类型的含义的说明，而且这说明应视同OWS通用实现规范[OGC 06-121r3]的11.6.3中规定的规范性要求。

附录B包含该规范性XML模式文档集的更多细节。

7.1.1.3 ServiceMetadata document example

A WMTS server might generate a ServiceMetadata document that looks like the following example. Another example can be found in Annex D:

```
<?xml version="1.0" encoding="UTF-8"?>
<Capabilities xmlns="http://www.opengis.net/wmts/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.opengis.net/wmts/1.0 ../wmtsGetCapabilities_res
ponse.xsd" version="1.0.0">
 <ows:ServiceIdentification>
  <ows:Title>World example Web Map Tile Service</ows:Title>
  <ows:Abstract>Example service that constrains some world layers in the
GlobalCRS84Pixel Well-known scale set</ows:Abstract>
  <ows:Keywords>
   <ows:Keyword>World</ows:Keyword>
   <ows:Keyword>Global</ows:Keyword>
   <ows:Keyword>Digital Elevation Model</ows:Keyword>
   <ows:Keyword>Administrative Boundaries</ows:Keyword>
  </ows:Keywords>
  <ows:ServiceType>OGC WMTS</ows:ServiceType>
  <ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
  <ows:Fees>none</ows:Fees>
  <ows:AccessConstraints>none</ows:AccessConstraints>
 </ows:ServiceIdentification>
 <ows:ServiceProvider>
  <ows:ProviderName>UAB-CREAF-MiraMon</ows:ProviderName>
  <ows:ProviderSite xlink:href="http://www.creaf.uab.cat/miramon"/>
  <ows:ServiceContact>
   <ows:IndividualName>Joan Maso Pau</ows:IndividualName>
   <ows:PositionName>Senior Software Engineer</ows:PositionName>
   <ows:ContactInfo>
    <ows:Phone>
     <ows:Voice>+34 93 581 1312</ows:Voice>
     <ows:Facsimile>+34 93 581 4151</ows:Facsimile>
    </ows:Phone>
    <ows:Address>
     <ows:DeliveryPoint>Fac Ciencies UAB</ows:DeliveryPoint>
     <ows:City>Bellaterra</ows:City>
     <ows:AdministrativeArea>Barcelona</ows:AdministrativeArea>
     <ows:PostalCode>08193</ows:PostalCode>
     <ows:Country>Spain</ows:Country>
     <ows:ElectronicMailAddress>joan.maso@uab.cat</ows:ElectronicMailAddress>
    </ows:Address>
   </ows:ContactInfo>
  </ows:ServiceContact>
 </ows:ServiceProvider>
 <ows:OperationsMetadata>
  <ows:Operation name="GetCapabilities">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/maps.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
```

7.1.1.3服务元数据文档实例

WMTS服务器会生成一个ServiceMetadata文档，类似以下实例。附录D给出了另一个实例。

```
<?xml version="1.0" encoding="UTF-8"?>
<Capabilities xmlns="http://www.opengis.net/wmts/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.opengis.net/wmts/1.0 ../wmtsGetCapabilities_res
ponse.xsd" version="1.0.0">
 <ows:ServiceIdentification>
  <ows:Title>World example Web Map Tile Service</ows:Title>
  <ows:Abstract>Example service that constrains some world layers in the
GlobalCRS84Pixel Well-known scale set</ows:Abstract>
  <ows:Keywords>
   <ows:Keyword>World</ows:Keyword>
   <ows:Keyword>Global</ows:Keyword>
   <ows:Keyword>Digital Elevation Model</ows:Keyword>
   <ows:Keyword>Administrative Boundaries</ows:Keyword>
  </ows:Keywords>
  <ows:ServiceType>OGC WMTS</ows:ServiceType>
  <ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
  <ows:Fees>none</ows:Fees>
  <ows:AccessConstraints>none</ows:AccessConstraints>
 </ows:ServiceIdentification>
 <ows:ServiceProvider>
  <ows:ProviderName>UAB-CREAF-MiraMon</ows:ProviderName>
  <ows:ProviderSite xlink:href="http://www.creaf.uab.cat/miramon"/>
  <ows:ServiceContact>
   <ows:IndividualName>Joan Maso Pau</ows:IndividualName>
   <ows:PositionName>Senior Software Engineer</ows:PositionName>
   <ows:ContactInfo>
    <ows:Phone>
     <ows:Voice>+34 93 581 1312</ows:Voice>
     <ows:Facsimile>+34 93 581 4151</ows:Facsimile>
    </ows:Phone>
    <ows:Address>
     <ows:DeliveryPoint>Fac Ciencies UAB</ows:DeliveryPoint>
     <ows:City>Bellaterra</ows:City>
     <ows:AdministrativeArea>Barcelona</ows:AdministrativeArea>
     <ows:PostalCode>08193</ows:PostalCode>
     <ows:Country>Spain</ows:Country>
     <ows:ElectronicMailAddress>joan.maso@uab.cat</ows:ElectronicMailAddress>
    </ows:Address>
   </ows:ContactInfo>
  </ows:ServiceContact>
 </ows:ServiceProvider>
 <ows:OperationsMetadata>
  <ows:Operation name="GetCapabilities">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/maps.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
```

```
      </ows:Constraint>
     </ows:Get>
     <ows:Post xlink:href="http://www.maps.bob/maps.cgi?">
      <ows:Constraint name="PostEncoding">
       <ows:AllowedValues>
        <ows:Value>SOAP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Post>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
  <ows:Operation name="GetTile">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/maps.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Get>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
  <ows:Operation name="GetFeatureInfo">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/maps.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Get>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
 </ows:OperationsMetadata>
 <Contents>
  <Layer>
   <ows:Title>etopo2</ows:Title>
   <ows:Abstract>ETOPO2 - 2 minute Worldwide Bathymetry/Topography
Data taken from National Geophysical Data Center(NGDC),
ETOPO2 Global 2' Elevations, September 2001...</ows:Abstract>
   <ows:WGS84BoundingBox>
    <ows:LowerCorner>-180 -90</ows:LowerCorner>
    <ows:UpperCorner>180 90</ows:UpperCorner>
   </ows:WGS84BoundingBox>
   <ows:Identifier>etopo2</ows:Identifier>
   <ows:Metadata xlink:href="http://www.maps.bob/etopo2/ metadata.htm"/>
   <Style isDefault="true">
    <ows:Title>default</ows:Title>
    <ows:Identifier>default</ows:Identifier>
    <LegendURL format="image/png"
xlink:href="http://www.maps.bob/etopo2/legend.png"/>
   </Style>
```

```
      </ows:Constraint>
     </ows:Get>
     <ows:Post xlink:href="http://www.maps.bob/maps.cgi?">
      <ows:Constraint name="PostEncoding">
       <ows:AllowedValues>
        <ows:Value>SOAP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Post>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
  <ows:Operation name="GetTile">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/maps.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Get>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
  <ows:Operation name="GetFeatureInfo">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/maps.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Get>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
 </ows:OperationsMetadata>
 <Contents>
  <Layer>
   <ows:Title>etopo2</ows:Title>
   <ows:Abstract>ETOPO2 - 2 minute Worldwide Bathymetry/Topography
Data taken from National Geophysical Data Center(NGDC),
ETOPO2 Global 2' Elevations, September 2001...</ows:Abstract>
   <ows:WGS84BoundingBox>
    <ows:LowerCorner>-180 -90</ows:LowerCorner>
    <ows:UpperCorner>180 90</ows:UpperCorner>
   </ows:WGS84BoundingBox>
   <ows:Identifier>etopo2</ows:Identifier>
   <ows:Metadata xlink:href="http://www.maps.bob/etopo2/ metadata.htm"/>
   <Style isDefault="true">
    <ows:Title>default</ows:Title>
    <ows:Identifier>default</ows:Identifier>
    <LegendURL format="image/png"
xlink:href="http://www.maps.bob/etopo2/legend.png"/>
   </Style>
```

```
   <Format>image/png</Format>
   <InfoFormat>application/gml+xml; version=3.1</InfoFormat>
   <TileMatrixSetLink>
    <TileMatrixSet>WholeWorld_CRS_84</TileMatrixSet>
   </TileMatrixSetLink>
  </Layer>
  <Layer>
   <ows:Title>Administrative Boundaries</ows:Title>
   <ows:Abstract>The sub Country Administrative Units 1998 GeoDataset represents
a small-scale political map of the world...</ows:Abstract>
   <ows:WGS84BoundingBox>
    <ows:LowerCorner>-180 -90</ows:LowerCorner>
    <ows:UpperCorner>180 84</ows:UpperCorner>
   </ows:WGS84BoundingBox>
   <ows:Identifier>AdminBoundaries</ows:Identifier>
   <ows:Metadata
xlink:href="http://www.maps.bob/AdminBoundaries/metadata.htm"/>
   <Style isDefault="true">
    <ows:Title>default</ows:Title>
    <ows:Identifier>default</ows:Identifier>
   </Style>
   <Format>image/png</Format>
   <TileMatrixSetLink>
    <TileMatrixSet>WholeWorld_CRS_84</TileMatrixSet>
   </TileMatrixSetLink>
  </Layer>
  <TileMatrixSet>
   <ows:Identifier>WholeWorld_CRS_84</ows:Identifier>
   <ows:SupportedCRS>urn:ogc:def:crs:OGC:1.3:CRS84</ows:SupportedCRS>
<WellKnownScaleSet>urn:ogc:def:wkss:OGC:1.0:GlobalCRS84Pixel</WellKnownScaleSe
t>
   <TileMatrix>
    <ows:Identifier>2g</ows:Identifier>
    <ScaleDenominator>795139219.9519541</ScaleDenominator>
    <!-- top left point of tile matrix bounding box -->
    <TopLeftCorner>-180 90</TopLeftCorner>
    <!-- width and height of each tile in pixel units -->
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
    <!-- width and height of matrix in tile units -->
    <MatrixWidth>1</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>1g</ows:Identifier>
    <ScaleDenominator>397569609.9759771</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
    <MatrixWidth>2</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>30m</ows:Identifier>
    <ScaleDenominator>198784804.9879885</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
```

```
   <Format>image/png</Format>
   <InfoFormat>application/gml+xml; version=3.1</InfoFormat>
   <TileMatrixSetLink>
    <TileMatrixSet>WholeWorld_CRS_84</TileMatrixSet>
   </TileMatrixSetLink>
  </Layer>
  <Layer>
   <ows:Title>Administrative Boundaries</ows:Title>
   <ows:Abstract>The sub Country Administrative Units 1998 GeoDataset represents
a small-scale political map of the world...</ows:Abstract>
   <ows:WGS84BoundingBox>
    <ows:LowerCorner>-180 -90</ows:LowerCorner>
    <ows:UpperCorner>180 84</ows:UpperCorner>
   </ows:WGS84BoundingBox>
   <ows:Identifier>AdminBoundaries</ows:Identifier>
   <ows:Metadata
xlink:href="http://www.maps.bob/AdminBoundaries/metadata.htm"/>
   <Style isDefault="true">
    <ows:Title>default</ows:Title>
    <ows:Identifier>default</ows:Identifier>
   </Style>
   <Format>image/png</Format>
   <TileMatrixSetLink>
    <TileMatrixSet>WholeWorld_CRS_84</TileMatrixSet>
   </TileMatrixSetLink>
  </Layer>
  <TileMatrixSet>
   <ows:Identifier>WholeWorld_CRS_84</ows:Identifier>
   <ows:SupportedCRS>urn:ogc:def:crs:OGC:1.3:CRS84</ows:SupportedCRS>
<WellKnownScaleSet>urn:ogc:def:wkss:OGC:1.0:GlobalCRS84Pixel</WellKnownScaleSe
t>
   <TileMatrix>
    <ows:Identifier>2g</ows:Identifier>
    <ScaleDenominator>795139219.9519541</ScaleDenominator>
    <!-- top left point of tile matrix bounding box -->
    <TopLeftCorner>-180 90</TopLeftCorner>
    <!-- width and height of each tile in pixel units -->
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
    <!-- width and height of matrix in tile units -->
    <MatrixWidth>1</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>1g</ows:Identifier>
    <ScaleDenominator>397569609.9759771</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
    <MatrixWidth>2</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>30m</ows:Identifier>
    <ScaleDenominator>198784804.9879885</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
```

```
   <MatrixWidth>3</MatrixWidth>
   <MatrixHeight>2</MatrixHeight>
  </TileMatrix>
  <TileMatrix>
   <ows:Identifier>20m</ows:Identifier>
   <ScaleDenominator>132523203.3253257</ScaleDenominator>
   <TopLeftCorner>-180 90</TopLeftCorner>
   <TileWidth>320</TileWidth>
   <TileHeight>200</TileHeight>
   <MatrixWidth>4</MatrixWidth>
   <MatrixHeight>3</MatrixHeight>
  </TileMatrix>
  <TileMatrix>
   <ows:Identifier>10m</ows:Identifier>
   <ScaleDenominator>66261601.66266284</ScaleDenominator>
   <TopLeftCorner>-180 90</TopLeftCorner>
   <TileWidth>320</TileWidth>
   <TileHeight>200</TileHeight>
   <MatrixWidth>7</MatrixWidth>
   <MatrixHeight>6</MatrixHeight>
  </TileMatrix>
  <TileMatrix>
   <ows:Identifier>5m</ows:Identifier>
   <ScaleDenominator>33130800.83133142</ScaleDenominator>
   <TopLeftCorner>-180 90</TopLeftCorner>
   <TileWidth>320</TileWidth>
   <TileHeight>200</TileHeight>
   <MatrixWidth>14</MatrixWidth>
   <MatrixHeight>11</MatrixHeight>
  </TileMatrix>
  <TileMatrix>
   <ows:Identifier>2m</ows:Identifier>
   <ScaleDenominator>13252320.33253257</ScaleDenominator>
   <TopLeftCorner>-180 84</TopLeftCorner>
   <TileWidth>320</TileWidth>
   <TileHeight>200</TileHeight>
   <MatrixWidth>34</MatrixWidth>
   <MatrixHeight>28</MatrixHeight>
  </TileMatrix>
 </TileMatrixSet>
</Contents>
<Themes>
 <Theme>
  <ows:Title>Foundation</ows:Title>
  <ows:Abstract>World reference data</ows:Abstract>
  <ows:Identifier>Foundation</ows:Identifier>
  <Theme>
   <ows:Title>Digital Elevation Model</ows:Title>
   <ows:Identifier>DEM</ows:Identifier>
   <LayerRef>etopo2</LayerRef>
  </Theme>
  <Theme>
   <ows:Title>Administrative Boundaries</ows:Title>
   <ows:Identifier>AdmBoundaries</ows:Identifier>
   <LayerRef>AdminBoundaries</LayerRef>
  </Theme>
 </Theme>
</Themes>
```

```
    <MatrixWidth>3</MatrixWidth>
    <MatrixHeight>2</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>20m</ows:Identifier>
    <ScaleDenominator>132523203.3253257</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
    <MatrixWidth>4</MatrixWidth>
    <MatrixHeight>3</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>10m</ows:Identifier>
    <ScaleDenominator>66261601.66266284</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
    <MatrixWidth>7</MatrixWidth>
    <MatrixHeight>6</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>5m</ows:Identifier>
    <ScaleDenominator>33130800.83133142</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
    <MatrixWidth>14</MatrixWidth>
    <MatrixHeight>11</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>2m</ows:Identifier>
    <ScaleDenominator>13252320.33253257</ScaleDenominator>
    <TopLeftCorner>-180 84</TopLeftCorner>
    <TileWidth>320</TileWidth>
    <TileHeight>200</TileHeight>
    <MatrixWidth>34</MatrixWidth>
    <MatrixHeight>28</MatrixHeight>
   </TileMatrix>
  </TileMatrixSet>
 </Contents>
 <Themes>
  <Theme>
   <ows:Title>Foundation</ows:Title>
   <ows:Abstract>World reference data</ows:Abstract>
   <ows:Identifier>Foundation</ows:Identifier>
   <Theme>
    <ows:Title>Digital Elevation Model</ows:Title>
    <ows:Identifier>DEM</ows:Identifier>
    <LayerRef>etopo2</LayerRef>
   </Theme>
   <Theme>
    <ows:Title>Administrative Boundaries</ows:Title>
    <ows:Identifier>AdmBoundaries</ows:Identifier>
    <LayerRef>AdminBoundaries</LayerRef>
   </Theme>
  </Theme>
 </Themes>
</Capabilities>
```

7.1.2 GetCapabilities operation (mandatory in procedure oriented architectural style)

The GetCapabilities operation in procedure oriented architectural style allows WMTS clients to retrieve a service metadata document from a server. The response to a GetCapabilities request SHALL be a document containing service metadata about the server, including specific information about the layers that can be requested, the tile sets in which these layers are available, and (optionally) one or more theme sets. The GetCapabilities operation also includes a version-negotiation mechanism, allowing the client and server to agree on a standard version on which to base all future communication. The subclause 7.1.1 specifies the sections for the ServiceMetadata document that a WMTS server SHALL return to describe its service metadata (usually encoded in an XML file).

7.1.2.1 GetCapabilities operation request

The GetCapabilities operation request SHALL be as specified in Subclauses 7.2 and 7.3 of OWS Common [OGC 06-121r3] and SHALL follow the Tables 3 and 7 of OWS Common [OGC 06-121r3]. Parameters of the GetCapabilities request are described in Table 3 of OWS Common [OGC 06-121r3] with the restriction in the value of the ―service‖ parameter that SHALL be ―WMTS‖ as listed in Table 17 below.

7.1.2 GetCapabilities操作（面向过程架构风格中为必选）

面向过程架构风格的GetCapabilities操作允许WMTS客户端从服务器获取一个服务元数据文档。对GetCapabilities请求的响应应为一个包含关于服务器的服务元数据文档，包括关于可以请求的图层、这些图层的图块集以及1或多个专题集的细节信息。GetCapabilities操作还包含一个版本协商机制，让客户端与服务器就标准版本达成一致并基于此进行之后的通信。7.1.1规定了为了说明其服务元数据，WMTS服务器应返回的服务元数据文档的各个子集（通常为XML编码的文件）。

7.1.2.1 GetCapabilities操作请求

GetCapabilities操作请求在OWS通用实现规范[OGC 06-121r3]的7.2和7.3中规定，并遵循OWS通用实现规范[OGC 06-121r3]表3和表7的要求。GetCapabilities请求的参数在OWS通用实现规范[OGC 06-121r3]的表3中说明，以此为基础，对service参数取值限定为‘WMTS’，如下面表17所示：

Table 17 — Parameters in GetCapabilities operation request

Names	Definition	Data type and values	Multiplicity and use
service `Service`	Service type identifier	Character String type, not empty SHALL be "WMTS"	One (mandatory)
request `Request`	Operation name	Character String type, not empty SHALL be "GetCapabilities"	One (mandatory)
acceptVersions `AcceptVersions`	Prioritized sequence of one or more standard versions accepted by client, with preferred versions listed first	Sequence of Character String type, each not empty Value is list of x.y.z ―version‖ values. SHALL contain "1.0.0"	Zero or one (optional) When omitted, return latest supported version
sections `Sections`	Unordered list of zero or more names of requested sections in complete service metadata document	Sequence of Character String type, each not empty Value is list of section names Allowed section names are in Table 18	Zero or one (optional) When omitted or not supported by server, return complete service metadata document
updateSequence `UpdateSequence`	Service metadata document version, value is ―increased‖ whenever any change is made in complete service metadata document	Character String type, not empty Values are selected by each server, and are always opaque to clients	Zero or one (optional) When omitted or not supported by server, return latest service metadata document
acceptFormats `AcceptFormats`	Prioritized sequence of zero or more response formats desired by client, with preferred formats listed first	Sequence of Character String type, each not empty Value is list of format identifiers Identifiers are MIME types of formats useful for service metadata documents	Zero or one (optional) When omitted or not supported by server, return service metadata document using MIME type "application/xml"

Sections name values of the WMTS service valid in a GetCapabilities request are specified in Table 6 of OWS Common [OGC 06-121r3] with the addition of the Themes section as listed in Table 18 below.

Table 18 — Meaning of section name values

Section name	Meaning
ServiceIdentification	Return ServiceIdentification element in service metadata document
ServiceProvider	Return ServiceProvider metadata element in service metadata document
OperationsMetadata	Return OperationsMetadata element in service metadata document
Contents	Return Contents metadata element in service metadata document
Themes	Return Themes metadata element in service metadata document
All	Return complete service metadata document, containing all elements

表17 —GetCapabilities操作请求的参数

名称	定义	数据类型与取值	出现次数与用法
service Service	服务类型标识符	字符串，非空 应取值为"WMTS"	1(必选)
request Request	操作名称	字符串，非空 应取值为 "GetCapabilities"	1(必选)
acceptVersions AcceptVersions	客户端可以接受的标准版本号序列，优先考虑的版本排在前面	字符串序列，非空 取值为x.y.z形式版本号的序列，应包含"1.0.0"	0或1 (可选) 如果省略，返回支持的最新版本
sections Sections	请求的0或多个完整服务元数据文档中的子集名称的无序列表	字符串序列，非空 取值为子集名称构成的列表 可用的子集名称在表18中给出	0或1 (可选) 如果省略或服务器不支持，返回整个服务元数据文档
updateSequence UpdateSequence	服务元数据文档的版本。整个服务元数据文档有任何变动时，该值应自增	字符串，非空 取值取决于各个服务器，对客户端始终是不透明的。	0或1 (可选) 如果省略或服务器不支持，返回最新的服务元数据文档
acceptFormats AcceptFormats	客户端可以接受的响应格式序列，优先考虑的格式排在前面	字符串序列，非空 取值为格式标识符构成的列表 标识符是服务元数据文档可用格式的MIME类型	0或1 (可选) 如果省略或服务器不支持，使用MIME类型"application/xml"返回服务元数据文档

GetCapabilities 请求中有效的WMTS服务子集名称取值在OWS通用实现规范[OGC 06-121r3]的表6中规定，另外再加上Themes子集，如下面表18所示。

表18 — 各子集名称取值含义

子集名称	含义
ServiceIdentification	返回服务元数据文档中的ServiceIdentification元素
ServiceProvider	返回服务元数据文档中的 ServiceProvider 元素
OperationsMetadata	返回服务元数据文档中的 OperationsMetadata 元素
Contents	返回服务元数据文档中的 Contents 元素
Themes	返回服务元数据文档中的 Themes 元素
All	返回完整的服务元数据文档，包含所有元素

The ―Multiplicity and use‖ column in Table 1 of OWS Common [OGC 06-121r3] specifies the optionally of each listed parameter in the GetCapabilities operation request. Table 19 specifies the implementation of those parameters by WMTS clients and servers.

Table 19 — Implementation of parameters in GetCapabilities operation request

Names	Multiplicity	Client implementation	Server implementation
service `Service`	One (mandatory)	Each parameter SHALL be implemented by all clients, using specified value	Each parameter SHALL be implemented by all servers, checking that each parameter is received with specified value
request `Request`	One (mandatory)		
acceptVersions `AcceptVersions`	Zero or one (optional)	SHOULD be implemented by all software clients, using specified values	SHALL be implemented by all servers, checking if parameter is received with specified value(s)
sections `Sections`	Zero or one (optional)	Each parameter may be implemented by each client If parameter not provided, SHALL expect default response	Each parameter may be implemented by each server If parameter not implemented or not received, SHALL provide default response
updateSequence `UpdateSequence`	Zero or one (optional)		
acceptFormats `AcceptFormats`	Zero or one (optional)	If parameter provided, SHALL allow default or specified response	If parameter implemented and received, SHALL provide specified response

7.1.2.2 GetCapabilites exceptions in procedure oriented architectural style

When a WMTS server encounters an error while performing a GetCapabilities operation, it SHALL return an exception-report message as specified in clause 8 of OWS Common [OGC 06-121r3]. The allowed exception codes SHALL include those listed in Table 20 assuming the updateSequence parameter is implemented by the server.

NOTE 1 To reduce the need for readers to refer to other documents, the table below is copied from Table 8 in subclause 7.4.1 of OWS Common [OGC 06-121r3].

Table 20 — Exception codes for GetCapabilities operation

exceptionCode value	Meaning of code	―locator‖ value
MissingParameterValue	Operation request does not include a parameter value	Name of missing parameter
InvalidParameterValue	Operation request contains an invalid parameter value	Name of parameter with invalid value
VersionNegotiationFailed	List of versions in ―AcceptVersions‖ parameter value, in GetCapabilities operation request, did not include any version supported by this server	None, omit ―locator‖ parameter
InvalidUpdateSequence	Value of (optional) updateSequence parameter, in GetCapabilities operation request, is greater than current value of service metadata updateSequence number	None, omit ―locator‖ parameter
NoApplicableCode	No other exceptionCode specified by this service and server applies to this exception	None, omit ―locator‖ parameter

OWS通用实现规范[OGC 06-121r3]的表1中出现次数与用法一列规定了GetCapabilities操作请求中每个列出参数是可选还是必选。表19规定WMTS客户端与服务器端对这些参数的实现方式。

表19 —GetCapabilities 操作请求中各参数的实现

名称	出现次数	客户端实现	服务器端实现
service Service	1(必选)	所有客户端应使用指定的值实现每个参数	所有服务器端应实现每个参数，并确保每个参数按指定值被接收。
request Request	1(必选)		
acceptVersions AcceptVersions	0或1 (可选)	所有客户端应使用指定的值实现	所有服务器端应实现，并确保参数按指定值被接收。
sections Sections	0或1 (可选)	每个客户端可以实现每个参数，如果没有提供参数，则表示期望获得默认响应	每个服务器可以实现每个参数，如果没有实现参数或未接收到，应提供默认响应
updateSequence UpdateSequence	0或1 (可选)		
acceptFormats AcceptFormats	0或1 (可选)	如果提供了参数，应允许默认或指定的响应	如果实现了参数并被接收，应提供指定的响应

7.1.2.2 面向过程架构风格下GetCapabilites操作中的异常情况

当WMTS服务器在进行GetCapabilities操作遇到错误时，应依照OWS通用实现规范[OGC 06-121r3]第8章的规定返回一个异常报告消息。表20列出了在假定服务器支持updateSequence参数的情况下允许使用的异常代码。

注1 为了读者免于参考其他文档的麻烦，下表为对OWS通用实现规范[OGC 06-121r3]的7.4.1一节中的表8的拷贝。

表20 —GetCapabilities操作的异常代码

exceptionCode取值	代码含义	locator的取值
MissingParameterValue	操作请求不含参数值	缺失参数的名称
InvalidParameterValue	操作请求所含参数值无效	含无效值的参数名称
VersionNegotiationFailed	操作请求中为AcceptVersions参数提供的版本列表中所有版本号该服务器均不支持	无，忽略locator参数
InvalidUpdateSequence	操作请求中updateSequence参数值比服务元数据中updateSequence的当前数值大	无，忽略 locator 参数
NoApplicableCode	该服务和服务器没有为该异常定义代码	无，忽略 locator 参数

If the client sends a KVP encoded request using unknown parameters these unknown

parameters SHALL be ignored by the server and will not cause an exception to be generated.

When a WMTS server responds with an ExceptionReport and the report is transmitted via HTTP, the WMTS server should set the status code of the HTTP response to the corresponding value for the given exceptionCode values, as shown in Table 21. When the ExceptionReport contains more than one Exception, then the HTTP status code value should be based upon the exceptionCode of the first Exception in the ExceptionReport.

Table 21 — HTTP exception codes and meanings on GetCapabilities operation

exceptionCode value	HTTP Status Code	
	Code	**Message**
MissingParameterValue	400	Bad request
InvalidParameterValue	400	Bad request
VersionNegotiationFailed	400	Bad request
InvalidUpdateSequence	400	Bad request
NoApplicableCode	500	Internal server error

7.1.3 ServiceMetadata resource request (mandatory in resource oriented architectural style)

WMTS servers using a resource oriented architectural style provide standard endpoints from which a representation of the ServiceMetadata resource can be obtained. The endpoint SHALL also be specified in the ServiceMetadata document although it will generally be obtained prior to communication with the server.

The client will request the representation of the ServiceMetadata resource by performing a standard request to the endpoint. In response to a correct request, the server SHALL provide a representation of its ServiceMetadata document which conforms with subclause 7.1.1. Incorrect requests shall be handled according to the standard semantics for errors for the transport protocol used for communication between the client and the server.

7.2 Tile

WMTS servers are designed to serve map image tiles. The ServiceMetadata document described in the previous subclause lists the tiles available on the server and the requirements for requesting a tile. Typically clients will first request the ServiceMetadata document from the server and then use the information in that document to discover how to perform valid requests for tiles.

7.2.1 Tile resource

The tile resource is generally a rectangular image containing cartographic data. Alternatively, this resource might be a non-image representation of the tile such as a description of the tile or a link to the actual image. For example, the tile resource could be a KML document used in a superoverlay, or a tile metadata document. When returning an image tile, a full single tile SHALL always be returned. Also, the background pixels of a tile SHOULD be transparent when possible so that the client can overlay the tiles on top of other map data (possibly other tiles).

The Tile resource representation SHALL be returned in the format specified in the request when the format has been advertised in the ServiceMetadata document as available for that Tile resource.

如果服务器端使用未知的参数发送一个KVP编码的请求，这些未知参数应被服务器忽略而不致产生异常。

当WMTS服务器响应一个ExceptionReport且该异常报告通过HTTP传输时，WMTS服务器应根据表21给定的exceptionCode值与状态码对应关系设定相应的HTTP响应状态代码。如果ExceptionReport中包含多个Exception，HTTP的状态值应根据ExceptionReport中第一个Exception的exceptionCode来设定。

表21 — 关于GetCapabilities 操作的HTTP异常代码与含义

exceptionCode 取值	HTTP状态码	
	代码	消息
MissingParameterValue	400	不良请求
InvalidParameterValue	400	不良请求
VersionNegotiationFailed	400	不良请求
InvalidUpdateSequence	400	不良请求
NoApplicableCode	500	服务器内部错误

7.1.3 ServiceMetadata资源请求(面向资源架构风格下为必选)

WMTS提供面向资源架构风格的标准访问端点，通过这些端点可以获得ServiceMetadata资源的表示结果。在ServiceMetadata文档中也规定了该端点，虽然通常需要在与服务器进行信息通讯之前就获得该端点。

客户端通过向该端点发出一个标准请求，即可请求与7.1.1一致的ServiceMetadata资源的表示结果。不正确的请求将按照客户端与服务器通讯所用传输协议针对错误的标准语义进行处理。

7.2 图块

WMTS服务器是设计用来提供图块服务的。前面说明的服务元数据文档列出了服务器上可用的图块以及请求图块的要求。典型情况是客户端先从服务器请求一个服务元数据文档，然后根据该文档中提供的信息确定如何发送获取图块的有效请求。

7.2.1 图块资源

图块资源通常是一个包含图形数据的矩形图片。或者，该资源也可以是图块的一个非图片形式的表达，如对图块的说明，或者是对实际图片的链接。例如，图块资源可以是应用程序中使用的KML文档，或者一个图块元数据文档。但返回一个图片块时，将始终返回一个完整的单个块。同时，图块的背景像素尽可能为透明，这样客户端可以把这些图块叠加到其他地图数据（可能是其他图块）之上。

如果在ServiceMetadata文档中已经申明了该图块资源所支持的格式，图块资源的表达应按照请求中据此指定的格式返回。

7.2.2 GetTile operation (mandatory in procedure oriented architectural style)

The GetTile operation in procedure oriented architectural style allows WMTS clients to request a particular tile of a particular tile matrix set in a predefined format. This operation has some parameters in common with WMS GetMap but it has been deliberately simplified. For instance, only one layer can be retrieved at a time. WMTS servers that want to allow a combination of layers to be served and requested have to give this combination an identifier and add it as a new layer in the service metadata document. Nevertheless, clients are expected to be able easily to overlay layers themselves eliminating the need for servers to offer layers by combination.

7.2.2.1 GetTile operation request

A request to perform the GetTile operation SHALL use the data structure specified in Table 22. This table also specifies the UML model data type, source of values, and multiplicity of each listed parameter, plus the default server behavior when an optional parameter is not included in the operation request.

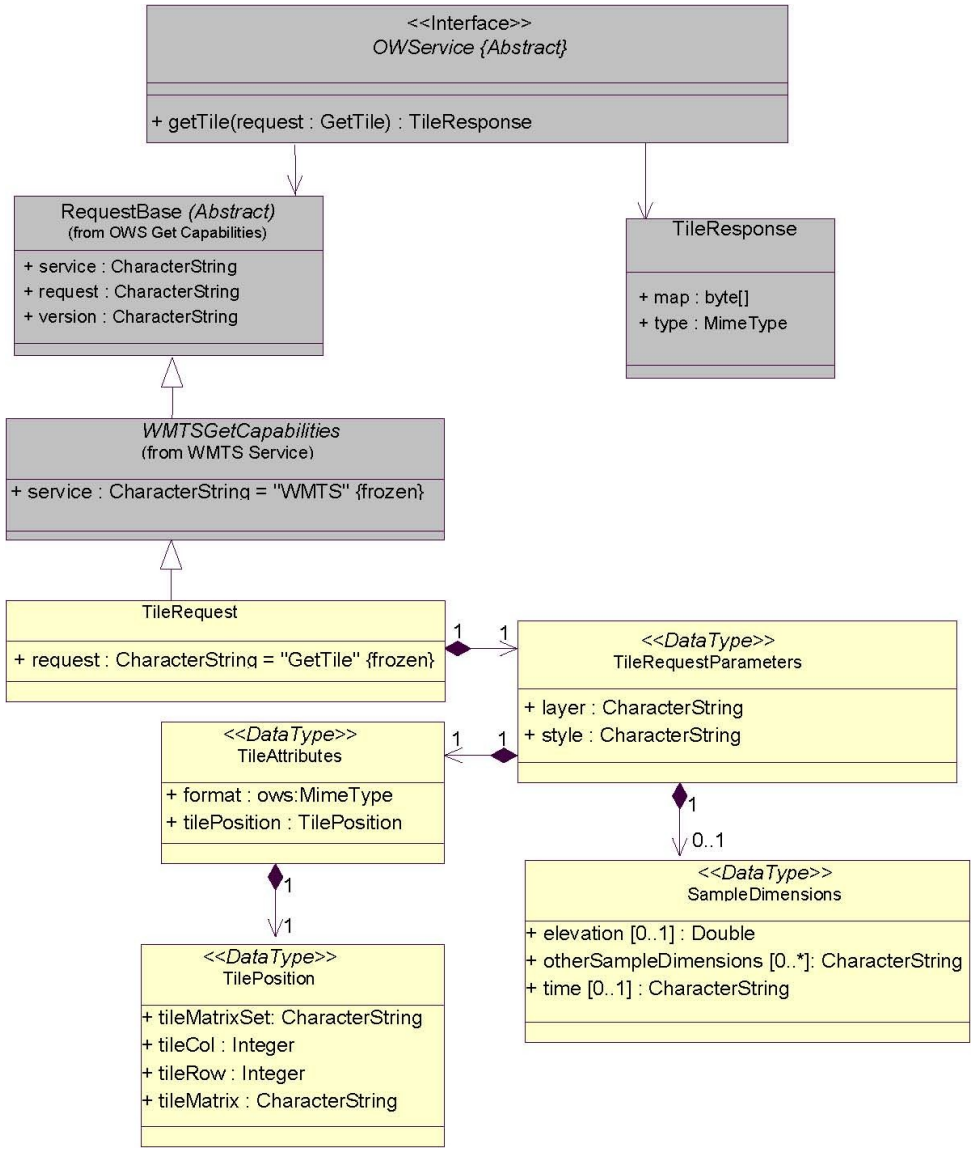

Figure 10 — GetTile operation request UML class diagram

7.2.2 GetTile 操作（面向过程架构风格下必选)

面向过程架构风格下的GetTile操作可以让WMTS客户端以某种预定义的格式请求某一块阵集中的特定图块。该操作与WMS GetMap具有一些相同的参数，但特意对该操作进行了简化。例如，每次只能请求一个图层。希望允许多个图层组合起来提供服务或接受请求的WMTS服务器，必须给这种图层组合一个标识符并作为一个新的图层添加到服务元数据文档中。但是，希望客户端可以很容易自己把各个图层叠加起来，以降低服务器上提供组合图层的必要性。

7.2.2.1 GetTile操作请求

发起GetTile操作的请求应使用表22的数据结构。该表也规定了UML模型数据类型、取值来源以及每个参数的出现次数和当某个可选参数未包含在操作请求中时默认的服务器响应行为。

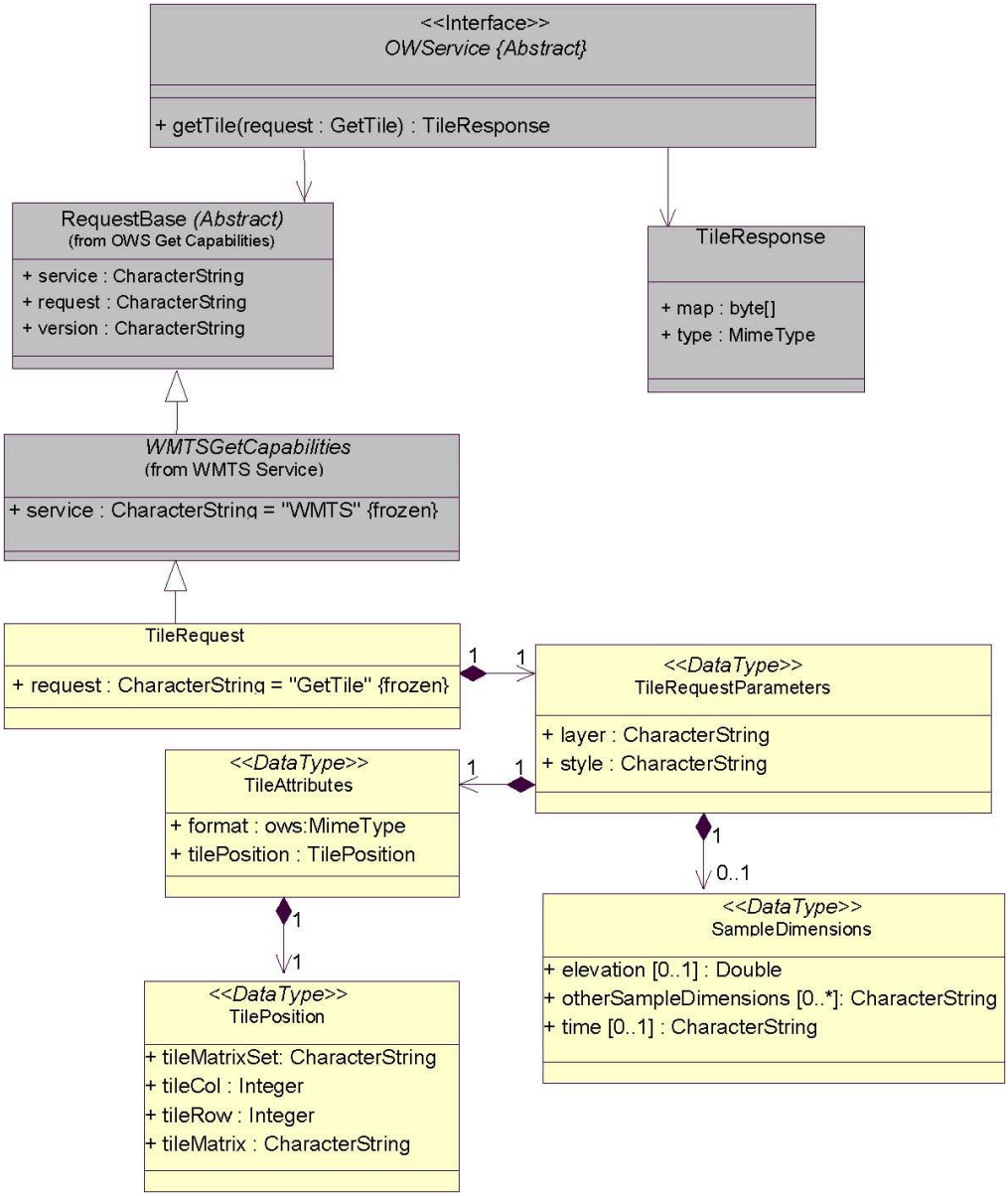

图10 — GetTile操作请求的UNL类图

Table 22 — Parameters in GetTile operation request

Names	Definition	Data type and values	Multiplicity and use
service `Service`	Service type identifier	Character String type, not empty SHALL be "WMTS"	One (mandatory)
request `Request`	Operation name	Character String type, not empty SHALL be "GetTile"	One (mandatory)
version `Version`	Standard version for operation	Character String type, not empty SHALL contain "1.0.0"	One (mandatory)
layer `Layer`	Layer identifier	Character String type, not empty identifier that is defined in the ServiceMetadata document	One (mandatory)
style `Style`	Style identifier	Character String type, not empty identifier that is defined in the ServiceMetadatadocument	One (mandatory)
format `Format`	Output format of the tile	Character String type, not empty value that is defined in the ServiceMetadata document	One (mandatory)
Other sample dimensions a	Value allowed for this dimension	Character String type, not empty a single value from a list or a range defined in the ServiceMetadata document	Zero or one (optional)
tileMatrixSet `TileMatrixSet`	TileMatrixSet identifier	Character String type, not empty identifier that is defined in the ServiceMetadata document	One (mandatory)
tileMatrix `TileMatrix`	TileMatrix identifier b	Character String type, not empty value that is defined in the ServiceMetadata document	One (mandatory)
tileRow `TileRow`	Row index of tile matrix	Non negative integer type value between 0 and MatrixHeight-1 of this tile matrix defined in the ServiceMetadata document	One (mandatory)
tileCol `TileCol`	Column index of tile matrix	Non negative integer type value between 0 and MatrixWidth-1 of this tile matrix defined in the ServiceMetadata document	One (mandatory)

a The name of this parameter is, in fact, the identifier of the dimension specified in the ServiceMetadata. See WMS 1.3.0 annex C for reference. This parameter appear once for each dimension specified for this Layer in the ServiceMetadata document.
b This will be the identifier of the tileMatrix of the desired scale denominator for the tileMatrixSet requested.

NOTE 1 To reduce the need for readers to refer to other documents, the first three parameters listed below are largely copied from Table 21 in subclause 9.2.1 of OWS Common [OGC 06-121r3].

NOTE 2 The UML class diagram in Figure 10 provides a useful graphical view of the contents of the GetTile operation request listed in Table 22.

表22 —GetTile操作请求的参数

名称	定义	数据类型与取值	出现次数与用法
service Service	服务类型标识符	字符串，非空，固定取值为"WMTS"	1(必选)
request Request	操作名	字符串，非空，固定取值为"GetTile"	1(必选)
version Version	操作的标准版本	字符串，非空，应包含"1.0.0"	1(必选)
layer Layer	图层标识符	字符串，非空。ServiceMetadata文档中定义的标识符	1(必选)
style Style	图式标识符	字符串，非空。ServiceMetadata文档中定义的标识符	1(必选)
format Format	图块输出格式	字符串，非空。ServiceMetadata文档中定义的取值	1(必选)
Other sample dimensions a	该维度允许的值	字符串，非空。ServiceMetadata文档中定义的值域或值列表中的一个值。	0或1(可选)
tileMatrixSet TileMatrixSet	TileMatrixSet的标识符	字符串，非空。ServiceMetadata文档中定义的标识符。	1(必选)
tileMatrix TileMatrix	TileMatrix的标识符b	字符串，非空。ServiceMetadata文档中定义的值。	1(必选)
tileRow TileRow	块阵的行号	非负整数。0到ServiceMetadata 中定义的该块阵的MatrixHeight-1之间的值	1(必选)
tileCol TileCol	块阵的列号	非负整数。0到ServiceMetadata 中定义的该块阵的MatrixWidth-1之间的值	1(必选)
a 该参数的名称实际上是ServiceMetadata中定义的维度的标识符，请参见WMS 1.3.0附录C。对于ServiceMetadata文档中该图层定义的每一个维度都需要一个该参数。 b 这将是请求块阵集中对应所要比例尺分母的块阵的标识符。			

注1 为读者免于查找其他文件的麻烦，前面3个参数基本从OWS通用实现规范[OGC 06-121r3]的9.2.1中的表21拷贝而来。

注2 图10中的UML类图提供了一个表22中列出的GetTile操作请求内容的图形视图。

7.2.2.2 GetTile exceptions in procedure oriented architectural style

When a WMTS server encounters an error while performing a GetTile operation, it SHALL return an exception report message as specified in subclause 7.4.1 of OWS Common [OGC 06-121r3]. The allowed standard exception codes SHALL include those listed in Table 23. For each listed exceptionCode, the contents of the ―locator‖ parameter value SHALL be as specified in the right column of Table 23.

NOTE 1 To reduce the need for readers to refer to other documents, four values listed below are copied from Table 8 in subclause 7.4.1 of OWS Common [OGC 06-121r3].

Table 23 — Exception codes for GetTile operation

exceptionCode value	Meaning of code	―locator‖ value
OperationNotSupported	Request is for an operation that is not supported by this server	Name of operation not supported
MissingParameterValue	Operation request does not include a parameter value, and this server did not declare a default value for that parameter	Name of missing parameter
InvalidParameterValue	Operation request contains an invalid parameter value	Name of parameter with invalid value
TileOutOfRange	TileRow or TileCol out of range	Name of the out-of-range parameter
NoApplicableCode	No other exceptionCode specified by this service and server applies to this exception	None, omit ―locator‖ parameter

If the client sends a GetTile request using unknown parameters (for example time, elevation or any other dimension that are not advertised in the ServiceMetadata document) these unknown parameters SHALL be ignored by the server and will not cause an exception to be generated.

When a WMTS server responds with an ExceptionReport and the report is transmitted via HTTP, the WMTS server should set the status code of the HTTP response to the corresponding value for the given exceptionCode values, as shown in Table 24. When the ExceptionReport contains more than one Exception then the HTTP status code value should be based upon the exceptionCode of the first Exception in the ExceptionReport.

Table 24 — HTTP exception codes and meanings on GetTile operation

exceptionCode value	HTTP Status Code	
	Code	Message
OperationNotSupported	501	Not implemented
MissingParameterValue	400	Bad request
InvalidParameterValue	400	Bad request
TileOutOfRange	400	Bad request
NoApplicableCode	500	Internal server error

7.2.2.2 面向过程架构风格下GetTile操作的异常

当WMTS服务器在执行GetTile操作出错时，应根据OWS通用实现规范[OGC 06-121r3]7.4.1的规定返回一个异常报告消息。可用的标准异常代码应包括在表23中的列出的代码。对于每一个exceptionCode，locator参数的值应指定为表23中右列中的值。

注1 为读者免于查找其他文件的麻烦，以下4个参数基本从OWS通用实现规范[OGC 06-121r3]的7.4.1中的表8拷贝而来。

表23 —GetTile 操作的异常代码

exceptionCode值	代码含义	“locator”取值
OperationNotSupported	请求的操作本服务器不支持	不支持的操作名称
MissingParameterValue	操作请求不包含参数值，服务器也没有为此参数申明默认值	缺失参数的名称
InvalidParameterValue	操作请求包含无效参数值	具有无效值的参数名称
TileOutOfRange	TileRow或TileCol取值不在值域范围内	越界参数的名称
NoApplicableCode	该服务或服务器没有为此异常定义异常代码	无，忽略“locator”参数

如果客户端使用未知的参数发送一个GetTile请求（如time，elevation或其他在ServiceMetadata文档中为申明法人维度），这些未知参数应被服务器忽略而且不致产生异常。

当WMTS服务器响应一个ExceptionReport且该异常报告通过HTTP传输，WMTS服务器应根据表24给定的exceptionCode值与状态码对应关系设定相应的HTTP响应状态代码。如果ExceptionReport中包含多个Exception，HTTP的状态值应根据ExceptionReport中第一个Exception的exceptionCode来设定。

表24 —GetTile操作的 HTTP异常代码及其含义

exceptionCode 取值	**HTTP状态代码**	
	代码	**消息**
OperationNotSupported	501	未实现
MissingParameterValue	400	不良请求
InvalidParameterValue	400	不良请求
TileOutOfRange	400	不良请求
NoApplicableCode	500	服务器内部错误

7.2.3 Tile resource request (mandatory in resource oriented architectural style)

WMTS servers using a resource oriented architectural style provide standard endpoints from which a representation of each Tile resource can be obtained. The endpoints SHALL be specified in the ServiceMetadata document using an address template.

The client will request the representation of any offered Tile resource by performing a request to the address following the standard semantics of the transport protocol used for communication between the client and the server. In response to a correct request, the server SHALL provide a representation of the Tile resource. Incorrect requests shall be handled according to the standard semantics for the transport protocol.

7.3 FeatureInfo

WMTS servers may support requests for information about the features present at a particular pixel location on a map tile. Requests for feature information will specify the tile along with a pixel location on that tile. The WMTS server will provide information on the features present at or near the location specified by the client request. The WMTS server may choose what information to provide about the nearby features.

7.3.1 FeatureInfo document

A FeatureInfo document is either the response of a GetFeatureInfo request in procedure oriented architectural style or the resource representation of a FeatureInfo resource in resource oriented architectural style. The FeatureInfo document SHALL be in the format specified in the request when that format has been advertised in the ServiceMetadata document as available for that FeatureInfo resource.

For a better interoperability between servers and clients we strongly recommend GML Simple Features Profile [06-049r1] as a supported document format for FeatureInfo resources. That standard defines three levels of content in three profiles with different degrees of constraints to the GML flexibility. We strongly recommend support of the most constrained one (level 0) that results in a simpler GML document. In the context of that profile only simple XML types can be used as thematic properties and cardinality greater than one is not allowed. Servers and clients SHALL specify the MIME type "application/gml+xml; version=3.1" as an InfoFormat value and the GML application schema of the response SHOULD conform to GML Simple Features profile level 0 when that GML profile is used. In most cases, only thematic attributes of the features are intended to be included in a FeatureInfo document but the Simple Feature profiles were evidently intended to include the geometric information of the features in the GML objects. However, it is possible to generate an application schema that does not include feature geometry and only describes non-geometric feature attribute types. This can be very useful to avoid unnecessarily requesting long sequences of position values in line or polygon layers.

Also, to allow easy presentation of the data, support for the HTML format (represented by an InfoFormat MIME type of ―text/html‖) is also recommended.

NOTE OGC 06-049r1 recommends the use of "text/xml; subType=gml/3.1.1/profiles/gmlsf/1.0.0/0" but this has been corrected by OGC 09-144r1 Technical Committee Policies and Procedures: MIME Media Types for GML. This document adopts the new policy.

7.2.3 图块资源请求 (面向资源架构风格下必选)

WMTS提供面向资源架构风格的标准访问端点，通过这些端点可以获得图块资源的表示结果。在ServiceMetadata文档应使用地址模板机制规定了该端点。

按照客户端与服务器间使用的通信协议的标准语义，客户端通过向该端点发出一个请求，即可请求服务器所提供的任何图块资源的表示结果。要响应一个正确的请求，服务器应提供图块资源的一个表达结果。不正确的请求将按照传输协议的标准语义进行处理。

7.3 FeatureInfo

WMTS服务器可以支持对地图图块上某一特定像素位置处要素信息的查询请求。请求要素信息时将通过图块上的一个像素位置来指明具体的图块。WMTS服务器将提供位于客户端请求中指定位置上或附近要素的信息，提供关于这些要素什么样的信息，由WMTS服务器决定。

7.3.1 FeatureInfo 文档

FeatureInfo文档要么是面向过程架构风格的GetFeatureInfo请求的响应结果，要么是面向资源架构风格下FeatureInfo资源的资源表达。FeatureInfo文档的格式应为客户端请求根据服务元数据文档中已申明该FeatureInfo资源可用格式而指定的格式。

为了支持服务器与客户端更好的互操作性，我们强烈建议把GML简单要素专用标准[06-049r1]作为FeatureInfo资源支持的文档格式。该标准在3个方案中通过对GML灵活性的不同程度限定定义了3个级别的内容。我们强烈建议支持约束最甚的方案（第0级），其结果是一个更加简单的GML文档。其中，只有能使用简单XML类型作为专题属性，而且不允许基数大于1。服务器和客户端应把InfoFormat参数值的MIME类型指定为"application/gml+xml; version=3.1"，而且如果使用GML专用标准，响应的GML应用模式应与GML简单要素专用标准第0级一致。大部分情况下，只有要素的专题属性需要含在FeatureInfo文档中，但是简单要素专用标准则明确倾向于把要素的图形信息包含到GML对象中。但是，也可以生成一个不含要素的图形信息而只说明非图形的要素属性类型的应用模式。这对于避免对线或多边形图层中长串位置坐标不必要的请求非常有用。

同时，为了可以轻松实现对数据的表达，建议支持HTML格式（InfoFormat的值设为MIME类型的text/html）。

注 OGC 06-049r1建议使用"text/xml; subType=gml/3.1.1/profiles/gmlsf/1.0.0/0"，但这已经被*OGC 09-144r1技术委员会政策与程序：GML的MIME媒体类型*所纠正，本文档采用该新规定。

7.3.2 GetFeatureInfo operation (optional in procedure oriented architectural style)

The GetFeatureInfo operation in procedure oriented architectural style allows WMTS clients to request information at a particular position of a particular tile for a particular queryable layer. A layer is queryable if the Contents section of the ServiceMetadata document specifies one or more InfoFormats for this layer.

NOTE 1 This criterion is different from the one used in WMS. In WMTS, the queryable property of WMS has been substituted by the presence or absence of an InfoFormat element in the ServiceMetadata document.

The GetFeatureInfo operation is designed to provide clients of a WMTS with more information about features rendered in a previously returned tile. The canonical use case for GetFeatureInfo is that a user chooses a pixel (I,J) on a particular tile at which the user would like to obtain more information. Because the WMTS protocol is stateless, the GetFeatureInfo request indicates to the WMTS server what tile the user is viewing by including the original GetTile request parameters but modifying the request value to 'GetFeatureInfo' and adding the pixel offset parameters. From the spatial context information (TileRow, TileCol and TileMatrixSet), along with the I,J position the user requested, the WMTS can return additional information about that position. The other GetTile parameters (*e.g.*, Style) may play a role in the server's decision as to what information to return.

NOTE 2When the user chooses a point (I,J) in a client that is showing overlapped layers, the client will need to make a separate GetFeatureInfo request for each layer.

7.3.2.1 GetFeatureInfo operation request

A request to perform the GetFeatureInfo operation SHALL include the use of the data parameters as specified in Figure 11 and in Table 25. This table also specifies the UML model data type, source of values, and multiplicity of each listed parameter, plus the meaning to servers when each optional parameter is not included in the operation request.

7.3.2 GetFeatureInfo操作(面向过程架构风格下为可选)

面向过程架构风格的GetFeatureInfo操作允许WMTS客户端针对某一特定可查询图层的特定图块上特定位置请求获取其信息。如果服务元数据文档中的Contents子集中针对一个图层指定了一或多个InfoFormats，则该图层是可查询的。

注1 这一标准与WMS中不同。WMTS中，WMS的可查询属性被服务元数据文档中InfoFormat元素是否出现替代。

GetFeatureInfo操作设计用于向WMTS客户端提供之前返回的图块中所含要素的更多信息。GetFeatureInfo的标准用例是用户在某个图块上希望获取更多信息的位置选择一个像素（I,J）。由于WMTS的协议是无状态的，GetFeatureInfo请求通过包含先前的GetTile请求参数向WMTS服务器表明哪个图块是用户正在查看的，但修改request的参数值为‘GetFeatureInfo’并加上以像素为单位的偏离值。基于空间上下文信息（TileRow, TileCol 以及TileMatrixSet），结合用户请求的位置（I,J），WMTS可以返回该位置的其他信息。其他GetTile参数（如Style）可以在服务器决定返回何种信息的过程中起某种作用。

注2 当用户在一个显示多个叠加图层的客户端中选择某个点（I,J）后，客户端需要为每个图层生成一个GetFeatureInfo请求。

7.3.2.1 GetFeatureInfo操作请求

GetFeatureInfo请求应使用图11和表25中给出的参数。该表也指定了UML模型数据类型、取值来源，每个参数的出现次数，以及操作请求中不包含每个可选参数时对服务器来说的具体含义。

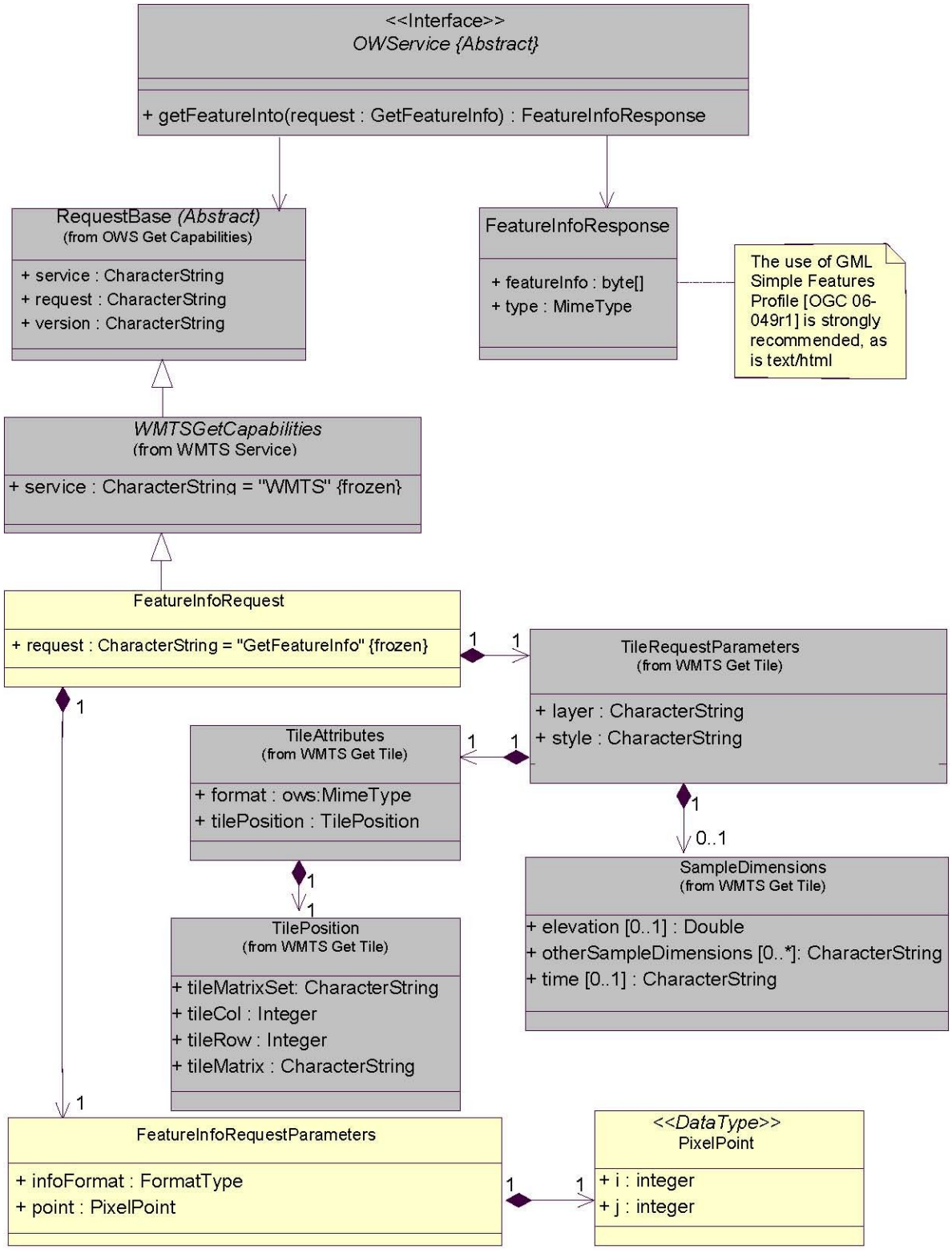

Figure 11 — GetFeatureInfo operation request UML class diagram

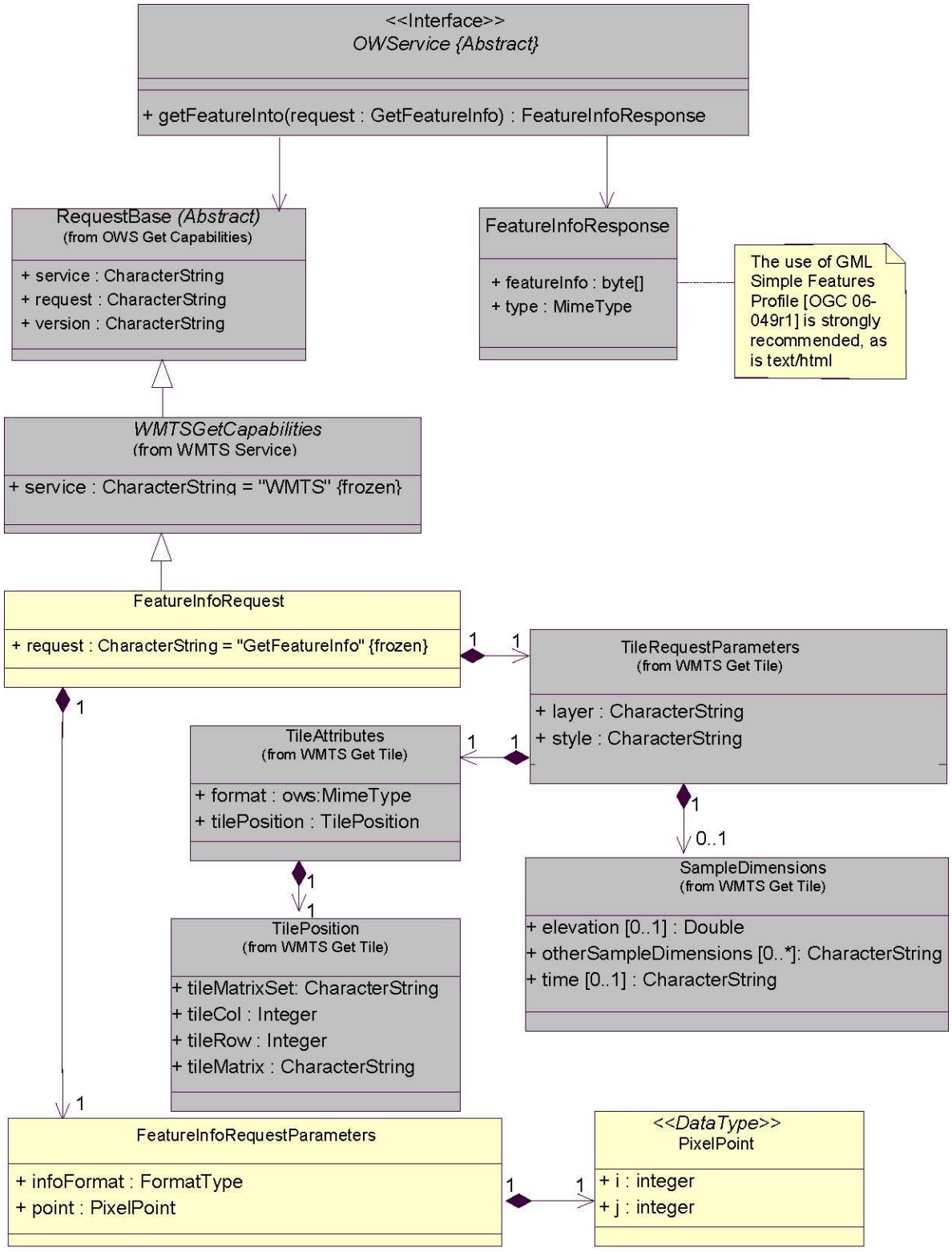

图11 — GetFeatureInfo操作请求的UML类图

Table 25 — Parameters in GetFeatureInfo operation request

Names	Definition	Data type and values	Multiplicity and use
service `Service`	Service type identifier	Character String type, not empty SHALL be ―WMTS‖	One (mandatory)
request `Request`	Operation name	Character String type, not empty SHALL be "GetFeatureInfo‖	One (mandatory)
version `Version`	Standard version for operation	Character String type, not empty SHALL contain 1.0.0	One (mandatory)
layer, style, format, *Sample dimension,* tileMatrixSet, tileMatrix, tileRow, tileCol	These correspond to the parameters of the same name in the corresponding GetTile request described in Table 22	The values of these parameters SHALL match those in the corresponding GetTile request described in Table 22	Multiplicity and use of these parameters SHALL match those in the corresponding GetTile request described in Table 22
j `J`	Row index of a pixel within the tile[a]	Non negative integer type value between 0 and TileHeight-1 of this tile matrix defined in the ServiceMetadata document	One (mandatory)
i `I`	Column index of a pixel within the tile[b]	Non negative integer type value between 0 and TileWidth-1 of this tile matrix defined in the ServiceMetadata document	One (mandatory)
infoFormat `InfoFormat`	Output format of the retrieved information	Character String type, not empty value that is defined in the ServiceMetadata document	One (mandatory)
a Number of full pixels in the Tile to the left of the requested location b Number of full pixels in the Tile to the top of the requested location			

NOTE 1 To reduce the need for readers to refer to other documents, the first three parameters listed are largely copied from Table 26 in subclause 9.2.1 of OWS Common [OGC 06-121r3].

NOTE 2 The UML class diagram in Figure 11 provides a useful graphical view of the contents of the GetFeatureInfo operation request listed in Table 25.

Although some values listed in the "Names" column appear to contain white spaces, they SHALL not contain any spaces.

7.3.2.2 GetFeatureInfo exceptions in procedure oriented architectural style

When a WMTS server encounters an error while performing a GetFeatureInfo operation, it SHALL return an exception report message as specified in subclause 7.4 of OWS Common [OGC 06-121r3]. The allowed standard exception codes SHALL include those listed in Table 26. For each listed exceptionCode, the contents of the ―locator‖ parameter value SHALL be as specified in the right column of Table 26.

NOTETo reduce the need for readers to refer to other documents, the first four values listed below are copied from Table 25 in subclause 8.3 of OWS Common [OGC 06-121r3].

表25 —GetFeatureInfo操作请求中的参数

名称	定义	数据类型与取值	出现次数与用法
service Service	服务类型标识符	字符串类型，非空 取值为“WMTS”	1(必选)
request Request	操作名称	字符串类型，非空 取值为"GetFeatureInfo“	1(必选)
version Version	操作的标准版本	字符串类型，非空 取值应包含1.0.0	1(必选)
layer, style, format, *Sample dimension,* tileMatrixSet, tileMatrix, tileRow, tileCol	这些参数与表22中GetTile请求的同名参数一致	与表22中GetTile的同名参数一样	与表22中GetTile的同名参数一
j J	图块中某个像素的行号[a]	非负数，为0到块阵TileHeight-1	1(必选)
i I	图块中某个像素的列号[b]	非负数，为0到块阵TileWidtht-1	1(必选)
infoFormat InfoFormat	请求信息的输出格式	非空字符串类型，可取的值在服务元数据文档中定义	1(必选)
a 请求位置到图块左侧的像素数 b 请求位置到图块顶部的像素数			

注1 为读者免于查找其他文件的麻烦，前面3个参数基本从OWS通用实现规范[OGC 06-121r3]的9.2.1中的表26拷贝而来。

注2 图11中的UML类图为表25中列出的GetFeatureInfo操作请求的参数提供了一个图形视图。

尽管在“名称”一列列出的一些名称包括空格，但实际上不应包含任何空格。

7.3.2.2 面向过程架构风格下GetFeatureInfo操作的异常

当WMTS服务器在执行GetFeatureInfo操作出错时，应根据OWS通用实现规范[OGC 06-121r3]7.4的规定返回一个异常报告消息。可用的标准异常代码应包括在表26中的列出的代码。对于每一个exceptionCode，locator参数的值应指定为表26中右列中的值。

注1 为读者免于查找其他文件的麻烦，下表前面4个参数基本从OWS通用实现规范[OGC 06-121r3]的8.3中的表25拷贝而来。

Table 26 — Exception codes for GetFeatureInfo operation

exceptionCode value	Meaning of code	―locator‖ value
OperationNotSupported	Request is for an operation that is not supported by this server or you are requesting a GetFeatureInfo response for a non queryable layer	Name of operation not supported
MissingParameterValue	Operation request does not include a parameter value, and this server did not declare a default value for that parameter	Name of missing parameter
InvalidParameterValue	Operation request contains an invalid parameter value	Name of parameter with invalid value
TileOutOfRange	TileRow or TileCol out of range	Name of the out-of-range parameter
PointIJOutOfRange	I or J out of range	Name of the out-of-range parameter
NoApplicableCode	No other exceptionCode specified by this service and server applies to this exception	None, omit ―locator‖ parameter

If the client sends a GetFeatureInfo request using unknown parameters (for example time, elevation or any other dimension that are not advertised in the ServiceMetadata document) these unknown parameters SHALL be ignored by the server and will not cause an exception to be generated.

When a WMTS server responds with an ExceptionReport and the report is transmitted via HTTP the WMTS server should set the HTTP response's status code to the corresponding value for the given exceptionCode values, as shown in Table 27. When the ExceptionReport contains more than one Exception then the HTTP status code value should be based upon the exceptionCode of the first Exception in the ExceptionReport.

Table 27 — HTTP exception codes and meanings on GetFeatureInfo operation

exceptionCode value	HTTP Status Code	
	Code	**Message**
OperationNotSupported	501	Not implemented
MissingParameterValue	400	Bad request
InvalidParameterValue	400	Bad request
TileOutOfRange	400	Bad request
PointIJOutOfRange	400	Bad request
NoApplicableCode	500	Internal server error

7.3.3 FeatureInfo resource request (optional in resource oriented architectural style)

WMTS servers using a resource oriented architectural style provide standard endpoints from which representations of the FeatureInfo resources can be obtained. The endpoints SHALL be specified in the ServiceMetadata document based on an address template.

表26 —GetFeatureInfo操作的异常代码

exceptionCode取值	代码含义	Locator取值
OperationNotSupported	请求的操作该服务器不支持，或者请求的图层不可查询	不支持操作的名称
MissingParameterValue	请求中没有包括必须的参数值，该服务器也没有为该参数设定默认值	缺失参数的名称
InvalidParameterValue	请求中包含无效参数值	含无效值的参数的名称
TileOutOfRange	TileRow或TileCol取值越界	越界参数的名称
PointIJOutOfRange	I 或 J 取值越界	越界参数的名称
NoApplicableCode	该服务和服务器没有定义针对该异常的异常代码	无，忽略Locator参数

如果客户端使用未知的参数发送一个GetFeatureInfo请求（如time，elevation或其他在ServiceMetadata文档中未申明其他维度），这些未知参数应被服务器忽略而且不致产生异常。

当WMTS服务器响应一个ExceptionReport且该异常报告通过HTTP传输，WMTS服务器应根据表27给定的exceptionCode值与状态码对应关系设定相应的HTTP响应状态代码。如果ExceptionReport中包含多个Exception，HTTP的状态值应根据ExceptionReport中第一个Exception的exceptionCode来设定。

表27 —关于GetFeatureInfo操作的HTTP异常代码与含义

exceptionCode 取值	**HTTP状态码**	
	代码	**消息**
OperationNotSupported	501	未实现
MissingParameterValue	400	不良请求
InvalidParameterValue	400	不良请求
TileOutOfRange	400	不良请求
PointIJOutOfRange	400	不良请求
NoApplicableCode	500	服务器内部错误

7.3.3 FeatureInfo资源请求(面向资源架构风格下可选)

WMTS提供面向资源架构风格的标准访问端点，通过这些端点可以获得FeatureInfo资源的表示结果。在ServiceMetadata文档应使用地址模板机制规定了该端点。

The client will request the representation of a FeatureInfo resource by performing a request to the address following the standard semantics of the transport protocol used for communication between the client and the server. In response to a correct request, the server SHALL provide a representation of each FeatureInfo resource. Incorrect requests shall be handled according to the standard semantics for the transport protocol.

7.4 Operation request encoding

The encoding of procedure oriented architectural style operation requests performed over HTTP SHALL use HTTP GET with KVP encoding or HTTP POST with KVP or SOAP encoding as specified in clause 11 of OWS Common [OGC 06-121r3]. Table 28 summarizes the WMTS Service operations and their encoding methods defined in this standard.

Table 28 — Procedure oriented architectural style operation request encoding

Operation name	Request encodings
GetCapabilities (required)	KVP, SOAP
GetTile (required)	KVP, SOAP
GetFeatureInfo	KVP, SOAP

HTTP GET with KVP is described in clause 8 and HTTP POST with SOAP is defined in described in clause 9.

A resource oriented architectural style using HTTP is also defined in this standard. For that approach, HTTP GET operations SHALL be used to access the resources (GetResourceRepresentation) available on the server. The resource repesentations SHALL be equivalent to those that are obtained as a result of the GetCapabilites, GetTile and GetFeatureInfo operations in the procedure oriented architectural style. This approach will be described in clause 10.

8 WMTS using HTTP KVP encoding

WMTS servers may support service requests made using lists of parameters and their values defined as lists of Key-Value Pairs (KVP) and sent over HTTP using either GET or POST messages. Each pair is defined using the name of the parameter followed by an equals sign, '=' or ASCII character 61 (decimal), followed by the value given to the parameter, for example "service=WMTS". Parameter names for the operations are defined in subclause 7.1.2.1. For GET HTTP messages, the KVP lists are sent as part of the URL, as in the example of subclause 08.1.1. In POST HTTP messages, the KVP lists are sent in the message body, one pair per line.

Any server wishing to support KVP requests SHALL declare its support by providing an OperationsMetadata section in its ServiceMetadata document with an Operation section for each supported operation and a section for each supported HTTP request type, GET or POST, in which KVP is declared as an AllowedValue, as explained in subclause 7.1.1.1.1. An example of this practice, declaring support for GetCapabilities operations using KVP with HTTP GET, can be seen in subclause 8.1.3.

按照客户端与服务器间使用的通信协议的标准语义，客户端通过向该端点发出一个请求，即可请求FeatureInfo资源的表示结果。要响应一个正确的请求，服务器应提供FeatureInfo资源的一个表达结果。不正确的请求将按照传输协议的标准语义进行处理。

7.4 操作请求编码

面向过程架构风格下基于HTTP的操作请求编码按照OWS通用实现规范[OGC 06-121r3]的规定使用KVP编码的HTTP GET或KVP/SOAP编码的HTTP POST方式。表28总结了本标准规定的WMTS服务的各个操作及其编码。

表28 —面向过程架构风格下操作请求的编码方法

操作名称	请求编码
GetCapabilities (required)	KVP, SOAP
GetTile (required)	KVP, SOAP
GetFeatureInfo	KVP, SOAP

KVP编码的HTTP Get方式在第8章说明，SOAP编码的HTTP POST方式在第9章说明。

使用HTTP的面向资源架构风格方式在本标准中也进行了定义。对于该方法，应使用HTTP Get操作来访问服务器上的资源。资源表达结果应与面向过程架构风格下GetCapabilites, GetTile和GetFeatureInfo操作获得的结果等同。该方法将在第10章说明。

8 使用HTTP KVP编码的WMTS

WMTS服务器可以支持使用定义为键值对列表形式的参数和值列表并通过HTTP GET或POST方式发送的服务请求。每一对定义为：参数名称加上等号“=”或ASCII字符61（十进制），再加上参数值，如"service=WMTS"。参数名称在7.1.2.1定义。对于GET方式的HTTP消息，KVP列表被作为URL的一部分发出，如8.1.1中所述。POST方式的HTTP消息，KVP列表作为消息主体发送，每行一对。

任何希望支持KVP请求的服务器应在服务元数据文档中的OperationsMetadata子集中对每一个支持的操作定义一个Operation子集，并为每一个支持的HTTP请求类型，即GET还是POST方式，提供一个子集，并根据7.1.1.1.1的要求说明支持情况。8.1.3中给出了一个申明GetCapabilities操作支持HTTP GET方式的KVP的实际例子，可供参考。

8.1 GetCapabilities

WMTS servers may support KVP requests for a representation of the ServiceMetadata document by declaring support for and correctly handling GetCapabilities requests.

8.1.1 GetCapabilities request HTTP KVP encoding

A client performs a GetCapabilities operation using KVP over HTTP by sending a GET or POST HTTP message with the 'request' parameter set to 'GetCapabilities'.

8.1.2 GetCapabilities request HTTP KVP encoding example

To request a WMTS ServiceMetadata document, a client could issue the following KVP-encoded GetCapabilities operation request with minimal contents:

```
http://www.maps.bob/maps.cgi?service=WMTS&version=1.0.0&request=GetCapabilitie
s
```

8.1.3 GetCapabilities HTTP KVP encoding response

In response to a valid GetCapabilities request from a client, a WMTS server SHALL send a ServiceMetadata document which conforms with the XML schema defined in Annex B.

8.1.4 GetCapabilities HTTP KVP encoding response example

In response to a valid GetCapabilities operation request in KVP encoding, a WMTS server might generate a document that looks like the one in subclause 7.1.1.3.

The following fragment declares support for KVP encoded GetCapabilities operations using HTTP GET:

```
...
 <ows:OperationsMetadata>
<ows:Operation name="GetCapabilities">
 <ows:DCP>
<ows:HTTP>
 <ows:Get xlink:href="http://www.maps.bob/maps.cgi?">
<ows:Constraint name="GetEncoding">
 <ows:AllowedValues>
<ows:Value>KVP</ows:Value>
 </ows:AllowedValues>
</ows:Constraint>
 </ows:Get>
</ows:HTTP>
 </ows:DCP>
</ows:Operation>
...
```

8.2 GetTile

WMTS servers may support KVP requests for representations of image Tiles by declaring support for and correctly handling GetTile requests.

8.2.1 GetTile request HTTP KVP encoding

Servers may implement HTTP GET transfer of the GetTile operation request, using KVP encoding. The KVP encoding of the GetTile operation request SHALL use the parameters specified in Table 29 which follows the abstract description specified in Table 22 above.

8.1 GetCapabilities

WMTS服务器可以通过声明支持并正确处理GetCapabilities请求的方式，表明对获取服务元数据文档表示结果的KVP请求的支持。

8.1.1 GetCapabilities请求的HTTP KVP 编码

客户端通过发送'request'参数取值为'GetCapabilities'的GET或POST HTTP消息，使用基于HTTP的KVP发起GetCapabilities操作。

8.1.2 GetCapabilities请求的HTTP KVP编码实例

要请求WMTS的服务元数据文档，客户端可以发出包含以下最少内容的KVP编码的GetCapabilities操作请求：

```
http://www.maps.bob/maps.cgi?service=WMTS&version=1.0.0&request=GetCapabilities
```

8.1.3 GetCapabilities HTTP KVP编码响应

WMTS服务器在响应客户端一个有效的GetCapabilities请求时，应发送一个与附录B中定义的XML模式相一致的服务元数据文档。

8.1.4 GetCapabilities HTTP KVP编码响应实例

WMTS服务器在响应一个KVP编码的有效GetCapabilities请求时，应生成一个形似7.1.1.3中那样的一个文档。

以下片段申明对KVP编码的使用HTTP GET方式的GetCapabilities操作的支持：

```
 ...
 <ows:OperationsMetadata>
<ows:Operation name="GetCapabilities">
 <ows:DCP>
<ows:HTTP>
 <ows:Get xlink:href="http://www.maps.bob/maps.cgi?">
<ows:Constraint name="GetEncoding">
 <ows:AllowedValues>
<ows:Value>KVP</ows:Value>
 </ows:AllowedValues>
</ows:Constraint>
 </ows:Get>
</ows:HTTP>
 </ows:DCP>
</ows:Operation>
...
```

8.2 GetTile

WMTS服务器可以通过声明支持并正确处理GetTile请求的方式，表明对获取图块表示结果的KVP请求的支持。

8.2.1 GetTile请求的HTTP KVP编码

服务器可以使用KVP编码实现GetTile操作请的HTTP GET传输。GetTile操作请求的KVP编码应使用根据前面表22中的抽象说明要求列出的表29中的参数。

Table 29 — GetTile operation request URL parameters

Name and example a	Optionality and use	Definition and format
`Service=WMTS`	Mandatory	Service type identifier
`Request=GetTile`	Mandatory	Operation name
`Version=1.0.0`	Mandatory	Standard and schema version for this operation
`Layer`	Mandatory	Layer identifier
`Style`	Mandatory	Style identifier
`Format`	Mandatory	Output format of tile
Sample dimensions b	Optional	Value allowed for this dimension
`TileMatrixSet`	Mandatory	TileMatrixSet identifier
`TileMatrix`	Mandatory	TileMatrix identifier
`TileRow`	Mandatory	Row index of tile matrix
`TileCol`	Mandatory	Column index of tile matrix
a All parameter names are here listed using mostly lower case letters. However, any parameter name capitalization SHALL be allowed in KVP encoding, see subclause 11.5.2 of OWS Common [OGC 06-121r3]. b Names for these parameters SHALL be the names indicated in the ServiceMetadata document. Typical examples are Time, Elevation and Band.		

Parameters in a GetTile request may be specified in any order. However, in order to facilitate the caching mechanisms already available on the web, the parameters SHOULD be specified in the exact order that appears in Table 29.

8.2.2 GetTile request HTTP KVP encoding example

An example GetTile operation request KVP encoded for HTTP GET is:

```
http://www.maps.bob/maps.cgi?service=WMTS&request=GetTile&version=1.0.0&layer=etopo2&style=default&format=image/png&TileMatrixSet=WholeWorld_CRS_84&TileMatrix=10m&TileRow=1&TileCol=3
```

8.2.3 GetTile HTTP KVP encoding response

The normal response to a valid GetTile operation request SHALL be a tile map that complies with the requested parameters and as described in subclause 7.2.1.

8.2.4 GetTile HTTP KVP encoding response example

A GetTile operation response for the GetTile request example in subclause 8.2.1 that corresponds with the ServiceMetadata document shown in subclause 7.1.1.3 is shown on Figure 12.

表29 — GetTile操作请求的URL 参数

名称与实例a	用法	定义与格式
Service=WMTS	必选	服务类型标识符
Request=GetTile	必选	操作名称
Version=1.0.0	必选	该操作的标准与模式版本
Layer	必选	图层标识符
Style	必选	图式标识符
Format	必选	图块输出格式
Sample dimensions b	可选	该文维度允许的取值
TileMatrixSet	必选	TileMatrixSet标识符
TileMatrix	必选	TileMatrix标识符
TileRow	必选	块阵的行号
TileCol	必选	块阵的列号
a 这里**列出的所有**参数大部分时间都使用小写，但是KVP编码中任何参数名大写方式都是允许的，参见OWS通用实现规范[OGC 06-121r3]的11.5.2. b 这些参数的名称应是服务元数据文档中所包含的名称，典型实例包括Time, Elevation 和 Band.		

GetTile请求中的参数可以按任意次序指定。但是，为了顺应Web上已有的缓存机制，这些参数应严格按照表29中出现的顺序进行指定。

8.2.2 GetTile请求的HTTP KVP编码实例

下面是KVP编码的HTTP Get方式GetTile操作请求的一个实例：

```
http://www.maps.bob/maps.cgi?service=WMTS&request=GetTile&version=1.0.0&layer=etopo2&style=default&format=image/png&TileMatrixSet=WholeWorld_CRS_84&TileMatrix=10m&TileRow=1&TileCol=3
```

8.2.3 GetTile HTTP KVP编码的响应

对一个有效GetTile操作请求的正常响应应为一个符合7.2.1规定的请求参数的图块。

8.2.4 GetTile HTTP KVP编码响应实例

图12对8.2.1给出的与7.1.1.3中服务元数据文档一致的针对GetTile请求的操作响应实例进行了图示。

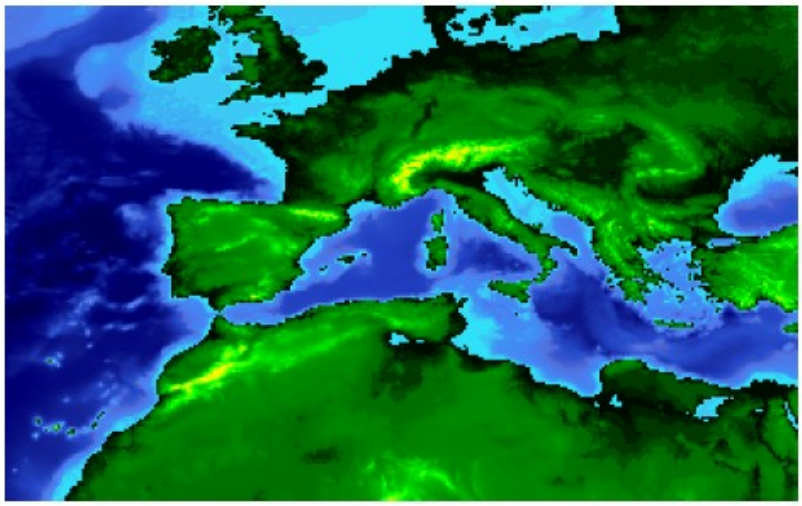

Figure 12 — GetTile response example

8.3 GetFeatureInfo

WMTS servers may support KVP requests for representations of documents furnishing information related to the features at particular pixel positions on particular image Tiles by declaring support for and correctly handling GetFeatureInfo requests.

8.3.1 GetFeatureInfo request HTTP KVP encoding

Servers may implement HTTP GET transfer of the GetFeatureInfo operation request, using KVP encoding. The KVP encoding of the GetFeatureInfo operation request SHALL follow the requirements for operation parameters specified in Table 30 that follows the abstract description specified in Table 25 above.

Table 30 — GetFeatureInfo operation request URL parameters

Name and example a	Optionality and use	Definition and format
`Service=WMTS`	Mandatory	Service type identifier
`Request=GetFeatureInfo`	Mandatory	Operation name
`Version=1.0.0`	Mandatory	Standard and schema version for this operation
Sample dimensions b	Optional	Value allowed for this dimension
`Layer, Style, Format, Sample dimensions, TileMatrixSet, TileMatrix, TileRow, and TileCol`	Optionality and use of these parameters SHALL match those in the corresponding GetTile request described in Table 29	The values of these parameters SHALL match those in the corresponding GetTile request described in Table 29
`J`	Mandatory	Row index of a pixel in the tile
`I`	Mandatory	Column index of a pixel in the tile
`InfoFormat`	Mandatory	Output format of the retrieved information
a All parameter names are here listed using mostly lower case letters. However, any parameter name capitalization SHALL be allowed in KVP encoding, see subclause 11.5.2 of OWS Common [OGC 06-121r3]. b Names for these parameters SHALL be the names indicated in the ServiceMetadata document. Typical examples are Time, Elevation and Band.		

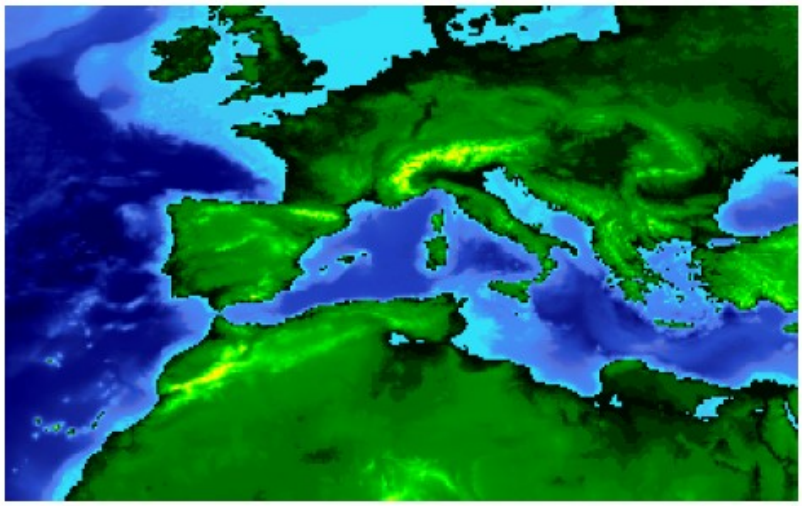

图12 — GetTile响应实例

8.3 GetFeatureInfo

WMTS服务器可以通过申明支持并正确处理GetFeatureInfo请求的方式，表明对特定图块特定像素位置的要素相关信息的KVP请求的支持。

8.3.1 HTTP KVP编码的GetFeatureInfo请求

服务器可以使用HTTP Get传输方式实现GetFeatureInfo操作请求。GetFeatureInfo操作请求的KVP编码应遵循根据前面表25的抽象描述而规定的表30中的操作参数的要求。

表 30 — GetFeatureInfo操作请求的URL参数

名称和实例 a	出现次数与用法	定义与格式
`Service=WMTS`	必选	服务类型标识符
`Request=GetFeatureInfo`	必选	操作名称
`Version=1.0.0`	必选	针对该操作的标准于模式的版本
Sample dimensions b	可选	该维度可用的取值
`Layer, Style, Format, Sample dimensions, TileMatrixSet, TileMatrix, TileRow, and TileCol`	与表29中GetTile请求的对应参数相同	取值与表29中GetTile请求的对应参数相同
`J`	必选	图块中像素的行号
`I`	必选	图块中像素的列号
`InfoFormat`	必选	访问信息的输出格式

a 这里列出的所有参数大都使用小写，但是，任何大写参数名称在KVP编码中都是允许的，见OWS通用实现规范的11.5.2 [OGC 06-121r3].
b 这些参数的名称应该是服务元数据文档中所表示的名称，典型的例如Time，Elevation和Band。

Parameters in a GetFeatureInfo request may be specified in any order. However, in order to facilitate the caching mechanisms already available on the web, the parameters SHOULD be specified in the exact order that appears in Table 30

8.3.2 GetFeatureInfo request HTTP KVP encoding example

An example GetFeatureInfo operation request KVP encoded for HTTP GET is:

```
http://www.maps.bob/maps.cgi?service=WMTS&request=GetFeatureInfo&version=1.0.0&layer=coastlines&style=default&format=image/png&TileMatrixSet=WholeWorld_CRS_84&TileMatrix=10m&TileRow=1&TileCol=3&J=86&I=132&InfoFormat=application/gml+xml; version=3.1
```

8.3.3 GetFeatureInfo HTTP KVP encoding response

The normal response to a valid GetFeatureInfo operation request SHALL be a FeatureInfo document as described in subclause 7.3.1.

8.3.4 GetFeatureInfo HTTP KVP encoding response example

A GetFeatureInfo operation response for the GetFeatureInfo request example in subclause 8.3.1 may look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<ReguralGriddedElevations
xmlns="http://www.maps.bob/etopo2"xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.maps.bob/etopo2
GetFeatureInfoExampleSchema.xsd">
 <featureMember>
<GridPoint_etopo2>
 <elevation>503.0</elevation>
 <TileRow>1</TileRow>
 <TileCol>2</TileCol>
 <J>86</J>
 <I>132</I>
</GridPoint_etopo2>
 </featureMember>
</ReguralGriddedElevations>
```

8.4 Exceptions in HTTP KVP encoded operations

If an error is detected while processing an operation request encoded with KVP over HTTP, the WMTS server SHALL generate an ExceptionReport element (as defined in clause 8.5 of OWS Common [OGC 06-121r3]) and which conforms to the schemas described in Annex B.

9 WMTS using SOAP encoding

A WMTS server SHALL declare support for SOAP encoding for each operation by means of the OperationsMetadata section of its ServiceMetadata document as explained in subclause 7.1.1.1.1. An example of this practice for GetCapabilities operation can be seen in subclause 9.1.2. Annex F contains information on how to write a WSDL description document of the service.

9.1 GetCapabilities

9.1.1 GetCapabilities request SOAP encoding

Servers may also implement SOAP encoding using HTTP POST transfer of the GetCapabilities operation request, using SOAP version 1.2 encoding.

GetFeatureInfo请求中的参数可按任意顺序指定。但是，为了充分利用Web上已有的缓存机制，这些参素应严格按照在表30中出现的顺序进行指定。

8.3.2 HTTP KVP编码的GetFeatureInfo请求实例

针对HTTP GET的KVP编码的GetFeatureInfo操作请求实例如下：

```
http://www.maps.bob/maps.cgi?service=WMTS&request=GetFeatureInfo&version=1.0.0&layer=coastlines&style=default&format=image/png&TileMatrixSet=WholeWorld_CRS_84&TileMatrix=10m&TileRow=1&TileCol=3&J=86&I=132&InfoFormat=application/gml+xml; version=3.1
```

8.3.3 HTTP KVP 编码的GetFeatureInfo的响应

对一个有效GetFeatureInfo操作请求的正常响应应该是符合7.3.1的一个FeatureInfo文档。

8.3.4 HTTP KVP 编码的GetFeatureInfo的响应实例

一个针对8.3.2中的GetFeatureInfo请求实例的响应形似如下片段：

```
<?xml version="1.0" encoding="UTF-8"?>
<RegularGriddedElevations
xmlns="http://www.maps.bob/etopo2"xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.maps.bob/etopo2
GetFeatureInfoExampleSchema.xsd">
 <featureMember>
<GridPoint_etopo2>
 <elevation>503.0</elevation>
 <TileRow>1</TileRow>
 <TileCol>2</TileCol>
 <J>86</J>
 <I>132</I>
</GridPoint_etopo2>
 </featureMember>
</RegularGriddedElevations>
```

8.4 HTTP KVP编码操作的异常

如果在处理使用基于HTTP的KVP编码的操作请求时检测到错误，WMTS服务器应产生一个ExceptionReport元素（在OWS通用实现规范[OGC 06-121r3]的8.5中进行了规定），并要求符合附录B给出的模式。

9 WMTS using SOAP encoding 使用SOAP编码的WMTS

WMTS服务器应使用7.1.1.1.1描述的ServiceMetadata文档中OperationMetadata模块来声明对每个操作请求SOAP编码的支持。9.1.2节给出了GetCapabilities操作的SOAP编码实例。附件F含有WSDL描述文件写法的有关信息。

9.1 GetCapabilities

9.1.1 GetCapabilities请求的 SOAP编码

服务器也可采用SOAP 1.2的编码方法使用GetCapabilities操作请求的HTTP POST传输方式实现SOAP编码。

9.1.2 GetCapabilities request SOAP encoding example

To request a WMTS ServiceMetadata document, a client could issue the following SOAP encoded GetCapabilities operation request:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<GetCapabilities service="WMTS"
xmlns="http://www.opengis.net/ows/1.1">
<AcceptVersions>
<Version>1.0.0</Version>
</AcceptVersions>
<AcceptFormats>
<OutputFormat>application/xml</OutputFormat>
</AcceptFormats>
</GetCapabilities>
</soap:Body>
</soap:Envelope>
```

9.1.3 GetCapabilities SOAP encoding response

In response to GetCapabilities operation request in SOAP encoding, a WMTS server SHALL generate a document that looks like the one in subclause 7.1.1.3 wrapped in the SOAP version 1.2 envelope.

9.1.4 GetCapabilities SOAP encoding response example

The following fragment remarks the SOAP envelope and the encoding of GetCapabilities response in the OperationsMetadata section:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<Capabilities version="1.0.0"
xmlns="http://www.opengis.net/wmts/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:gml="http://www.opengis.net/gml">
...
<ows:OperationsMetadata>
<ows:Operation name="GetCapabilities">
<ows:DCP>
<ows:HTTP>
<ows:Post
xlink:href="http://www.maps.bob/maps.cgi?">
<ows:Constraint name="PostEncoding">
<ows:AllowedValues>
<ows:Value>SOAP</ows:Value>
</ows:AllowedValues>
</ows:Constraint>
</ows:Post>
</ows:HTTP>
</ows:DCP>
</ows:Operation>
...
<WSDL xlink:role="http://schemas.xmlsoap.org/wsdl/1.0"
xlink:show="none" xlink:type="simple"
xlink:href="wmtsConcrete.wsdl"/>
</Capabilities>
</soap:Body>
</soap:Envelope>
```

9.1.2 GetCapabilities请求SOAP编码实例

为请求WMTS ServiceMetadata文档，客户端可发出如下所示的SOAP编码的GetCapabilities请求：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<GetCapabilities service="WMTS"
xmlns="http://www.opengis.net/ows/1.1">
<AcceptVersions>
<Version>1.0.0</Version>
</AcceptVersions>
<AcceptFormats>
<OutputFormat>application/xml</OutputFormat>
</AcceptFormats>
</GetCapabilities>
</soap:Body>
</soap:Envelope>
```

9.1.3 SOAP编码的GetCapabilities响应

要响应SOAP编码的GetCapabilities操作请求，WMTS服务器应生成一个形似7.1.1.3节里由SOAP 1.2信报封装的文档。

9.1.4 GetCapabilities响应SOAP编码示例

以下片段显示了OperationsMetadata子集中GetCapabilities响应的SOAP信报和编码：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<Capabilities version="1.0.0"
xmlns="http://www.opengis.net/wmts/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:gml="http://www.opengis.net/gml">
...
<ows:OperationsMetadata>
<ows:Operation name="GetCapabilities">
<ows:DCP>
<ows:HTTP>
<ows:Post
xlink:href="http://www.maps.bob/maps.cgi?">
<ows:Constraint name="PostEncoding">
<ows:AllowedValues>
<ows:Value>SOAP</ows:Value>
</ows:AllowedValues>
</ows:Constraint>
</ows:Post>
</ows:HTTP>
</ows:DCP>
</ows:Operation>
...
<WSDL xlink:role="http://schemas.xmlsoap.org/wsdl/1.0"
xlink:show="none" xlink:type="simple"
xlink:href="wmtsConcrete.wsdl"/>
</Capabilities>
</soap:Body>
</soap:Envelope>
```

9.2 GetTile

9.2.1 GetTile request SOAP encoding

Servers may also implement SOAP encoding using HTTP POST transfer of the GetTile operation request, using SOAP version 1.2 encoding.

9.2.2 GetTile request SOAP encoding example

An example of the SOAP encoding of the GetTile operation request equivalent of the request in subclause 8.2.1 is:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<GetTile service="WMTS" version="1.0.0"
xmlns="http://www.opengis.net/wmts/1.0">
<Layer>etopo2</Layer>
<Style>default</Style>
<Format>image/png</Format>
<TileMatrixSet> WholeWorld_CRS_84</TileMatrixSet>
<TileMatrix>10m</TileMatrix>
<TileRow>1</TileRow>
<TileCol>3</TileCol>
</GetTile>
</soap:Body>
</soap:Envelope>
```

9.2.3 GetTile SOAP encoding response

The response of a successful SOAP-encoded GetTile operation request SHALL be an image with the MIME type specified by the Format parameter of the request, wrapped in the SOAP version 1.2 envelope. If the image is binary (such as is the case with image/png and image/jpeg images), it SHALL be base64 encoded and placed within the following XML element:

```
<element name="BinaryPayload">

<complexType>
<sequence>
<element name="Format" type="ows:MimeType"/>
<element name="PayloadContent" type="base64Binary"/>
</sequence>
</complexType>
</element>
```

The xs:base64Binary type and associated base64-encoding rules are defined in the XML Schema Part 2 W3C recommendation. MIME element includes the MIME type of the original BinaryPayload.

NOTE 1The reason for using embedded encoded data instead of linking to an external source is to allow secured implementations.

Since the main part of the SOAP message is the base64 encoded binary content, base64 data SHOULD be enclosed inside a <![CDATA[ ]]>. This will prevent unnecessary parse of the base64 data resulting in a fast XML parse and validation.

NOTE 2Current JavaScript XML parsers have limits on the length of the element content that are often too low to contain a base64 256x256 image in a single element. It as been seen that the use of <![CDATA[ ]]> is a workaround for this limitation.

9.2 GetTile

9.2.1 GetTile请求SOAP编码

服务器也可采用SOAP 1.2的编码方法使用GetTile操作请求的HTTP POST传输方式实现SOAP编码。

9.2.2 GetTile请求SOAP编码示例

下面是与8.2.1节中请求对应的SOAP编码的GetTile操作请求示例：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<GetTile service="WMTS" version="1.0.0"
xmlns="http://www.opengis.net/wmts/1.0">
<Layer>etopo2</Layer>
<Style>default</Style>
<Format>image/png</Format>
<TileMatrixSet> WholeWorld_CRS_84</TileMatrixSet>
<TileMatrix>10m</TileMatrix>
<TileRow>1</TileRow>
<TileCol>3</TileCol>
</GetTile>
</soap:Body>
</soap:Envelope>
```

9.2.3 GetTile响应SOAP编码

对一个成功的SOAP编码的GetTile请求的响应应该是一个由请求中Format参数指定MIME类型的图像，并用SOAP 1.2信报进行了封装。如果图片是二进制（image/png和image/jpg）格式，应该采用base64编码，并且放在如下的XML元素里：

```
<element name="BinaryPayload">

<complexType>
<sequence>
<element name="Format" type="ows:MimeType"/>
<element name="PayloadContent" type="base64Binary"/>
</sequence>
</complexType>
</element>
```

xs:base64Binary类型和相关的base64编码规则由W3C推荐的XML模式第二部分定义。MIME元素包含原始BinaryPayload的MIME类型。

注1：使用嵌入式编码数据替代外部链接是为了可靠实现的需要。

因为SOAP消息的主体部分是base64编码的二进制内容，base64数据应当包含在<![CDATA[ ]]>内，这样可避免对base64数据的不必要解析，从而快速完成XML处理和验证。

注2：当前JavaScript XML处理器有内容长度的限制，通常很难在单一元素内包含一幅256x256的图像。可见<![CDATA[ ]]>的使用是应对该限制的一个办法。

9.2.4 GetTile SOAP encoding response example

An example of the SOAP response may look like:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<wmts:BinaryPayload>
<wmts:Format>image/png</wmts:Format>
<wmts:BinaryContent>
<!-- base64-encoded -->
<![CDATA[R0lGODdh4AEOAfYAAOGW/9aM9MuC6L943LRu
0KlkxJ1auJJQrIdGoXs8lXAyiWUofVkecU4U
<!-- ... -->
Ah0ianvIh+7Fb38oehcBI4NIiPdXhECyf4zY
iGNnOq7FcfZTiUJ1hfyjVCW3bJ3IiYEAADs=]]>
</wmts:BinaryContent>
</wmts:BinaryPayload>
</soap:Body>
</soap:Envelope>
```

9.3 GetFeatureInfo

9.3.1 GetFeatureInfo request SOAP encoding

Servers may also implement SOAP encoding using HTTP POST transfer of the GetFeatureInfo operation request, using SOAP version 1.2 encoding.

9.3.2 GetFeatureInfo request SOAP encoding example

An example of the SOAP encoding of the GetFeatureInfo operation request equivalent of the request in subclause 8.3.1 is:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<GetFeatureInfo service="WMTS" version="1.0.0"
xmlns="http://www.opengis.net/wmts/1.0">
<GetTile service="WMTS" version="1.0.0"
xmlns="http://www.opengis.net/wmts/1.0">
<Layer>etopo2</Layer>
<Style>default</Style>
<Format>image/png</Format>
<TileMatrixSet> WholeWorld_CRS_84</TileMatrixSet>
<TileMatrix>10m</TileMatrix>
<TileRow>1</TileRow>
<TileCol>3</TileCol>
</GetTile>
<J>86</J>
<I>132</I>
<InfoFormat>application/gml+xml; version=3.1</InfoFormat>
</GetFeatureInfo>
</soap:Body>
</soap:Envelope>
```

9.2.4 GetTile响应SOAP编码示例

SOAP响应示例如以下片段所示：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<wmts:BinaryPayload>
<wmts:Format>image/png</wmts:Format>
<wmts:BinaryContent>
<!-- base64-encoded -->
<![CDATA[R0lGODdh4AEOAfYAAOGW/9aM9MuC6L943LRu
0KlkxJ1auJJQrIdGoXs8lXAyiWUofVkecU4U
<!-- ... -->
Ah0ianvIh+7Fb38oehcBI4NIiPdXhECyf4zY
iGNnOq7FcfZTiUJ1hfyjVCW3bJ3IiYEAADs=]]>
</wmts:BinaryContent>
</wmts:BinaryPayload>
</soap:Body>
</soap:Envelope>
```

9.3 GetFeatureInfo

9.3.1 GetFeatureInfo请求的SOAP编码

服务器也可采用SOAP 1.2使用HTTP POST传输来实现GetFeatureInfo请求的SOAP编码。

9.3.2 GetFeatureInfo请求SOAP编码实例

与8.3.1节中的请求对应的GetFeatureInfo的SOAP编码示例如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
<soap:Body>
<GetFeatureInfo service="WMTS" version="1.0.0"
xmlns="http://www.opengis.net/wmts/1.0">
<GetTile service="WMTS" version="1.0.0"
xmlns="http://www.opengis.net/wmts/1.0">
<Layer>etopo2</Layer>
<Style>default</Style>
<Format>image/png</Format>
<TileMatrixSet> WholeWorld_CRS_84</TileMatrixSet>
<TileMatrix>10m</TileMatrix>
<TileRow>1</TileRow>
<TileCol>3</TileCol>
</GetTile>
<J>86</J>
<I>132</I>
<InfoFormat>application/gml+xml; version=3.1</InfoFormat>
</GetFeatureInfo>
</soap:Body>
</soap:Envelope>
```

9.3.3 GetFeatureInfo SOAP encoding response

The response of a successful SOAP-encoded GetFeatureInfo operation request SHALL be a document with the MIME type specified by the InfoFormat parameter of the request, wrapped in the SOAP version 1.2 envelope.

Since the GetFeatureInfo response format does not mandate any particular response format, the following flexible XML element that emphasizes the recommendation of GML Simple Features Profile level 0 response format is shown below:

```
<element name="FeatureInfoResponse">
<complexType>
<choice>
<element ref="gml:_FeatureCollection">
<element ref="wmts:TextPayload">
<element ref="wmts:BinaryPayload">
<element name="AnyContent" type="anyType">
</choice>
</complexType>
</element>
```

9.3.4 GetFeatureInfo SOAP encoding response example

A GetFeatureInfo operation SOAP response for the GetFeatureInfo SOAP request example in subclause 9.3.2 may look like this in GML encoding:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope
http://www.w3.org/2003/05/soap-envelope.xsd">
<soap:Body>
<FeatureInfoResponse>
<RegularGridedElevations
xmlns="http://www.maps.bob/etopo2"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.maps.bob/etopo2
wmtsGetFeatureInfo_response.xsd">
<featureMember>
<GridPoint_etopo2>
<elevation>503.0</elevation>
<TileRow>1</TileRow>
<TileCol>2</TileCol>
<J>86</J>
<I>132</I>
</GridPoint_etopo2>
</featureMember>
</RegularGridedElevations>
</FeatureInfoResponse>
</soap:Body>
</soap:Envelope>
```

A GetFeatureInfo operation SOAP response with the same information in HTML encoding may look like this:

9.3.3 SOAP编码的GetFeatureInfo响应

GetFeatureInfo SOAP编码请求的成功响应应该是一个由请求中InfoFormat参数指定的MIME类型的文档，且该文档由SOAP 1.2封装。

GetFeatureInfo响应格式没有指定任何特定响应格式，下面是一段灵活的XML元素，它着重于“GML简单要素专用标准第0级”推荐的响应格式：

```
<element name="FeatureInfoResponse">
<complexType>
<choice>
<element ref="gml:_FeatureCollection">
<element ref="wmts:TextPayload">
<element ref="wmts:BinaryPayload">
<element name="AnyContent" type="anyType">
</choice>
</complexType>
</element>
```

9.3.4 SOAP编码的GetFeatureInfo响应示例

9.3.2节定义的GetFeatureInfo SOAP请求的响应采用如下所示的GML编码：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope
http://www.w3.org/2003/05/soap-envelope.xsd">
<soap:Body>
<FeatureInfoResponse>
<ReguralGridedElevations
xmlns="http://www.maps.bob/etopo2"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.maps.bob/etopo2
wmtsGetFeatureInfo_response.xsd">
<featureMember>
<GridPoint_etopo2>
<elevation>503.0</elevation>
<TileRow>1</TileRow>
<TileCol>2</TileCol>
<J>86</J>
<I>132</I>
</GridPoint_etopo2>
</featureMember>
</ReguralGridedElevations>
</FeatureInfoResponse>
</soap:Body>
</soap:Envelope>
```

含有同样信息的GetFeatureInfo请求的HTML编码的SOAP响应结果如下所示：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope
http://www.w3.org/2003/05/soap-envelope.xsd">
<soap:Body>
<FeatureInfoResponse xmlns="http://www.opengis.net/wmts/1.0"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.opengis.net/wmts/1.0
wmtsGetFeatureInfo_response.xsd">
<TextPayload>
<Format>text/html</Format>
<TextContent>
<![CDATA[
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML>
<title>GetFeatureInfoResponse<title>
<b>Elevation</b>503.0<br>
<b>TileRow</b>1<br>
<b>TileCol</b>2<br>
<b>J</b>86<br>
<b>I</b>132<br>
</HTML>]]>
</TextContent>
</TextPayload>
</FeatureInfoResponse>
</soap:Body>
</soap:Envelope>
```

NOTEThe use of `<![CDATA[ ]]>` is needed for embedded HTML data but will not be needed for XHTML.

A GetFeatureInfo operation SOAP response with the same information in arbitrary xml encoding may look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope
http://www.w3.org/2003/05/soap-envelope.xsd">
<soap:Body>
<FeatureInfoResponse xmlns="http://www.opengis.net/wmts/1.0"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.opengis.net/wmts/1.0
wmtsGetFeatureInfo_response.xsd">
<AnyContent>
<GridPoint_etopo2>
<elevation>503.0</elevation>
<TileRow>1</TileRow>
<TileCol>2</TileCol>
<J>86</J>
<I>132</I>
</GridPoint_etopo2>
</AnyContent>
</FeatureInfoResponse>
</soap:Body>
</soap:Envelope>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope
http://www.w3.org/2003/05/soap-envelope.xsd">
<soap:Body>
<FeatureInfoResponse xmlns="http://www.opengis.net/wmts/1.0"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.opengis.net/wmts/1.0
wmtsGetFeatureInfo_response.xsd">
<TextPayload>
<Format>text/html</Format>
<TextContent>
<![CDATA[
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML>
<title>GetFeatureInfoResponse<title>
<b>Elevation</b>503.0<br>
<b>TileRow</b>1<br>
<b>TileCol</b>2<br>
<b>J</b>86<br>
<b>I</b>132<br>
</HTML>]]>
</TextContent>
</TextPayload>
</FeatureInfoResponse>
</soap:Body>
</soap:Envelope>
```

注：<![CDATA[ ]]> 对于 HTML 数据是必须的但对于 XHTML 则不是必须的。

含有同样信息的GetFeatureInfo操作的任意XML编码的SOAP响应如下所示：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope
http://www.w3.org/2003/05/soap-envelope.xsd">
<soap:Body>
<FeatureInfoResponse xmlns="http://www.opengis.net/wmts/1.0"
xmlns:gml="http://www.opengis.net/gml"
xsi:schemaLocation="http://www.opengis.net/wmts/1.0
wmtsGetFeatureInfo_response.xsd">
<AnyContent>
<GridPoint_etopo2>
<elevation>503.0</elevation>
<TileRow>1</TileRow>
<TileCol>2</TileCol>
<J>86</J>
<I>132</I>
</GridPoint_etopo2>
</AnyContent>
</FeatureInfoResponse>
</soap:Body>
</soap:Envelope>
```

9.4 Exceptions in SOAP encoding

If an error is detected while processing an operation request encoded in a SOAP envelope, the WMTS server SHALL generate a SOAP 1.2 response message where the content of the Body element is a Fault element containing an ExceptionReport element (as defined in clause 8.5 of OWS Common [OGC 06-121r3]). This SHALL be done using the following XML fragment:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
   <soap:Body>
         <soap:Fault>
               <soap:Code>
                     <soap:Value>soap:Receiver</soap:Value>
               </soap:Code>
               <soap:Reason>
                     <soap:Text>A server exception was
encountered.</soap:Text>
               </soap:Reason>
               <soap:Detail>
                     <ows:ExceptionReport
xmlns:ows="http://www.opengis.net/ows/1.1">…</ows:ExceptionReport>
               </soap:Detail>
         </soap:Fault>
   </soap:Body>
</soap:Envelope>
```

The Code element SHALL have the Value "soap:server" indicating that this is a server exception. The Reason element SHALL have the Text "A Server exception was encountered". This fixed string is used since the details of the exception SHALL be specified in the Detail element using an ows:ExceptionReport element.

10 WMTS using RESTful

A WMTS server that supports HTTP RESTful SHALL declare support for each resource by means of the ServiceMetadataURL (see Table 3) and the ResourceURL (see Table 6) elements of its ServiceMetadata document as explained in this clause. An example of this practice can be seen in the Annex D.

The first step in the resource oriented architectural style is to identify the resources and the relations between the resources. This version of WMTS standard identifies 3 resources: the service (ServiceMetadata), the map tiles (Tile) and the feature information related to a pixel of a tile (FeatureInfo). The RESTful approach provides a way to manipulate these resources via standard HTTP requests. This standard only defined the use of HTTP GET to download resource representations (that are equivalent to the ones that can be retrieved by the GetCapabilities, GetTile and GetFeatureInfo operations in the procedure oriented architectural style).

NOTE 1Some RESTful literature calls this action of retrieval (downloading) of a resource using HTTP GET "GetResource" or "GetResourceRepresentation", the action to delete a resource using HTTP DELETE DeleteResource, etc. This is useful to clarify the action of the HTTP operation but note that there is no operation called explicitly GetResourceRepresentation operation in RESTful. The WMTS standard also adopts this notation allowing easy extension to other future RESTful actions.

9.4 SOAP编码的异常

当检测到处理SOAP编码的操作请求有错误时，WMTS服务器应当生成一个符合SOAP1.2的响应消息，该消息Body元素的内容是包含ExceptionReport元素（OWS通用规范[OGC 06-121r3]8.5节定义的）的Fault元素。如以下XML片段所示：

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
   <soap:Body>
         <soap:Fault>
               <soap:Code>
                     <soap:Value>soap:Receiver</soap:Value>
               </soap:Code>
               <soap:Reason>
                     <soap:Text>A server exception was
encountered.</soap:Text>
               </soap:Reason>
               <soap:Detail>
                     <ows:ExceptionReport
xmlns:ows="http://www.opengis.net/ows/1.1">…</ows:ExceptionReport>
               </soap:Detail>
         </soap:Fault>
   </soap:Body>
</soap:Envelope>
```

Code元素应该含有"soap:server"的Value元素，用以表明这是服务器异常。Reason元素应当包含"A Server exception was encountered"文本内容。该异常的细节应当由Detail元素使用以固定字符串为值的ows:ExceptionReport元素来描述。

10 REST风格的WMTS

一个支持HTTP REST的WMTS服务器，应遵循本章的要求，通过服务元数据文档中的ServiceMetadataURL（见表3）和ResourceURL（见表6）元素声明对每种资源的支持。操作实例可以参见附录D。

面向资源架构风格的第一步是确定资源和资源之间的关系。该版本的WMTS标准确定了3类资源：服务（ServiceMetadata）、图块（Tle）和图块上像素关联的要素信息（FeatureInfo）。基于REST的方法提供了通过标准HTTP请求操作这些资源的途径。本标准只是定义了下载资源表示的HTTP GET的用法（这与面向过程风格的GetCapabilities, GetTile和GetFeatureInfo操作可以访问到的东西是等同的）。

注1 一些关于REST风格的文献把这种使用HTTP Get获取（下载）资源的动作称为"GetResource" 或"GetResourceRepresentation"，使用HTTP DELETE删除资源的动作称为DeleteResource，等等。这对于区分HTTP协议操作非常有用，但是注意在REST风格中并没有一种明确叫做GetResourceRepresentation的操作。WMTS标准也采用这种记法，以便于将来扩展出其他REST风格的动作。

The RESTful encoding for WMTS consists of a set of canonical URLs to the service metadata document, to tiles, and to FeatureInfo documents (one for each pixel). The service metadata document is the single entry point and all other URL endpoints can be obtained by analyzing the templates contained in the ResourceURL elements of each layer element in the ServiceMetadata document.

NOTE 2Other ways of connecting resources were studied during WMTS standard elaboration and OWS-6 and URL templates seemed an interesting convenience that allows the description of thousands of tile URLs and millions of pixels FeatureInfo URLs with a few expressions rather than describing every required URL endpoint individually.

10.1 ServiceMetadata resource (mandatory in resource oriented architectural style)

10.1.1 GetResourceRepresentation request

In the resource oriented architectural style, clients will access the ServiceMetadata document simply by requesting a file to a standard HTTP server using the URL. The ServiceMetadata document SHALL have one or more `<ServiceMetadataURL>` elements indicating a URL where this document can be obtained.

The URL referencing a ServiceMetadata document can have any form but we recommend the following syntax and format:

```
{WMTSBaseURL}/1.0.0/WMTSCapabilities.xml
```

Clients can also specify the MIME type of the ServiceMetadata document by including an "Accept: " parameter in the HTTP header of the request.

10.1.2 GetResourceRepresentation request example

To request WMTS ServiceMetadata document in XML format, from a WMTS server that implements the resource oriented architectural style, a client could issue the following RESTful URL:

```
http://www.maps.bob/1.0.0/WMTSCapabilities.xml
```

10.1.3 ServiceMetadata representation

In response to a valid request for a ServiceMetadata representation from a client, a WMTS server SHALL send a ServiceMetadata document which conforms to the model described in 7.1.1. Servers SHALL be able to respond with a ServiceMetadata document in XML format (application/xml) that conforms to the XML schema described in Annex B but other formats are also allowed.

10.1.4 ServiceMetadata representation example

In response to a valid request for a ServiceMetadata representation from a client, a WMTS server might generate a document that looks like the one in subclause 7.1.1.3.

10.1.5 GetResourceRepresentation exception

If a client requests a document version or a format extension that is not available on a particular server, the server SHALL return an HTTP Error 404 (File Not Found).

WMTS的REST风格编码由一组分别指向服务元数据文档、各个图块以及每个像素处的要素信息的规范URL组成。服务元数据文档是唯一的入口，其他URL端点都可以通过分析服务元数据文档中每个图层元素内各个ResourceURL包含的模板来获得。

注2 获取资源的其他方式在WMTS标准研究过程中和OWS-6中都进行了研究，发现URL模板机制非常便于描述成千上万的图块URL和针对无数像素点FeatureInfo的URL，不需要单独描述每一个需要的端点，只需要很少几个表达式即可。

10.1 ServiceMetadata资源(面向资源架构风格下为必选)

10.1.1 GetResourceRepresentation请求

在面向资源架构风格下，客户端仅需使用特定的URL向一个标准HTTP服务器请求文件的方式即可访问ServiceMetadata文档。ServiceMetadata文档应包含一或多个<ServiceMetadataURL>元素以表明可以获取该文档的一个URL。

该URL指向的ServiceMetadata文档可以是任何形式，但是我们建议采用如下语法和格式：

{WMTSBaseURL}/1.0.0/WMTSCapabilities.xml

客户端可以通过在HTTP请求头中包含一个"Accept: "的参数来指定ServiceMetadata文档的MIME类型。

10.1.2 GetResourceRepresentation请求实例

要从一个实现了面向资源架构风格的WMTS服务器请求一个XML格式的WMTSServiceMetadata文档，客户端可以发送以下REST风格的URL：

```
http://www.maps.bob/1.0.0/WMTSCapabilities.xml
```

10.1.3 ServiceMetadata表示

要响应来自客户端对ServiceMetadata表示结果的有效请求，WMTS服务器应发送一个符合7.1.1中要求的ServiceMetadata文档。服务器应能够响应一个符合附录B给出的XML模式的XML格式ServiceMetadata文档，但其他格式也是允许的。

10.1.4 ServiceMetadata表示实例

要响应来自客户端对ServiceMetadata表示结果的有效请求，WMTS服务器可以生成一个类似7.1.1.3中的那样一个文档。

10.1.5 GetResourceRepresentation异常

如果某个服务器上没有客户端请求的文档版本或格式，服务器应返回一个HTTP 404错误（文件未发现）。

10.2 Tile resource (mandatory in resource oriented architectural style)

10.2.1 GetResourceRepresentation request

The ServiceMetadata document in the resource oriented architectural style MAY contain a list of Layer elements and each layer that is available to be retrieved in this architectural style SHALL have one or more <ResourceURL> elements with the "resourceType" attribute set to "tile" and a template attribute. In this RESTful approach the template attribute contains a URL template that can be converted to a URL by using a template processor and then get the expected tile in the format specified by the attribute "format" by requesting the resource with a standard HTTP GET.

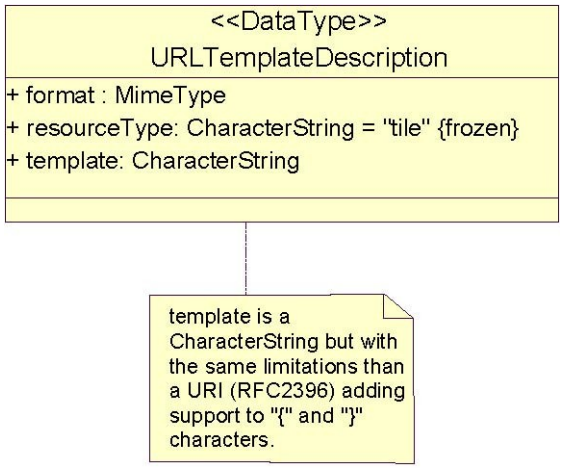

Figure 13 — URLTemplate for tile UML class diagram

Table 31 — Parts of the URLTemplate data structure for tiles

Names	Definition	Data type and values	Multiplicity and use
format `format`	Format of the resource representation that can be retrieved one resolved the URL template	ows:MimeType	One (mandatory)
resourceType `resourceType`	Resource type to be retrieved	Character String type, not empty SHALL contain "tile"	One (mandatory)
template `template`	URL template a.	Character String type but with the same limitations than a URI (RFC2396) adding support to "{" and "}" characters.	One (mandatory)
a A template processor will apply the rules in Table 32 to get a URL to a resource.			

A template processor is a program or library that runs on the client side and converts a URL template into a URL. It will have to process the URL template that contains variable names marked off in matching braces ('{', '}') and substitute them with the corresponding valid value. The template processor SHALL support the following variable names: "Style", "TileMatrixSet", "TileMatrix", "TileRow", "TileCol" and any dimension identifier that has been defined for this layer in the element ./Dimension/ows:Identifier. The template processor SHALL substitute the variable names by the values of the elements as shown in Table 32.

10.2 图块资源（面向资源架构风格下必选）

10.2.1 GetResourceRepresentation请求

面向资源架构风格的ServiceMetadata文档可以包含一组Layer元素，该架构风格下可以访问的每一层应有一或多个<ResourceURL>元素,并包含一个取值为tile的resourceType属性和一个template属性。在REST风格方法中，template属性包含一个URL模板，该模板可以通过模板处理器转换成一个URL，然后据此使用标准的HTTP GET方式请求资源就可以获得format属性指定格式的图块。

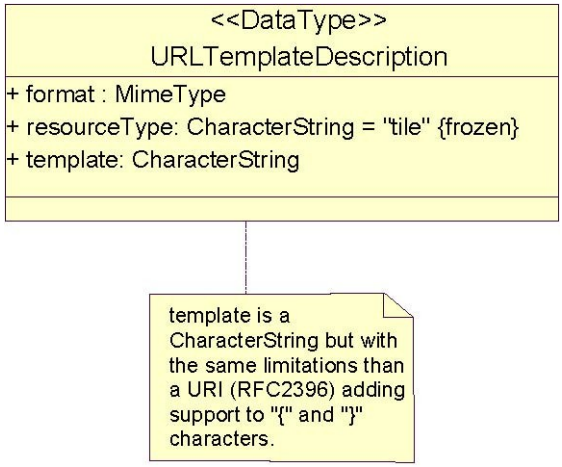

图 13 —Tile 的 URLTemplate 的 UML 类图

表 31 — 针对 Tile 的 URLTemplate 数据结构组成

名称	定义	数据类型与取值	出现次数与用法
format format	URL模板解析后可获得资源表示结果的格式	ows:MimeType	1 (必选)
resourceType resourceType	待获取资源的类型	字符串，非空。取值应包含"tile"	1 (必选)
template template	URL模板a.	字符串，但符合URI (RFC2396)的约束过则，且增加对"{" 和"}" 两个字符的支持。	1 (必选)
a 模板处理器将使用表32中的规则获得资源的URL。			

模板处理器是运行在客户端、负责把URL模板转换成URL的一个程序或软件库。它需要把URL模板中包含的由匹配符('{', '}')标记的变量名称替换成相应有效值。模板处理器应支持以下变量名称："Style", "TileMatrixSet", "TileMatrix", "TileRow", "TileCol"以及在该图层的./Dimension/ows:Identifier元素中定义的任何支持的维度。模板处理器应使用表32所示的元素值对变量名称进行替换。

Possible values and ranges of the variables in the URL template can be extracted from the SeviceMetadata document parameters. The following table lists all the possible variable names, their description, possible values and multiplicity. Assuming a ServiceMetadata in XML format the possible values are also given using XPath expressions that point to the ServiceMetadata document. When a relative XPath is used, it is relative to its layer element of the ServiceMetadata document.

When a variable has only one possible value for this layer, the use of the direct value instead of the variable is recommended on the URL template.

Table 32 — URL template variables and possible values for tile

URL template variable	Meaning	Possible values	Multiplicity
`"style"`	Style identifier	identifier in Table 7 `./style/ows:Identifier`	One (mandatory) [a]
`./Dimension/ows:Identifier`	Dimension value	identifier in Table 9 `./Dimension[ows:Identifier={./Dimension/ows:Identifier}]/Value`	One for each dimension available (mandatory if there are dimensions defined)
`"TileMatrixSet"`	tile matrix set identifier	identifier in Table 6 `./ TileMatrixSetLink/TileMatrixSet`	One (mandatory) [a]
`"TileMatrix"`	tile matrix identifier	identifier in Table 14 `/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix/ows:Identifier`	One (mandatory) [a]
`"TileRow"`	row index of tile matrix	If TileMatrixSetLimits is present, see Table 10, (`./TileMatrixSetLimits`), SHALL be any integer value between MinTileRow and MaxTileRow in Table 12 (both included) (./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MinTileRowand ./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MaxTileRow). else SHALL be any integer value between 0 and MatrixHeight – 1, see Table 14, (both included) (0and/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/MatrixHeight – 1)	One (mandatory) [a]
`"TileCol"`	col index of tile matrix	If TileMatrixSetLimits is present, see Table 10, (`./TileMatrixSetLimits`), SHALL be any integer value between MinTileCol and MaxTileCol in Table 12 (both included) (./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MinTileColand ./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MaxTileCol) else SHALL be any integer value between 0 and MatrixWidth – 1, see Table 14, (both included) (0and/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/ MatrixWidth – 1)	One (mandatory) [a]
a When a variable has only one possible value for this layer the use of the direct value instead of the variable is recommended.			

URL模板中各变量的可能取值和值域可以从ServiceMetadata文档参数中抽取。下表列出了所有可用变量名称、含义、可能取值以及出现次数。如果服务元数据为XML格式，则可能取值使用指向服务元数据文档的XPATH表达式给出。如果使用了相关的XPath，则与其服务元数据文档的Layer元素相关联。

如果某个变量对于该图层只有一个可取的值，则建议在URL模板中直接使用该值而不使用变量。

表 32 — 针对图块的 URL 模板变量和可能取值

URL模板变量	含义	可能取值	出现次数
"style"	图式标识符	表7中定义的标识符 ./style/ows:Identifier	1 (必选) a
./Dimension/ows:Identifier	维度值	表9中定义的标识符 ./Dimension[ows:Identifier={./Dimension/ows:Identifier}]/Value	每一个可用维度一个 (若定义了维度，则必选)
"TileMatrixSet"	块阵集标识符	表6中定义的标识符 ./ TileMatrixSetLink/TileMatrixSet	1 (必选) a
"TileMatrix"	块阵标识符	表14中定义的标识符 /Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix/ows:Identifier	1 (必选) a
"TileRow"	块阵行号	如果存在TileMatrixSetLimits，见表10，(./TileMatrixSetLimits)，应为表12定义的MinTileRow和MaxTileRow之间（闭包含）的任何整型值。(./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MinTileRowand ./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MaxTileRow). 如果不存在TileMatrixSetLimits，为0到MatrixHeight – 1之间的任何整型值，见表14（闭包含） (0和/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/MatrixHeight – 1)	1 (必选) a
"TileCol"	块阵列号	如果存在TileMatrixSetLimits，见表10,(./TileMatrixSetLimits)，应为表12定义的MinTileCol和MaxTileCol之间（闭包含）的任何整型值。(./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MinTileColand ./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MaxTileCol) 如果不存在TileMatrixSetLimits，为0到MatrixWidth– 1之间的任何整型值，见表14（闭包含） (0和/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/MatrixWidth – 1)	1 (必选) a
a如果某个变量对于该图层只有一个可取的值，则建议直接使用该值而不使用变量。			

Any possible order of the variables and values in the URL template is valid. Neverdeless we recommend the following order:

style, firstDimension, ..., lastDimension, TileMatrixSet, TileMatrix, TileRow and TileCol

NOTE 1 It is not necessary for a file system with this particular structure to actually exist on the server. A WMTS server implementation is free to parse the request URL itself and either remap the request to a different internal directory structure or generate the response on the fly (at the expense of speed).

NOTE 2 This syntax and processing is in-line with the pre-existing templating schemas present in OpenSearch, WSDL and WADL and is a simplification of the description made by the IETF Network Working Group in the Internet-Draft called "URI Template draft-gregorio-uritemplate-03".

10.2.2 GetResourceRepresentation request example

An example of a tile URL for a resource representation in RESTful HTTP GET equivalent of the request in subclause 8.2.1 is:

```
http://www.maps.bob/etopo2/default/WholeWorld_CRS_84/10m/1/3.png
```

It corresponds to the following ResourceURL element:

```
<ResourceURL
format="image/png" resourceType="tile"
template="http://www.maps.bob/etopo2/default/{TileMatrixSet}/{TileMatrix}/{Til
eRow}/{TileCol}.png">
```

10.2.3 Tile representation

In response to a valid request for a Tile representation from a client, a WMTS server SHALL send either an image representation of the tile or a reference to an image, as stated in subclause 7.2.1. An image is the most typical representation but representations in other formats are also allowed.

10.2.4 Tile representation example

The response could be the same image shown in subclause 8.2.4

10.2.5 GetResourceRepresentation exception

If the response of a GetResourceRepresentation request for a tile is unsuccessful, the server SHALL return an HTTP error code that SHOULD be accompanied by an XML ExceptionReport document as defined in subclause 8.5 of OWS Common [OGC 06-121r3] and subclause 7.2.2.2 of this document. Determination of what error conditions map to which HTTP error codes is up to the discretion of the server. If the error condition is due to a malformed request or the resource does not exist (for instance in a request URL with illegal path values), the HTTP error code returned SHOULD be HTTP Error 404 (File Not Found).

NOTE For a server where the tile resource files actually exists and there is no specific application capable of generating specific responses on the fly, HTTP error code SHALL be expected but an XML ExceptionReport document cannot be expected.

URL模板中变量与值的任何排列顺序都是有效的，但我们建议采用如下顺序：

style, firstDimension, ..., lastDimension, TileMatrixSet, TileMatrix, TileRow and TileCol

注1 在服务器上并不需要实际存在与该结构一样的一个文件系统。WMTS服务器实现自行解析请求URL，可以把请求映射到一个不同的内部目录结构或实时生成一个响应结果（这样会牺牲速度）。

注2 该语法和处理方式与OpenSearch, WSDL 和 WADL的已有模板模式是一致的，同时也是IETF网络工作组给出的称为“URI模板draft-gregorio-uritemplate-03”因特网草案的一个简化。

10.2.2 GetResourceRepresentation请求实例

与8.2.1中请求对应的REST风格HTTP GET方式的图块资源获取URL实例：

http://www.maps.bob/etopo2/default/WholeWorld_CRS_84/10m/1/3.png

该URL对应以下ResourceURL元素：

```
<ResourceURL format="image/png" resourceType="tile"
template="http://www.maps.bob/etopo2/default/{TileMatrixSet}/{TileMatrix}/{TileRow}/{TileCol}.png">
```

10.2.3 图块的表示

要响应一个客户端针对图块表示结果的有效请求，WMTS服务器应发送一个图块的图像表示或者对图像的引用，如7.2.1中的说明。图像是最典型的表示形式，但是其他格式的表示形式也是允许的。

10.2.4 图块表示形式实例

响应结果可以为8.2.4所示的相同图片。

10.2.5 GetResourceRepresentation异常

如果针对图块的GetResourceRepresentation请求响应不成功，服务器应返回一个HTTP错误码，并伴随一个遵循OWS通用实现规范[OGC 06-121r3]8.5和本标准7.2.2.2要求的XML格式ExceptionReport文档。确定什么样的出错条件映射到相应的HTTP出错码取决于服务器的判断。如果出错条件是源于不符合要求的请求或资源不存在（如URL中存在非法的路径值），则HTTP出错码应为HTTP404错误（文件未发现）。

注 对于图块资源文件实际存在且没有明确应用能够实时生成响应的服务器，期望返回HTTP出错码，而不是XML格式的ExceptionReport文档。

10.3 FeatureInfo resource (optional in resource oriented architectural style)

10.3.1 GetResourceRepresentation request

The ServiceMetadata document in the resource oriented architectural style may contain a list of Layer elements and each layer that is available to be retrieved in this architectural style and is queryable SHALL have one or more <ResourceURL> elements with the "resourceType" attribute set to "FeatureInfo" and a template attribute. In this RESTful approach the template attribute contains a URL template that can be converted to a URL by using a template processor and then get the expected FeatureInfo in the format specified by the attribute "format" by requesting the resource with a standard HTTP GET.

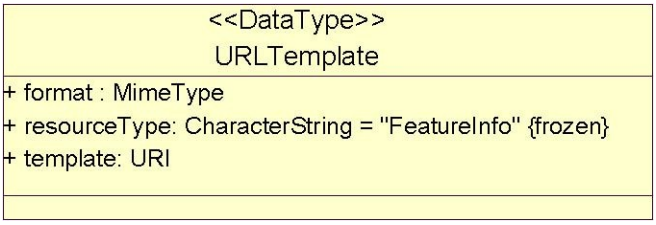

Figure 14 — URLTemplate for the FeatureInfo UML class diagram

Table 33 — Parts of the URLTemplate data structure for FeatureInfo

Names	Definition	Data type and values	Multiplicity and use
format `format`	Format of the resource representation that can be retrieved or resolved from the URL template	ows:MimeType	One (mandatory)
resourceType `resourceType`	Resource type to be retrieved	Character String type, not empty SHALL contain "FeatureInfo"	One (mandatory)
template `template`	URL template a.	URI	One (mandatory)
a A template processor will apply the rules in Table 34 to get a URL to a resource.			

A template processor is a program or library that runs on the client side and converts a URL template into a URL. It will have to process the URL template that contains variable names marked off in matching braces ('{', '}') and substitute them with the corresponding valid value. The template processor SHALL support the following variable names: "Style", "TileMatrixSet", "TileMatrix", "TileRow", "TileCol", "J", "I", and any dimension identifier that has been defined for this layer in the element ./Dimension/ows:Identifier. The template processor SHALL substitute the variable names by the values shown in Table 34.

Possible values and ranges of the variables in the URL template can be extracted from the SeviceMetadata document parameters. The following table lists all the possible variable names, their description, possible values and multiplicity. Assuming a ServiceMetadata in XML format the possible values are also given using XPath expressions that point to the ServiceMetadata document. When a relative XPath is used, it is relative to its layer element of the ServiceMetadata document.

When a variable has only one possible value for this layer, the use of the direct value instead of the variable is recommended on the URL template.

10.3 FeatureInfo资源 (面向资源架构风格下为可选)

10.3.1 GetResourceRepresentation请求

面向资源架构风格下的ServiceMetadata文档可以包含一组Layer元素，该架构下每个可查询图层应包含一或多个<ResourceURL>元素，该元素包含一个值为"FeatureInfo"的"resourceType"的属性和一个template属性。该架构下，该template属性包含一个URL模板，通过模板处理器可以吧URL模板转换为一个URL，然后利用该URL发送一个标准的HTTP GET请求可以获得期望的通过format属性指定格式的FeatureInfo。

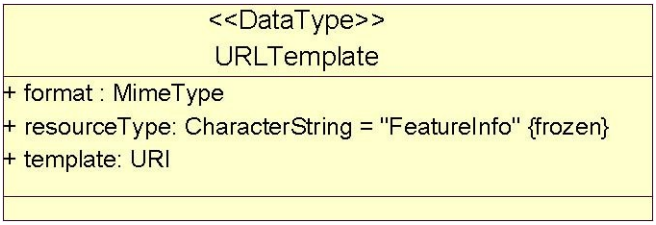

图 14 —FeatureInfo 的 URLTemplate UML 类图

表 33 —针对 FeatureInfo 的 URLTemplate 数据结构的组成

名称	定义	数据类型与取值	出现次数与用法
format format	从URL模板中可以获取或解析的资源表示形式的格式	ows:MimeType	1 (必选)
resourceType resourceType	待获取资源的类型	字符串类型，非空，取值为"FeatureInfo"	1 (必选)
template template	URL 模板 a.	URI	1 (必选)
a 模板处理器应实现表34的规则以获得资源的URL。			

模板处理器是运行在客户端、负责把URL模板转换成URL的一个程序或软件库。他需要把URL模板中包含的由匹配符('{', '}')标记的变量名称替换成相应有效值。模板处理器应支持以下变量名称："Style", "TileMatrixSet", "TileMatrix", "TileRow", "TileCol"，"J", "I"以及在该图层的./Dimension/ows:Identifier元素中定义的任何支持的维度。模板处理器应使用表34所示的元素值对变量名称进行替换。

URL模板中各变量的可能取值和值域可以从ServiceMetadata文档参数中提取。下表列出了所有可用变量的名称、其描述、可能取值和出现次数。如果服务元数据为XML格式，则可能取值使用指向服务元数据文档的XPATH表达式给出。如果使用了相关的XPath，则与其服务元数据文档的图层元素相关联。

如果某个变量对于该图层只有一个可取的值，则建议在URL模板中直接使用该值而不使用变量。

Table 34 — URL template variables and possible values for FeatureInfo

URL template variable	Meaning	Possible values	Multiplicity
"style"	Style identifier	identifier in Table 7 ./style/ows:Identifier	One (mandatory) [a]
./Dimension/ows:Identifier	Dimension value	identifier in Table 9 ./Dimension[ows:Identifier={./Dimension/ows:Identifier}]/Value	One for each dimension available (mandatory if there are dimensions defined)
"TileMatrixSet"	tile matrix set identifier	identifier in Table 6 ./ TileMatrixSetLink/TileMatrixSet	One (mandatory) [a]
"TileMatrix"	tile matrix identifier	identifier in Table 14 /Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix/ ows:Identifier	One (mandatory) [a]
"TileRow"	row index of tile matrix	If TileMatrixSetLimits is present, see Table 10, (./TileMatrixSetLimits), SHALL be any integer value between MinTileRow and MaxTileRow in Table 12 (both included) (./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MinTileRowand ./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MaxTileRow). else SHALL be any integer value between 0 and MatrixHeight – 1, see Table 14, (both included) (0and/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/MatrixHeight – 1)	One (mandatory) [a]
"TileCol"	col index of tile matrix	If TileMatrixSetLimits is present, see Table 10, (./TileMatrixSetLimits), SHALL be any integer value between MinTileCol and MaxTileCol in Table 12 (both included) (./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MinTileColand ./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MaxTileCol) else SHALL be any integer value between 0 and MatrixWidth – 1, see Table 14, (both included) (0and/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/ MatrixWidth – 1)	One (mandatory) [a]
"J"	Pixel row index in a tile	SHALL be any integer value between 0 and TileHeight-1 (see Table 14) (both included) (0 and /Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/TileHeight – 1)	One (mandatory) [a]
"I"	Pixel column index in a tile	SHALL be any integer value between 0 and TileWidth -1 (see Table14) (both included) (0 and /Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/TileWidth – 1)	One (mandatory) [a]
a When a variable has only one possible value for this layer the use of the direct value instead of the variable is recommended.			

表 34 — 针对 FeatureInfo 的 URL 模板变量和可能取值

URL模板变量	含义	可能取值	出现次数
"style"	图式标识符	表7中定义的标识符 ./style/ows:Identifier	1 (必选) [a]
./Dimension/ows:Identifier	维度的取值	表9中定义的标识符 ./Dimension[ows:Identifier={./Dimension/ows:Identifier}]/Value	每个可用维度一个（若定义了维度则为必选）
"TileMatrixSet"	块阵集的标识符	表6中定义的标识符 ./ TileMatrixSetLink/TileMatrixSet	1 (必选) [a]
"TileMatrix"	块阵的标识符	表14中定义的标识符 /Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix/ ows:Identifier	1 (必选) [a]
"TileRow"	块阵的行号	如果存在TileMatrixSetLimits，参见表10, (./TileMatrixSetLimits),应为MinTileRow到MaxTileRow间（表12）的任何整数（闭包含） (./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MinTileRowand ./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MaxTileRow). 如果不存在TileMatrixSetLimits，应为0到MatrixHeight – 1之间（闭包含）的任何整数值，见表14。 (0和/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/MatrixHeight – 1)	1 (必选) [a]
"TileCol"	块阵的列号	如果存在TileMatrixSetLimits，参见表10, (./TileMatrixSetLimits)，应为MinTileCol到MaxTileCol间（表12）的任何整数（闭包含） (./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MinTileColand ./tileMatrixSetLimits/tileMatrixLimits[./TileMatrix={TileMatrix}]/MaxTileCol) 如果不存在TileMatrixSetLimits，应为0到MatrixWidth – 1之间（闭包含）的任何整数值，见表14。 (0和/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/MatrixWidth – 1)	1 (必选) [a]
"J"	图块内像素的行号	应为0到TileHeight– 1之间（闭包含）的任何整数值，见表14。(0 和/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/TileHeight – 1)	1 (必选) [a]
"I"	图块内像素的列号	应为0到TileWidth– 1之间（闭包含）的任何整数值，见表14。(0 和/Capabilities/Contents/TileMatrixSet[ows:Identifier={TileMatrixSet}]/TileMatrix[ows:Identifier={TileMatrix}]/TileWidth – 1)	1 (必选) [a]
a 如果变量只有一个可取的值，就直接使用值而不使用变量。			

Any possible order of the variables and values in the URL template is valid. Neverdeless we recommend the following order:

style, firstDimension, ..., lastDimension, TileMatrixSet, TileMatrix, TileRow, TileCol, J and I.

NOTE It is highly improbable that all of the GetFeatureInfo response file URLs physically exist in a practical implementation, so for a GetFeatureInfo-enabled server it seems inevitable that a RESTful server implementation would generate them on the fly or cache them.

10.3.2 GetResourceRepresentation request example

An example FeatureInfo request for a resource representation in RESTful HTTP GET equivalent of request in subclause 8.2.1 is

```
http://www.maps.bob/etopo2/ default/WholeWorld_CRS_84/10m/1/3/86/132.xml
```

It corresponds to the following ResourceURL element:

```
<ResourceURL format="application/gml+xml; version=3.1" resourceType="FeatureInfo" template="http://www.maps.bob/etopo2/default/{TileMatrixSet}/{TileMatrix}/{TileRow}/{TileCol}/{J}/{I}.xml">
```

10.3.3 FeatureInfo representation

In response to a valid request for a FeatureInfo representation from a client, a WMTS server SHALL send a document with information related to a particular pixel of a tile or a reference to resources that represents the geographic data portrayed on that area, as stated in subclause 7.3.1. An XML document conforming GML Simple Features Profile [06-049r1] is the most typical representation but other representations and formats are also allowed.

10.3.4 FeatureInfo representation as an XML document example

The response could be the same xml document shown in subclause 8.3.4

10.3.5 GetResourceRepresentation exception

If the response of a GetResourceRepresentation request for a FeatureInfo is unsuccessful the server SHALL return an HTTP error code that SHOULD be accompanied by an XML ExceptionReport document as defined in subclause 8.5 of OWS Common [OGC 06-121r3] and subclause 7.3.2.2 of this document. Determination of what error conditions map to which HTTP error codes is up to the discretion of the server. If the error condition is due to a malformed request or the resource does not exist (for instance in a request URL with illegal path values), the HTTP error code returned SHOULD be HTTP Error 404 (File Not Found).

11 Recommendations to improve interoperability and performance

For maximum interoperability, this WMTS standard makes the following recommendations.

11.1 Server and Client support for KVP, SOAP and RESTful

This specification defines 3 interfaces KVP, SOAP and RESTful. Clients and servers are encouraged to support as many interfaces as possible to improve interoperability. The minimum recommended support is:

A WMTS client SHOULD support both KVP and RESTful. SOAP support is optional.

A WMTS server SHOULD support KVP and/or RESTful. SOAP support is optional.

URL模板中变量与值的任何排列顺序都是有效的，但我们建议采用如下顺序：

style, firstDimension, ..., lastDimension, TileMatrixSet, TileMatrix, TileRow, TileCol, J 和 I.

注 所有GetFeatureInfo响应文件的URL存在于一个实际实现中是基本不可能的，因此对于支持GetFeatureInfo的服务器，似乎显然REST风格的服务器实现需要实时生成这些响应结果或者缓存事先生成的结果。

10.3.2 GetResourceRepresentation请求实例

与8.2.1中请求等效的REST风格HTTP GET方式的FeatureInfo资源获取请求的实例：

```
http://www.maps.bob/etopo2/ default/WholeWorld_CRS_84/10m/1/3/86/132.xml
```

该URL对应以下ResourceURL元素：

```
   <ResourceURL format="application/gml+xml; version=3.1"
resourceType="FeatureInfo"
template="http://www.maps.bob/etopo2/default/{TileMatrixSet}/{TileMatrix}/{Til
eRow}/{TileCol}/{J}/{I}.xml">
```

10.3.3 FeatureInfo表示形式

要响应一个客户端针对FeatureInfo表示结果的有效请求，WMTS服务器应发送一个包含与某一图块特定像素位置关联信息的文档或一个对代表该区域图示的地理信息资源的引用，如7.3.1中的说明。一个与GML简单要素规范[06-049r1]一致的XML文档是最典型的表示形式，但是其他格式的表示形式也是允许的。

10.3.4 FeatureInfo表示为XML文档的实例

响应结果可以与8.3.4中所示的XML文档相同。

10.3.5 GetResourceRepresentation异常

如果针对FeatureInfo的GetResourceRepresentation请求响应不成功，服务器应返回一个HTTP错误码，并伴随一个遵循OWS通用实现规范[OGC 06-121r3]8.5和本标准7.3.2.2要求的XML格式ExceptionReport文档。确定什么样的出错条件映射到相应的HTTP出错码取决于服务器的判断。如果出错条件是源于不符合要求的请求或资源不存在（如URL中存在非法的路径值），则HTTP出错码应为HTTP404错误（文件未发现）。

11 关于提高互操作性和性能的建议

为了实现最大可能的互操作性，本WMTS标准提出以下建议：

11.1 服务器与客户端对KVP、SOAP和REST的支持

本规范定义了KVP、SOAP和REST风格的三种接口。鼓励客户端和服务器应尽可能支持较多的接口以提高互操作性。建议至少应支持：

WMTS客户端应同时支持KVP和REST风格的接口，SOAP方式的接口为可选。

WMTS服务器应同时支持KVP和REST，或支持其中之一，SOAP方式的接口为可选。

11.2 A standard set of scales

It is strongly recommended that WMTS servers offer their Layers using, where possible, the well-known scale sets defined in the Annex E and reference them in the service metadata document in the TileMatrixSetDef section. This would aid interoperability between systems and allow users to "mash-up" suitable services.

11.3 A standard image format and FeatureInfo document response

It is recommended that servers offer Tiles in the image/png and image/jpeg file formats. The image/png image format is good for categorical maps and image/jpeg is better for imagery but since image/jpeg does not support transparency, image/png may be used for images. It is recommended that WMTS Clients support both. Since a GetTile operation can serve only one tile at a time, it is important that clients have the ability to support transparency and also be able to overlap tiles from the same geographical area.

It is recommended that FeatureInfo documents be offered in the MIME type format " application/gml+xml; version=3.1" and to follow GML Simple Features Profile [06-049r1]. See subclause 7.3.1.

11.4 Number of TileMatrixSets and TileMatrixSetLimits

In a server where all layers cover almost the same area, or layers can cover the same area in the future it is recommended to use the same TileMatrixSet for all layers and to use TileMatrixSetLimits to inform clients about the tile matrix indices currently available.

This will avoid changing all tile indices if some future change extends the area covered.

11.5 Cacheble resources

One of the main goals of this standard is to provide a better cache support for repetitive requests, helping WMTS servers to save resources and perform better. Web caching occurs at several levels, but for caching to occur at any of these levels caching mechanisms need to know that resources are cachable. Servers should include appropriate caching information (expiration date) in the WMTS server responses to avoid impacting performance. In a RESTful file based server where tiles are pre-rendered and stored in directories that are directly accessed by generic internet servers, the expiration date of the data can be configured as a property of this directory.

Internally this is achieved by means of the proper HTTP control headers:

HTTP 1.0, uses the "Expires" header. This header indicates an expiration date. If your data is guarantied to be static, or you know when the data is going to be updated, you can use a convenient future date in the Expires header.

HTTP 1.1, uses the "Cache-control" header. This header indicates a period of time to cache the data before expiration. If your data is guarantied to be static, or you know when the data is going to be updated, you can use a convenient period of time in the Cache-control header.

11.2 标准比例尺系列

强烈建议WMTS服务器可能情况下使用附录E定义的知名比例尺系列提供其图层，并在服务元数据文档中的TileMatrixSetDef引用使用的比例尺系列。这有助于系统间的互操作，并允许用户可以混搭相关服务。

11.3 标准图像格式和FeatureInfo文档响应

建议服务器使用image/png 和 image/jpeg格式提供图块服务。image/png图像格式适用于地图，image/jpeg更适于影像，但是由于image/jpeg不支持透明，image/png格式也可用于影像。建议WMTS客户端两者都支持。因为GetTile操作一次只能提供一个图块，客户端具有支持透明的能力且可以把属于相同区域的图块进行叠加非常重要。

建议FeatureInfo文档以" application/gml+xml; version=3.1"格式提供，并遵循GML简单要素规范[06-049r1]的要求，见7.3.1。

11.4 TileMatrixSets和TileMatrixSetLimits的数量

在一个所有图层基本都覆盖相同区域或将来会覆盖相同区域的服务器内，建议所有图层使用同一个TileMatrixSet，并使用TileMatrixSetLimits告知客户端当前可用的块阵行列值。

这可以避免将来可能的覆盖范围变化导致对所有行列值的改动。

11.5 可缓存的资源

本标准的主要目标是为重复请求提供一个更好的缓存支持，以帮助WMTS服务器节约资源并提供更高的性能。基于Web的缓存分为几个层面，但是要实现任何一个层面上的缓存，缓存机制都需要知道资源是可缓存的。服务器应在WMTS服务器响应中包含适当的缓存信息（过期时间）以避免影响性能。在REST风格的基于文件的服务器中，图块是预生成的且存储在目录中并直接通过通用因特网服务器提供访问，数据的过期时间可以作为目录的属性进行配置。

内部来说这可以通过适当的HTTP控制头来实现：

HTTP 1.0使用"Expires"头信息。该头信息表示过期日期。如果数据确定是静态的，或数据即将更新，可以在Expires中随便使用一个将来的日期。

HTTP1.1使用"Cache-control"头信息。该头信息表示过期前的缓存时间段。如果数据确保是静态的，或将被更新，可以在Cache-control中随便使用一个时间段。

Annex A（normative）Abstract test suite

A.1 Introduction

This abstract test suite specifies at a high level how server and client implementations of this standard SHALL be tested for conformance to this standard. The framework for such abstract test suites is specified in ISO 19105: Geographic information – Conformance and testing, especially Clauses 7 and 9.

An abstract test suite contains multiple abstract tests, grouped into one or more test modules. This abstract test suite consists of two top-level test modules:

a) Client test module – Abstract tests for checking conformance of client implementations with the requirements of this standard that are normatively referenced by this Implementation Specification.

b) Server test module – Abstract tests for checking conformance of server implementations with the requirements of this standard that are normatively referenced by this Implementation Specification.

Any of these modules could contain lower-level test modules. At this time, only the Server test module contains lower-level test modules, namely:

a) All operations implemented test module – Abstract tests for checking server properties that are common to all operations implemented.

b) GetCapabilities, GetTile and GetFeatureInfo operation test module – Abstract tests for checking server properties that are specific to an operation.

In the client and server test modules, all operations specified and implemented SHALL be tested, including KVP HTTP GET, and SOAP HTTP POST transfer and RESTFul HTTP GET transfer of each operation request. In the standard test module, all operations specified SHALL be checked, including KVP HTTP GET, SOAP HTTP POST and RESTFul HTTP GET transfer of operation requests. And all operation request and response parameters specified or implemented SHALL be tested. Of course, some operations, transfer methods, and parameters are specified as optional implementation by servers. Any optional item not implemented by a server SHALL not be tested. Also, items not implemented by a client SHALL not be tested.

A.2 Client test module

A.2.1 GetCapabilities operation request

a) Test Purpose: Verify that a client satisfies all requirements for a GetCapabilities operation request.

b) Test Method: Generate an adequate sample of GetCapabilities operation requests from the client, and verify that each is a valid request.

c) Reference: Subclause 7.1.2.1

d) Test Type: Basic

附录A（规范性）抽象测试套

A.1 引言

本抽象测试套在高层次说明应如何测试本标准的服务器和客户端实现是否与本标准一致。抽象测试套的框架在ISO 19105：地理信息——一致性与测试，特别是其中的第7和第9章中规定。

抽象测试套包括多个抽象测试，分为一或多个测试模块。本抽象测试套由两个顶层测试模块组成：

a) 客户端测试模块　根据本标准规范性引用的需求，检查客户端实现一致性的抽象测试。

b) 服务器端测试模块　根据本标准规范性引用的需求，检查服务器实现一致性的抽象测试。

任何这些模块可以包含低层次的测试模块。但本标准目前只是服务器测试模块包含低层次的测试模块，即：

a）所有实现操作的测试模块：检查所有实现操作共同的服务器特性的抽象测试。

b）GetCapabilities, GetTile 和 GetFeatureInfo操作测试模块：检查针对特定操作的服务器特性的抽象测试。

在客户端和服务器测试模块，所有规定和实现的操作都应测试，包括每个操作请求的KVP HTTP GET、SOAP HTTP POST传输和REST风格的HTTP GET传输。在标准测试模块中，所有规定的操作都应检查，包括操作请求的KVP HTTP GET、SOAP HTTP POST传输和REST风格的HTTP GET传输。所有规定和实现的操作请求和响应参数都应测试。当然，某些操作、传输方法和参数，标准规定在服务器实现时是可选的。任何服务器未实现的可选项不应测试。同样，客户端未实现的可选项也不应测试。

A.2 Client test module客户端测试模块

A.2.1 GetCapabilities操作请求

a）测试目的：检查客户端是否符合GetCapabilities操作请求的所有要求

b) 测试方法：从客户端生成一个包含足够GetCapabilities操作请求的样本，检查每个操作请求是否都是有效的。

c) 参照章节：7.1.2.1

d)测试类型：基本测试

A.2.2 GetTile operation request

a) Test Purpose: Verify that a client satisfies all requirements for a GetTile operation request.

b) Test Method: Generate an adequate sample of GetTile operation requests from the client, and verify that each is a valid request.

c) Reference: Subclause 7.2.2

d) Test Type: Basic

A.2.3 Contiguous GetTile operation requests

a) Test Purpose: Verify that a client is capable of generating contiguous GetTile operation requests.

b) Test Method: Generate adequate samples of GetTile operation requests from the client, for tiles that are contiguous, and verify that the client is able to show them without any discontinuity.

c) Reference: Subclause 7.2.1 and 7.2.2

d) Test Type: Capability

A.2.4 Overlay and transparency in GetTile operation request

a) Test Purpose: Verify that a client is capable to generating overlaying GetTile operation requests, each one for a different layer.

b) Test Method: Generate adequate samples of GetTile operation requests from the client, for tiles that overlays and verify that the client is able to show them correctly overlaid and with a transparency when NODATA is present on the upper images when the image format allows that.

c) Reference: Subclauses 7.2.1,7.2.2 and 11.3

d) Test Type: Capability

A.2.5 Different TileMatrixSets in GetTile operation requests

a) Test Purpose: Verify that a client can correctly overlay layers with layers having different TileMatrixSets.

b) Test Method: Generate adequate samples of GetTile operation requests from the client of tiles from layers with different TileMatrixSets, and verify that client is able to show them correctly overlaid.

c) Reference: Subclauses 7.2.1 and 7.2.2

d) Test Type: Capability

A.2.2 GetTile操作请求

a）测试目的：检查客户端是否符合GetTile操作请求的所有要求

b) 测试方法：从客户端生成一个包含足够GetTile操作请求的样本，检查每个操作请求是否都是有效的。

c) 参照章节：7.2.2

d)测试类型：基本测试

A.2.3 连续的GetTile操作请求

a）测试目的：检查客户端是否能够产生连续的GetTile操作请求

b) 测试方法：针对连片图块，从客户端生成一个包含足够GetTile操作请求的样本，检查客户端是否能够连续显示这些图块。

c) 参照章节：7.2.1和7.2.2

d)测试类型：能力测试

A.2.4 GetTile操作请求中的叠盖和透明

a）测试目的：检查客户端是否能够为不同图层产生叠盖的GetTile操作请求

b) 测试方法：针对叠盖图块，从客户端生成一个包含足够GetTile操作请求的样本，检查客户端是否能够正确叠盖显示这些图块，并且看如果格式允许且在上层图块有无数据部分时透明是否起效。

c) 参照章节：7.2.1、7.2.2和11.3

d)测试类型：能力测试

A.2.5 GetTile操作请求中的不同TileMatrixSet

a）测试目的：检查客户端是否能够正确叠加显示分属不同块阵集的图层。

b) 测试方法：针对分属不同块阵集的不同图层的图块，从客户端生成一个包含足够GetTile操作请求的样本，检查客户端是否能够正确叠加显示这些图块。

c) 参照章节：7.2.1和7.2.2

d)测试类型：能力测试

A.2.6 Optional GetFeatureInfo operation request

a) Test Purpose: Verify that a client satisfies all requirements for a GetFeatureInfo operation requests.

b) Test Method: Generate an adequate sample of GetFeatureInfo operation requests from the client, and verify that each is a valid request.

c) Reference: Subclause 7.3.2.1

d) Test Type: Basic

A.2.7 GetTile and GetFeatureInfo operation request form ServiceMetadata content.

e) Test Purpose: Verify that a client is able to parse a GetCapabilities response and generate GetTile and GetFeatureInfo operation request.

f) Test Method: Generate an adequate sample of GetCapabilities operation requests from the client, and verify that the client is able to generate valid GetTile and GetFeatureInfo requests conforming with the content section of the ServiceMetadata document.

g) Reference: Subclause 7.1.1

h) Test Type: Basic

A.3 Server test module

A.3.1 All operations implemented test module

A.3.1.1 HTTP protocol usage

a) Test purpose: Verify that the rules and conventions governing the use of HTTP are observed.

b) Test method: TBD

c) Reference: RFC 2616 (Hypertext Transfer Protocol -- HTTP/1.1). See <http://www.ietf.org/rfc/rfc2616>.

d) Test type: Capability

A.3.1.2 Accept HTTP GET and POST transferred operation requests

a) Test Purpose: Verify that a server accepts HTTP GET and/or HTTP POST transferred requests for each operation.

b) Test Method: Submit HTTP GET and/or HTTP POST transferred requests for each operation. Verify that the server accepts and responds to these requests as specified and implemented. Check that the server accepts at least one HTTP GET or HTTP POST transfer of requests for each operation.

c) Reference: Clause 8, and 9

d) Test Type: Capability

A.2.6 可选的GetFeatureInfo操作请求

a）测试目的：检查客户端是否符合GetFeatureInfo操作请求的所有要求

b) 测试方法：从客户端生成一个包含足够GetFeatureInfo操作请求的样本，检查是否每个都是有效请求。

c) 参照章节：7.3.2.1

d)测试类型：基本测试

A.2.7 根据服务元数据内容构造 GetTile和GetFeatureInfo操作请求

a）测试目的：检查客户端是否能够解析GetCapabilities响应结果，并据此生成GetTile和GetFeatureInfo操作请求。

b) 测试方法：从客户端生成一个包含足够GetCapabilities操作请求的样本，检查客户端是否能够生成与服务元数据文档content子集一致的有效GetTile 和GetFeatureInfo操作请求。

c) 参照章节：7.1.1

d)测试类型：基本测试

A.3 服务器测试模块

A.3.1 所有已实现操作的测试模块

A.3.1.1 HTTP协议用法

a）测试目的：检查使用HTTP需要遵循的规则和约定是否得到遵守。

b) 测试方法：待定。

c) 参照章节：RFC 2616 (超文本传输协议-- HTTP/1.1). 参见 <http://www.ietf.org/rfc/rfc2616>

d)测试类型：能力测试

A.3.1.2 接收HTTP GET 和 POST 方式传输的操作请求

a）测试目的：针对每个操作，检查服务器是否接收HTTP GET 和/或HTTP POST传输的请求。

b) 测试方法：针对每个操作，提交HTTP GET和/或HTTP POST传输的请求。检查服务器是否接收并对这些请求按规定和实现进行响应。检查对于每个操作，服务器至少接收HTTP GET和HTTP POST中的一种请求传输方式。

c) 参照章节：第8、9章

d)测试类型：能力测试

A.3.1.3 Handle KVP-encoded operation requests

a) Test Purpose: Verify that a server handles all parameter names in a KVP-encoded operation request in a capitalization- and sequence-insensitive manner.

b) Test Method: Submit KVP-encoded GetCapabilities and other operation requests containing parameter names using various combinations of cases, with a variety of parameter sequences. Verify that the server provides the same response when the same parameter names use different cases and combinations of cases.

c) Reference: Clause 8

d) Test Type: Capability

A.3.1.4 Handle SOAP-encoded operation requests

a) Test Purpose: Verify that a server handles all parameters in a XML-encoded operation request in a name-capitalization and parameter-sequence sensitive manner.

b) Test Method: Submit SOAP-encoded GetCapabilities request and other operation requests containing parameters using correct and incorrect name capitalizations and parameter sequences. Verify that the server accepts all correct requests, and returns ExceptionReport messages for all incorrect requests.

c) Reference: Clause 9

d) Test Type: Capability

A.3.1.5 Handle HTTP GET RESTful -encoded operation requests

a) Test Purpose: Verify that a server handles a URL service metadata request.

b) Test Method: Request a Service metadata URL and other URL resources using correct and incorrect URLs. Verify that the server respond with the right resource to correct URLs, and a returns HTTP errors for invalid URLs.

c) Reference: Clause 10

d) Test Type: Capability

A.3.1.6 KVP and SOAP HTTP response status code

a) Test purpose: Verify that a service request which generates an exception produces a response that contains 1) a correct service exception report, and 2) the correct status code indicating the error.

b) Test method: Check the response code in the Status-Line and the message body. Pass if the response code is either 4xx (Client error) or 5xx (Server error) and the body contains a service exception report. Fail otherwise.

c) Reference: RFC 2616, clause 11 and Subclauses 7.1.2.2, 7.2.2.2 and 7.3.2.2

d) Test type: Capability

A.3.1.3 处理KVP编码的操作请求

a）测试目的：检查服务器是否是以大小写和顺序无关的方式处理KVP编码的操作请求中的所有参数名称。

b) 测试方法：提交包含各种大小写和参数顺序的KVP编码的GetCapabilities和其他操作请求。对于同样的参数名称使用不同大小写的情况，检查服务器是否提供相同的响应。

c) 参照章节：第8章

d)测试类型：能力测试

A.3.1.4 处理SOAP编码的操作请求

a）测试目的：检查服务器是否是以大小写和顺序相关的方式处理XML编码的操作请求中的所有参数名称。

b) 测试方法：提交包含正确和不正确大小写和不同参数顺序的SOAP编码的GetCapabilities和其他操作请求。检查服务器是否接受所有正确请求，而对所有不正确的请求返回一个异常报告。

c) 参照章节：第9章

d)测试类型：能力测试

A.3.1.5 处理HTTP GET方式REST风格编码的操作请求

a）测试目的：检查服务器是否正确处理URL服务元数据请求。

b) 测试方法：分别使用正确的和错误的URL发送一个服务元数据URL和其他资源的URL。检查服务器是否对所有正确URL响应正确的资源，而对所有无效URL返回HTTP错误信息。

c) 参照章节：第10章

d)测试类型：能力测试

A.3.1.6 KVP 和SOAP HTTP 响应状态码

a）测试目的：检查产生异常的服务请求生成的响应是否包含1）一个正确的服务异常报告 2）表示该异常的正确状态码。

b) 测试方法：检查状态行中的响应码和消息体。如果响应码是4XX（客户端错误）或5XX（服务器错误），并且消息体中包含一个服务一场报告则通过，否则为失败。

c) 参照章节：RFC 2616，第11章、7.1.2.2, 7.2.2.2 和 7.3.2.2

d)测试类型：能力测试

A.3.2 GetCapabilities operation request test module (Procedure Oriented Architectural Style)

A.3.2.1 Accept HTTP GET transferred operation requests

a) Test Purpose: Verify that a server accepts at least HTTP GET transferred requests for the GetCapabilities operation.

b) Test Method: Submit HTTP GET transferred requests for the GetCapabilities operation. Verify that the server accepts and responds to these requests as specified.

c) Reference: Subclause 8.1.1

d) Test Type: Capability

A.3.2.2 GetCapabilities operation response

a) Test Purpose: Verify that a server satisfies all requirements of the GetCapabilities operation response.

b) Test Method: Make several GetCapabilities requests including a variety of input parameters. Verify that the specified correct response is returned to each request.

c) Reference: Subclause 8.1.3

d) Test Type: Capability

A.3.2.3 Version negotiation

a) Test Purpose: Verify that a server satisfies the requirements for version negotiation.

b) Test Method: Submit GetCapabilities operation requests containing version numbers lower than, higher than, and equal to the version supported by the server. Verify that the server responses are in accord with the specified rules for version negotiation.

c) Reference: Subclause 7.3.2 of OWS Common [OGC 06-121r3]

d) Test Type: Capability

A.3.2.4 Format selection

a) Test Purpose: Verify that a server satisfies the requirements for format selection, if the server implements the AcceptFormats request parameter.

b) Test Method: Submit GetCapabilities operation requests containing supported and unsupported values for the AcceptFormats parameter. Verify that the server responses are in accord with the specified rules for format selection.

c) Reference: Subclause 7.3.5 of OWS Common [OGC 06-121r3]

d) Test Type: Capability

A.3.2 GetCapabilities操作请求测试模块(面向过程架构风格下)

A.3.2.1 接收HTTP GET方式传输的操作请求

a）测试目的：针对GetCapabilities操作，检查服务器至少接受HTTP GET方式传输的请求。

b) 测试方法：针对GetCapabilities操作，提交HTTP GET传输的请求。检查服务器是否接收并对这些请求按规定进行响应。

c) 参照章节：8.1.1

d)测试类型：能力测试

A.3.2.2 GetCapabilities操作响应

a）测试目的：检查服务器是否满足所有GetCapabilities操作响应的要求。

b) 测试方法：生成几个包括若干输入参数的GetCapabilities请求。检查服务器是否对这些请求按规定进行正确响应。

c) 参照章节：8.1.3

d)测试类型：能力测试

A.3.2.3 版本协商

a）测试目的：检查服务器是否满足所有版本协商的要求。

b) 测试方法：提交包括低于、高于以及等于服务器支持版本的版本号的GetCapabilities操作请求。检查服务器响应是否与规定的版本协商要求一致。

c) 参照章节：7.3.2和OWS通用实现规范[OGC 06-121r3]

d)测试类型：能力测试

A.3.2.4 格式选择

a）测试目的：检查服务器是否满足格式选择的要求，是否实现了AcceptFormats请求参数。

b) 测试方法：提交包括支持和不支持的AcceptFormats参数值的GetCapabilities操作请求。检查服务器响应是否与规定的格式选择要求一致。

c) 参照章节：7.3.5和OWS通用实现规范[OGC 06-121r3]

d) 测试类型：能力测试

A.3.2.5 Handling updateSequence parameter

a) Test Purpose: Verify that a server satisfies the requirements for generating and using the updateSequence parameter, if the server implements the AcceptFormats request parameter.

b) Test Method: Submit GetCapabilities operation requests containing correct and incorrect values of the AcceptFormats parameter. Verify that the server provides the specified correct response to each request.

c) Reference: Subclause 7.3.4 of OWS Common [OGC 06-121r3]

d) Test Type: Capability

A.3.2.6 Section selection

a) Test Purpose: Verify that a server satisfies the requirements for using the Sections parameter, if the server implements the Sections request parameter.

b) Test Method: Submit GetCapabilities operation requests containing various values and combinations of values of the Sections parameter. Verify that the server provides the specified correct response to each request

c) Reference: Subclause 7.1.2.1 and Table 18

d) Test Type: Capability

A.3.3 ServiceMetadata resource test module (Resource Oriented Architectural Style)

A.3.3.1 Accept HTTP GET transferred operation requests

a) Test Purpose: Verify that a server sends a correct ServiceMetadata resource.

b) Test Method: Perform HTTP GET requests to the ServiceMetadata URL Verify that the specified correct response is returned to each request.

c) Reference: Subclause 10.1.1

d) Test Type: Capability

A.3.4 ServiceMetadata response

A.3.4.1 XML well formated

a) Test purpose: Verify that the ServiceMetadata document is a valid xml document.

b) Test method: Submit GetCapabilities operation requests and verify that the returned document is a well formed xml document.

c) Reference: Subclause 7.1.1

d) Test type: Capability

A.3.2.5 处理updateSequence 参数

a）测试目的：检查服务器是否满足产生和使用updateSequence参数的要求，是否实现了AcceptFormats请求参数。

b) 测试方法：提交包括支持和不支持的AcceptFormats参数值的GetCapabilities操作请求。检查服务器响应是否与规定的格式选择要求一致。

c) 参照章节：7.3.4和OWS通用实现规范[OGC 06-121r3]

d)测试类型：能力测试

A.3.2.6 子集（Section）选择

a）测试目的：检查服务器是否满足使用Sections参数的要求，是否实现了Sections请求参数。

b) 测试方法：提交包括各种Sections参数值及其组合的GetCapabilities操作请求。检查服务器对每个请求的响应是否与规定一致。

c) 参照章节：7.1.2.1和表18

d) 测试类型：能力测试

A.3.3 ServiceMetadata资源测试模块(面向资源架构风格下)

A.3.3.1 接收 HTTP GET方式传输的操作请求

a）测试目的：检查服务器是否发送了一个正确的ServiceMetadata资源。

b) 测试方法：向ServiceMetadata的URL提交若干HTTP Get请求，检查是否对这些请求返回了规定的正确响应。

c) 参照章节：10.1.11

d)测试类型：能力测试

A.3.4 ServiceMetadata响应

A.3.4.1 格式良好的XML

a）测试目的：检查ServiceMetadata文档是一个有效的xml文档。

b) 测试方法：提交一个GetCapabilities请求，检查返回的文档是否是格式良好的xml文档。

c) 参照章节：7.1.1

d) 测试类型：能力测试

A.3.4.2 XML references the normative schema

a) Test purpose: Verify that the normative content of the schema document referred to by the schemaLocation attribute in the ServiceMetadata document is identical to the normative content of the on-line schema referred to in the Annex B.

b) Test method: Pass if the normative content of the schema document referred to by the schemaLocation attribute in the ServiceMetadata document is identical to the normative content of the on-line schema referred to in the Annex B.1.

c) Reference: Subclause 7.1.1

d) Test type: Capability

A.3.4.3 XML validates against the schema

a) Test purpose: Verify that the response to a GetCapabilities request validates against the schema(s) provided with the schemaLocation attribute.

b) Test method: Pass if the response to a GetCapabilities request validates against the schema(s) provided with the schemaLocation attribute.

c) Reference: Subclause 7.1.1

d) Test type: Capability

A.3.4.4 OnLineResource is an only resource prefix

a) Test purpose: Verify that each OnlineResource URL intended for HTTP Get requests in the ServiceMetadata document is a URL prefix.

b) Test method: Pass if each OnlineResource URL intended for HTTP Get requests in the ServiceMetadata document is a URL prefix.

c) Reference: Subclause 7.1.1

d) Test type: Capability

A.3.4.5 XML format for GetCapabilities

a) Test purpose: Verify that the server advertises the application/xml format for the GetCapabilities operation.

b) Test method: Pass if the server advertises the application/xml format for the GetCapabilities operation.

c) Reference: Subclause 7.1.1

d) Test type: Capability

A.3.4.2 XML引用规范性模式

a）测试目的：检查ServiceMetadata文档中通过schemaLocation引用的模式文档的规范性内容与附录B中引用的在线模式的规范性内容是否相同。

b) 测试方法：如果ServiceMetadata文档中通过schemaLocation引用的模式文档的规范性内容与附录B.1中引用的在线模式的规范性内容相同，则通过。

c) 参照章节：7.1.1

d)测试类型：能力测试

A.3.4.3 依据模式对XML进行有效性检查

a）测试目的：检查GetCapabilities请求的响应结果是否符合schemaLocation属性指定模式的有效性要求。

b) 测试方法：如果GetCapabilities请求的响应结果符合schemaLocation属性指定模式的有效性要求，则通过。

c) 参照章节：7.1.1

d)测试类型：能力测试

A.3.4.4 OnLineResource仅为资源前缀

a）测试目的：检查ServiceMetadata文档中每个意图发起HTTP GET请求的OnlineResource URL是否都是一个URL前缀。

b) 测试方法：如果ServiceMetadata文档中每个意图发起HTTP GET请求的OnlineResource URL都是一个URL前缀，则通过。

c) 参照章节：7.1.1

d)测试类型：能力测试

A.3.4.5 GetCapabilities的XML格式

a）测试目的：检查服务器是否为GetCapabilities操作告示了application/xml格式。

b) 测试方法：如果服务器为GetCapabilities操作告示了application/xml格式，则通过。

c) 参照章节：7.1.1

d)测试类型：能力测试

A.3.4.6 ows:Constraint GetEncoding

a) Test Purpose: Verify that a server satisfies the requirements for using the ows:Constraint GetEncoding parameter, if the server implements the Sections request parameter.

b) Test Method: Verify that the server provides a Service Metadata document that includes ows:Constraint GetEncoding information on OperationsMetadata section. Verify that the server is able to respond the encodings specified.

c) Reference: Subclause 7.1.1

d) Test Type: Capability

A.3.4.7 ows:Constraint PostEncoding

a) Test Purpose: Verify that a server satisfies the requirements for using the ows:Constraint PostEncoding parameter, if the server implements the Sections request parameter.

b) Test Method: Verify that the server provides a Service Metadata document that includes ows:Constraint PostEncoding information on OperationsMetadata section. Verify that the server is able to respond the encodings specified.

c) Reference: Subclause 7.1.1

d) Test Type: Capability

A.3.4.8 Layer identifiers

a) Test purpose: Verify that the layer identifiers are different.

b) Test method: Pass if all the layers have different non empty identifiers.

c) Reference: Subclause 7.1.1.1.2 and Table 6

d) Test type: Capability

A.3.4.9 Layer LegendURL are correct resources

a) Test purpose: Verify that the metadata for each of the LegendURL resources is correct.

b) Test method: Pass if all the submodules and subtests pass.

c) Reference: Subclause 7.1.1.1.2 and Table 8

d) Test type: Capability

A.3.4.10 Layer LegendURL correct Format

a) Test purpose: Verify that the MIME-type returned for the LegendURL resource is the advertised format.

b) Test method: Pass if the MIME-type returned for the LegendURL resource is the advertised format.

c) Reference: Subclause 7.1.1.1.2 and Table 11

d) Test type: Capability

A.3.4.6 ows:Constraint的GetEncoding

a）测试目的：检查服务器是否满足使用ows:Constraint GetEncoding参数的要求，是否实现了Sections请求参数。

b) 测试方法：检查服务器提供的服务元数据文档在OperationsMetadata子集中是否包含ows:Constraint GetEncoding信息。检查服务器是否能够按指定的编码响应。

c) 参照章节：7.1.1

d)测试类型：能力测试

A.3.4.7 ows:Constraint的PostEncoding

a）测试目的：检查服务器是否满足使用ows:Constraint PostEncoding参数的要求，是否实现了Sections请求参数。

b) 测试方法：检查服务器提供的服务元数据文档在OperationsMetadata子集中是否包含ows:Constraint PostEncoding信息。检查服务器是否能够按指定的编码响应。

c) 参照章节：7.1.1

d)测试类型：能力测试

A.3.4.8 Layer标识符

a）测试目的：检查各个图层标识符是否各不相同。

b) 测试方法：如果所有的图层标识符都不相同，则通过。

c) 参照章节：7.1.11.2和表6

d) 测试类型：能力测试

A.3.4.9 Layer的LegendURL是正确的资源

a）测试目的：检查每个LegendURL资源的元数据是否都是正确的。

b) 测试方法：如果所有的子模块和下一测试都通过，则通过。

c) 参照章节：7.1.11.2和表8

d) 测试类型：能力测试

A.3.4.10 Layer的LegendURL资源的格式正确

a）测试目的：检查为LegendURL资源返回的MME类型与告示的格式是否一致。

b) 测试方法：如果为LegendURL资源返回的MME类型与就是告示的格式，则通过。

c) 参照章节：7.1.11.2和表11

d) 测试类型：能力测试

A.3.4.11 Layer LegendURL correct sizes

a) Test purpose: Verify that the size of the LegendURL resource is the advertised width and the advertised height.

b) Test method: Pass if the size of the LegendURL resource is the advertised width and the advertised height.

c) Reference: Subclause 7.1.1.1.2 and Table 8

d) Test type: Capability

A.3.4.12 Layer TileMatrixSet is valid

a) Test purpose: Verify that Layer TileMatrixSet contains a correct identifier.

b) Test method: Pass if Layer TileMatrixSet value is equal to a TileMatrixSet identifier in the content section.

c) Reference: Subclause 7.1.1.1.2 and Table 8

d) Test type: Capability

A.3.4.13 TileMatrixSet Identifier r

a) Test purpose: Verify that TileMatrixSet identifiers are correct.

b) Test method: Pass if all TileMatrixSet have different non empty identifiers.

c) Reference: Subclause 7.1.1.1.2 and Table 13

d) Test type: Capability

A.3.4.14 TileMatrix Identifier

a) Test purpose: Verify that TileMatrix identifiers are correct.

b) Test method: Pass if all TileMatrix in a TileMatrixSet have different non empty identifiers.

c) Reference: Subclause 7.1.1.1.2 and Table 14

d) Test type: Capability

A.3.4.15 TileMatrixSet ScaleDenominators

a) Test purpose: Verify that ScaleDenominator values are correct.

b) Test method: Pass if ScaleDenominators in a TileMatrixSet have different non empty values.

c) Reference: Subclause 7.1.1.1.2 and Table 14

d) Test type: Capability

A.3.4.11 Layer的LegendURL资源的大小正确

a）测试目的：检查LegendURL资源的大小与告示的格宽高是否一致。

b) 测试方法：如果LegendURL资源的大小与告示的格宽高一致，则通过。

c) 参照章节：7.1.11.2和表8

d) 测试类型：能力测试

A.3.4.12 Layer的TileMatrixSet有效

a）测试目的：检查Layer的TileMatrixSet是否包含一个正确的标识符。

b) 测试方法：如果Layer的TileMatrixSet的值等于content子集中的一个TileMatrixSet的标识符，则通过。

c) 参照章节：7.1.11.2和表8

d) 测试类型：能力测试

A.3.4.13 TileMatrixSet的Identifier

a）测试目的：检查各个TileMatrixSet的标识符是否正确。

b) 测试方法：如果所有TileMatrixSet的标识符都不同且非空，则通过。

c) 参照章节：7.1.11.2和表13

d) 测试类型：能力测试

A.3.4.14 TileMatrix的Identifier

a）测试目的：检查各个TileMatrix的标识符是否正确。

b) 测试方法：如果所有TileMatrix的标识符都不同且非空，则通过。

c) 参照章节：7.1.11.2和表14

d) 测试类型：能力测试

A.3.4.15 TileMatrixSet的ScaleDenominators

a）测试目的：检查TileMatrixSet的各个ScaleDenominator取值是否正确。

b) 测试方法：如果TileMatrixSet所有ScaleDenominator的取值都不同且非空，则通过。

c) 参照章节：7.1.11.2和表14

d) 测试类型：能力测试

A.3.4.16 TileMatrixSet WellKnownScaleSet

a) Test purpose: Verify that a WellKnownScaleSet is compatible with ScaleDenominator values.

b) Test method: When a WellKnownScaleSet is advertised, there has to be a TileMatrix for each different ScaleDenominator with values starting from the largest scale denominator in the WellKnownScaleSet table and all intermediate scales denominators down to some ScaleDenominator minimum value for this Layer.

c) Reference: Subclause 7.1.1.1.2 and Table 14

d) Test type: Capability

A.3.4.17 Theme LayerRef is valid

a) Test purpose: Verify that Theme LayerRef contains a correct identifier.

b) Test method: Pass if each Theme LayerRef value is equal to a Layer identifier in the content section.

c) Reference: Subclause 7.1.1.1.3 and Table 15

d) Test type: Capability

A.3.5 Tile request test module

A.3.5.1 GetTile Layer

a) Test purpose: Verify that when a request contains a Layer incorrect value, then the server throws an exception.

b) Test method: When a request contains a Layer that is not advertised in the ServiceMetadata document, then the server throws an exception (code= InvalidParameterValue) in Procedure Oriented Architectural Style.

c) Reference: Subclause 7.2.2.2 and Table 26

d) Test type: Capability

A.3.5.2 Tile ResourceURL template

a) Test Purpose: Verify that a client supports URL templates and server satisfies RESTful requests

b) Test Method: Verify that the server provides a Service Metadata document that includes complete ResourceURL information with resourceType=tile on Layer section if tiles of this layer are able for RESTful. Verify that the template processor in the client is able to convert the URL template in a correct URL to a tile and the server is able to respond RESTful requests.

c) Reference: Subclause 10.2.1

d) Test Type: Capability

A.3.4.16 TileMatrixSet的WellKnownScaleSet

a）测试目的：检查WellKnownScaleSet和ScaleDenominator的值是否匹配。

b) 测试方法：如果告示了WellKnownScaleSet，则针对从WellKnownScaleSet表内和所有中间比例尺分母中的最大比例尺分母到该图层某些最小比例尺取值的每一个不同ScaleDenominator，都必须有一个对应的TileMatrix。

c) 参照章节：7.1.11.2和表14

d) 测试类型：能力测试

A.3.4.17 Theme的LayerRef有效

a）测试目的：检查Theme的LayerRef是否包含一个正确的标识符。

b) 测试方法：如果Theme的LayerRef的值等于content子集中的一个图层标识符，则通过。

c) 参照章节：7.1.11.3和表15

d) 测试类型：能力测试

A.3.5 图块请求测试模块

A.3.5.1 GetTile图层

a）测试目的：检查如果请求中包含一个Layer的不正确取值，服务器是否抛出一个异常。

b) 测试方法：面向过程架构风格下，如果请求中包含一个在ServiceMetadata 文档中未告知的Layer，服务器应抛出一个异常（code= InvalidParameterValue）。

c) 参照章节：7.2.2.2和表26

d) 测试类型：能力测试

A.3.5.2 Tile的ResourceURL模板

a）测试目的：检查客户端是否支持URL模板，服务器端是否满足REST风格请求的要求。

b) 测试方法：检查服务器是否提供了一个在Layer子集中包含属性resourceType等于tile的完整的ResourceURL信息的服务元数据文档，而且该图层的所有图块都是可以通过REST风格方式获取到。检查客户端的模板处理器是否能够把URL模板转换成指向图块资源的正确URL，而且服务器是否能够对REST风格的请求作出响应。

c) 参照章节：10.2.1和表26

d) 测试类型：能力测试

A.3.5.3 Tile TileMatrixSet

a) Test purpose: Verify that when a request contains a TileMatrixSet incorrect value, then the server throws an exception.

b) Test method: When a request contains a TileMatrixSet value that is not advertised for this Layer in the ServiceMetadata document, then the server throws an exception (code= InvalidParameterValue) in Procedure Oriented Architectural Style or a HTTP 404 File not found in Resource Oriented Architectural Style.

c) Reference: Subclause 7.2.2.2 and Table 26

d) Test type: Capability

A.3.5.4 Tile TileMatrix

a) Test purpose: Verify that when a request contains a TileMatrix incorrect value, then the server throws an exception.

b) Test method: When a request contains a TileMatrix value that is not advertised for this TileMatrixSet in the ServiceMetadata document, then the server throws an exception (code= InvalidParameterValue) in Procedure Oriented Architectural Style or a HTTP 404 File not found in Resource Oriented Architectural Style.

c) Reference: Subclause 7.2.2.2 and Table 26

d) Test type: Capability

A.3.5.5 Tile TileRowand TileCol

a) Test purpose: Verify that when a request contains a TileRow or TileCol incorrect, then the server throws an exception.

b) Test method: When a request contains a TileRow or TileCol greater or equal to MatrixHeight and MatrixWidth respectively for the selected TileMatrix, then the server throws an exception (code= TileOutOfRange) in Procedure Oriented Architectural Style or a HTTP 404 File not found in Resource Oriented Architectural Style.

c) Reference: Subclause 7.2.2.2 and Table 26

d) Test type: Capability

A.3.5.6 Tile incorrect Sytle

a) Test purpose: Verify that when a request contains a Style incorrect value, then the server throws an exception.

b) Test method: When a request contains a Style value that is not advertised for this Layer in the ServiceMetadata document, then the server throws an exception (code= InvalidParameterValue) in Procedure Oriented Architectural Style or a HTTP 404 File not found in Resource Oriented Architectural Style.

c) Reference: Subclause 7.2.2.2 and Table 26

d) Test type: Capability

A.3.5.3 Tile的TileMatrixSet

a）测试目的：检查如果请求中包含一个TileMatrixSet的不正确取值，服务器是否抛出一个异常。

b) 测试方法：如果请求中包含一个在ServiceMetadata 文档中未告知的TileMatrixSet取值，面向过程架构风格下服务器应抛出一个异常（code= InvalidParameterValue），或者面向资源架构风格下返回一个HTTP 404（文件未找到）的错误。

c) 参照章节：7.2.2.2和表26

d) 测试类型：能力测试

A.3.5.4 Tile的TileMatrix

a）测试目的：检查如果请求中包含一个TileMatrix的不正确取值，服务器是否抛出一个异常。

b) 测试方法：如果请求中包含一个在ServiceMetadata 文档中未告知的TileMatrix取值，面向过程架构风格下服务器应抛出一个异常（code= InvalidParameterValue），或者面向资源架构风格下返回一个HTTP 404（文件未找到）的错误。

c) 参照章节：7.2.2.2和表26

d) 测试类型：能力测试

A.3.5.5 Tile的TileRow和TileCol

a）测试目的：检查如果请求中包含一个TileRow或TileCol的不正确取值，服务器是否抛出一个异常。

b) 测试方法：如果请求中的TileRow或TileCol取值分别大于或等于所选TileMatrix的MatrixHeight 和MatrixWidth，面向过程架构风格下服务器应抛出一个异常（code= TileOutOfRange），或者面向资源架构风格下返回一个HTTP 404（文件未找到）的错误。

c) 参照章节：7.2.2.2和表26

d) 测试类型：能力测试

A.3.5.6 Tile包含不正确的Sytle

a）测试目的：检查如果请求中包含一个Style的不正确取值，服务器是否抛出一个异常。

b) 测试方法：如果请求中包含一个在ServiceMetadata 文档中未告知的Style取值，面向过程架构风格下服务器应抛出一个异常（code= InvalidParameterValue），或者面向资源架构风格下返回一个HTTP 404（文件未找到）的错误。

c) 参照章节：7.2.2.2和表26

d) 测试类型：能力测试

A.3.5.7 Tile incorrect dimension value

a) Test purpose: Verify that when a request contains an incorrect dimension value, then the server throws an exception.

b) Test method: When a request contains a dimension value that is not advertised for this Layer in the ServiceMetadata document, then the server throws an exception (code= InvalidParameterValue) in Procedure Oriented Architectural Style or a HTTP 404 File not found in Resource Oriented Architectural Style.

c) Reference: Subclause 7.2.2.2 and Table 26

d) Test type: Capability

A.3.5.8 Tile dimension default and current

a) Test purpose: Verify that the server supports 'default' and 'current'.

b) Test method: When the ServiceMetadata document advertises a default value or the current value support, request that uses the 'default' and 'current' keywords returns a correct answer.

c) Reference: Subclause 7.1.1 Table 9

d) Test type: Capability

A.3.5.9 GetTile incorrect Format

a) Test purpose: Verify that when a request contains a Format incorrect value, then the server throws an exception.

b) Test method: When a request contains a Format value that is not advertised for this Layer in the ServiceMetadata document, then the server throws an exception (code= InvalidParameterValue) in Procedure Oriented Architectural Style

c) Reference: Subclause 7.2.2.2 and Table 26

d) Test type: Capability

A.3.5.10 Tile correct Format

a) Test purpose: Verify that for each GetTile format, when the Format parameter is set to that format or URLtemplate having a format parameter, the MIME type of the response matches that format.

b) Test method: Pass if for each GetTile format, when the Format parameter is set to that format, the MIME type of the response matches that format in Procedure Oriented Architectural Style or the URL template format of the response matches that format in Resource Oriented Architectural Style

c) Reference: Subclause 7.2.1 and 10.2.1

d) Test type: Capability

A.3.5.7 Tile包含不正确的维度值

a）测试目的：检查如果请求中包含一个dimension的不正确取值，服务器是否抛出一个异常。

b) 测试方法：如果请求中包含一个在ServiceMetadata 文档中未告知的dimension取值，面向过程架构风格下服务器应抛出一个异常（code= InvalidParameterValue），或者面向资源架构风格下返回一个HTTP 404（文件未找到）的错误。

c) 参照章节：7.2.2.2和表26

d) 测试类型：能力测试

A.3.5.8 Tile的维度取值为“default”和“current”

a）测试目的：检查服务器是否支持'default' 和'current'。

b) 测试方法：如果ServiceMetadata 文档中告知了对'default' 和'current'两个值的支持，使用'default' 和'current'两个值的请求应获得正确的响应。

c) 参照章节：7.1.1和表9

d) 测试类型：能力测试

A.3.5.9 GetTile的格式不正确

a）测试目的：检查如果请求中包含一个Format的不正确取值，服务器是否抛出一个异常。

b) 测试方法：如果请求中包含一个在ServiceMetadata 文档中未告知的Format取值，面向过程架构风格下服务器应抛出一个异常（code= InvalidParameterValue）。

c) 参照章节：7.2.2.2和表26

d) 测试类型：能力测试

A.3.5.10 Tile的格式不正确

a）测试目的：检查对于每一个GetTile的格式，当Format参数设为该格式或URL模板包含格式参数时，服务器响应的MIME类型是否与其匹配。

b) 测试方法：如果对于每一个GetTile的格式，在面向过程架构风格下，当Format参数设为该格式时，服务器响应的MIME类型都与其匹配，或者在面向资源架构风格下，响应的URL模板格式与其都匹配，则通过。

c) 参照章节：7.2.1和10.2.1

d) 测试类型：能力测试

A.3.5.11 GetTile size

a) Test purpose: Verify that the returned tile has the correct size.

b) Test method: Send a correct request for each TileMatrix in the TileMatrixSet of a Layer and the test passes if width and height of the returned image are equal to the advertised values in TileWidth and TileHeight in the ServiceMetadata.

c) Reference: Subclause 7.2.1

d) Test type: Capability

A.3.5.12 GetTile transparent color

a) Test purpose: Verify that the returned tile has transparent color for NODATA values.

b) Test method: Send a correct request for a Layer in a format that supports transparency and in a tile where NODATA values are expected and test for transparent color there.

c) Reference: Subclause 7.2.1

d) Test type: Capability

A.3.6 FeatureInfo request test module

In order to pass this test module test A.3.5.1 to A.3.5.9 has to be also passed.

A.3.6.1 GetTileFeatureInfo on non-queryable layer

a) Test purpose: Verify that when a request to a non-queryable layer, then the server throws an exception.

b) Test method: When a GetFeatureInfo is requested on a Layer that not advertises any InfoFormat in the ServiceMetadata document, then the server throws an exception (code=OperationNotSupported) in Procedure Oriented Architectural Style

c) Reference: Subclause 7.3.2.2

d) Test type: Capability

A.3.5.2 FeatureInfo ResourceURL template

a) Test Purpose: Verify that a client supports URL templates and server satisfies RESTful requests

b) Test Method: Verify that the server provides a ServiceMetadata document that includes complete ResourceURL information with resourceType=FeatureInfo on Layer section if FeatureInfos of this layer are able for RESTful. Verify that the template processor in the client is able to convert the URL template in a correct URL to a FeatureInfo and the server is able to respond RESTful requests.

c) Reference: Subclause 10.2.1

d) Test Type: Capability

A.3.5.11 GetTile的大小

a）测试目的：检查返回的图块的大小是否正确。

b) 测试方法：针对某一图层的TileMatrixSet中的每个TileMatrix发送一个正确的请求，如果返回的图片的宽和高分别与服务元数据文档中告知的TileWidth和TileHeight的值相等，则通过。

c) 参照章节：7.2.1

d) 测试类型：能力测试

A.3.5.12 GetTile的透明色

a）测试目的：检查返回的图块中NODATA部分是否具有透明色。

b) 测试方法：针对某一图层，指定一个支持透明的格式，发送一个正确的请求，在一个包含NODATA值的图块中，检查NODATA值的地方是否表现为透明色。

c) 参照章节：7.2.1

d) 测试类型：能力测试

A.3.6 FeatureInfo请求测试模块

要通过本模块的测试，A.3.5.1和A.3.5.9的测试也必须通过。

A.3.6.1 向不可查询图层发起GetTileFeatureInfo请求

a）测试目的：检查如果向一个不可查询图层发起请求，服务器是否抛出异常。

b) 测试方法：如果向一个在ServiceMetadata文档中未告知任何InfoFormat的图层发起GetFeatureInfo请求，则在面向过程建构风格下服务器应抛出一个异常（code=OperationNotSupported）。

c) 参照章节：7.3.2.2

d) 测试类型：能力测试

A.3.5.2 FeatureInfo的ResourceURL模板

a）测试目的：检查客户端是否支持URL模板，服务器端是否满足REST风格请求的要求。

b) 测试方法：检查服务器是否提供了一个在Layer子集中包含属性resourceType等于FeatureInfo的完整的ResourceURL信息的服务元数据文档，而且该图层的所有FeatureInfo都是可以通过REST风格方式获取到。检查客户端的模板处理器是否能够把URL模板转换成指向图块资源的正确URL，而且服务器是否能够对REST风格的请求作出响应。

c) 参照章节：10.2.1

d) 测试类型：能力测试

A.3.6.2 GetFeatureInfo incorrect InfoFormat

a) Test purpose: Verify that when a request contains an InfoFormat incorrect value, then the server throws an exception.

b) Test method: When a request contains an InfoFormat value that is not advertised for this Layer in the ServiceMetadata document, then the server throws an exception (code= InvalidParameterValue) in Procedure Oriented Architectural Style.

c) Reference: Subclause 7.3.2.2

d) Test type: Capability

A.3.6.3 FeatureInfo correct InfoFormat

a) Test purpose: Verify that for each GetTile format, when the InfoFormat parameter is set to that format or URLtemplate having a format parameter, the MIME type of the response matches that format.

b) Test method: Pass if for each GetTile format, when the InfoFormat parameter is set to that format, the MIME type of the response matches that format or the URL template format of the response matches that format in Resource Oriented Architectural Style

c) Reference: Subclause 7.3.1 and 10.3.1

d) Test type: Capability

A.3.6.4FeatureInfo J and I

a) Test purpose: Verify that when a request contains an I or J incorrect, then the server throws an exception.

b) Test method: When a request contains an I or J greater or equal to TileHeight and TileWidth respectively for the selected TileMatrix, then the server throws an exception (code=PointIJOutOfRange) in Procedure Oriented Architectural Style or a HTTP 404 File not found in Resource Oriented Architectural Style.

c) Reference: Subclause 7.3.2.2

d) Test type: Capability

A.3.6.2 GetFeatureInfo请求包含不正确的InfoFormat

a）测试目的：检查如果请求中包含一个InfoFormat的不正确取值，服务器是否抛出一个异常。

b) 测试方法：如果请求中包含一个在ServiceMetadata 文档中未告知的InfoFormat取值，面向过程架构风格下服务器应抛出一个异常（code= InvalidParameterValue）。

c) 参照章节：7.3.2.2

d) 测试类型：能力测试

A.3.6.3 FeatureInfo具有正确的InfoFormat

a）测试目的：检查对于每一个GetTile的格式，当InfoFormat参数设为该格式或URL模板包含格式参数时，服务器响应的MIME类型是否与其匹配。

b) 测试方法：如果对于每一个GetTile的格式，当InfoFormat参数设为该格式时，服务器响应的MIME类型都与其匹配，或者在面向资源架构风格下，响应的URL模板格式与其都匹配，则通过。

c) 参照章节：7.3.1和10.3.1

d) 测试类型：能力测试

A.3.6.4 位于（J，I）的 FeatureInfo

a）测试目的：检查如果请求中包含一个I或J的不正确取值，服务器是否抛出一个异常。

b) 测试方法：如果请求中的I或J取值分别大于或等于所选TileMatrix的TileHeight和TileWidth，面向过程架构风格下服务器应抛出一个异常（code= PointIJOutOfRange），或者面向资源架构风格下返回一个HTTP 404（文件未找到）的错误。

c) 参照章节：7.3.2.2

d) 测试类型：能力测试

Annex B (normative) **XML Schema Documents**

In addition to this document, this standard includes several normative XML Schema Documents. These XML Schema Documents may be bundled in a zip file with the present document. After OGC acceptance of a Version 1.0.0 of this standard, these XML Schema Documents will also be posted online at the URL http://schemas.opengis.net/wmts/1.0.0. In the event of a discrepancy between the bundled and online versions of the XML Schema Documents, the online files SHALL be considered authoritative.

The WMTS abilities specified in this document use one specified XML Schema Documents included in the zip file with this document. These XML Schema Documents combine the XML schema fragments listed in various subclauses of this document, eliminating duplications. These XML Schema Documents are named:

wmtsGetCapabilities_request.xsd

wmtsGetCapabilities_response.xsd

wmtsGetTile_request.xsd

wmtsGetFeatureInfo_request.xsd

wmtsGetFeatureInfo_response.xsd

wmtsPayload_response.xsd

In addition, the following XML Schema Documents imported by WSDL documents are used in the annex F and are also included:

wmts.xsd

wmtsKVP.xsd

These XML Schema Documents use and build on the OWS common XML Schema Documents specified OWS Common [OGC 06-121r3], named:

ows19115subset.xsd

owsCommon.xsd

owsDataIdentification.xsd

owsExceptionReport.xsd

owsGetCapabilities.xsd

owsOperationsMetadata.xsd

owsServiceIdentification.xsd

owsServiceProvider.xsd

All these XML Schema Documents contain documentation of the meaning of each element and attribute, and this documentation SHALL be considered normative as specified in subclause 11.6.3 of OWS Common [OGC 06-121r3].

Also complete examples for KVP, SOAP and RESTful can be found on the zip file and on the portal. Some fragments of these examples are shown throughout this document.

附录B（规范性）XML模式文档

除本文档外，该标准包括几个规范性的XML模式文档。这些XML模式文档可以与本文档打包成一个zip文件。在本标准的1.0.0版本被OGC接受后，这些XML模式文档也将放到网上，URL地址为http://schemas.opengis.net/wmts/1.0.0。如果XML模式文档的打包文件和在线版本有差异，则在线文件应视为权威版本。

本规范使用包含在与本文档一起的ZIP文件中的一系列指定的XML模式文档来详细说明WMTS的功能。这些XML模式文档把本文档中各个章节中列出的XML模式片段除去重复部分合在了一起。这些XML模式文档名如下：

wmtsGetCapabilities_request.xsd

wmtsGetCapabilities_response.xsd

wmtsGetTile_request.xsd

wmtsGetFeatureInfo_request.xsd

wmtsGetFeatureInfo_response.xsd

wmtsPayload_response.xsd

此外，以下在附录F中使用过的从WSDL文档中导入的XML模式文档也包含在内：

wmts.xsd

wmtsKVP.xsd

以下XML模式文档使用并建立在OWS通用实现规范中规定的XML模式文档之上：

ows19115subset.xsd

owsCommon.xsd

owsDataIdentification.xsd

owsExceptionReport.xsd

owsGetCapabilities.xsd

owsOperationsMetadata.xsd

owsServiceIdentification.xsd

owsServiceProvider.xsd

所有这些XML模式文档包含关于每个元素和属性含义的注释信息，而且本文档将按照OWS通用实现规范[OGC 06-121r3]11.6.3的要求被视作规范性的内容。

针对KVP、SOAP和REST风格接口的完整实例也可以在门户上的zip文件中找到。一些实例片段已在本文档中示出。

Annex C (informative) UML model

C.1 Introduction

This annex provides a UML model of the WMTS interface, using the OGC/ISO profile of UML summarized in subclause 5.2 of [06-121r3].

Figure C.1 is a simple UML diagram summarizing the WMTS interface. This class diagram shows that the WMTS class inherits the getCapabilities operation from the OGCWebService interface class, and adds the getTile and getFeatureInfo operations. (The capitalization of names uses the OGC/ISO profile of UML.)

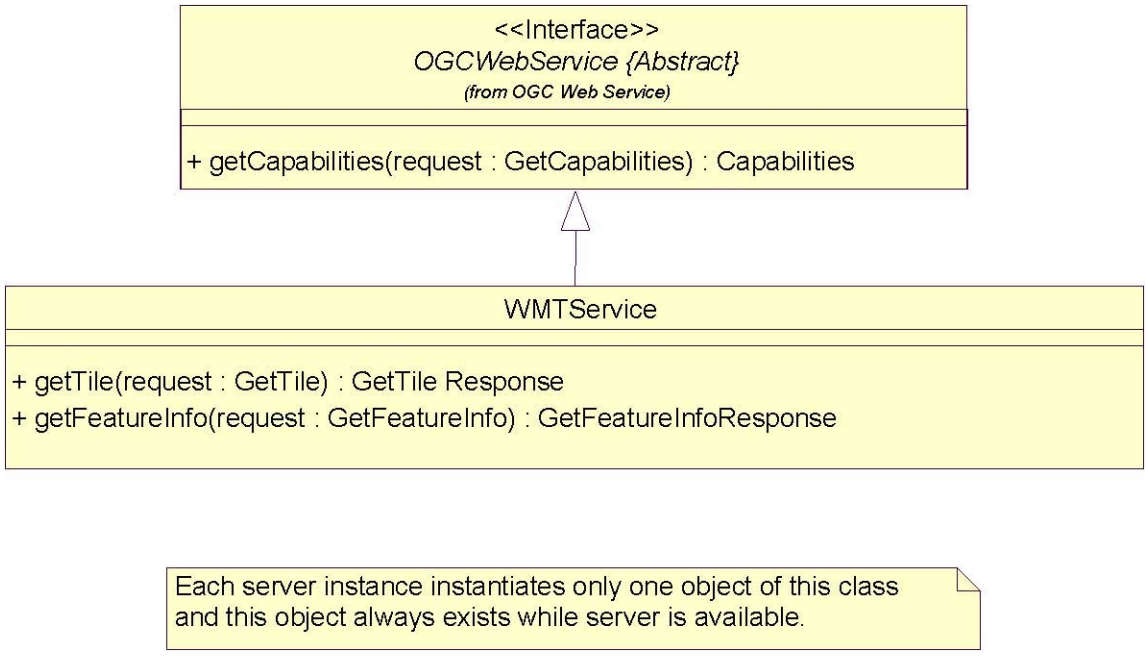

Figure C.1 — WMTS interface UML diagram

Each of the three operations uses a request and a response data type, each of which is defined by one or more additional UML classes. The following subclauses provide a more complete UML model of the WMTS interface, adding UML classes defining the operation request and response data types.

C.2 UML packages

The WMTS interface UML model is organized in six packages that will be described in the following subclauses. These six WMTS-specific packages make use of six non-WMTS-specific packages, named OWS Web Service, OWS Operations Metadata, OWS Service Identification, OWS Service Provider, and ISO 19115 Subset. This package diagram shows the dependencies among the various packages shown.

附录C （资料性）UML模型

C.1 引言

本附录给出了WMTS接口的UML模型，使用的是OGC/ISO的UML方案，该方案在[06-121r3]的5.2中进行了总结。

图C.1是WMTS接口的概要UML图。该类图表明WMTS接口从OGCWebService接口类继承了getCapabilities操作，增加了getTile和getFeatureInfo操作（名称的大小写方法依据OGC/ISO的UML方案）。

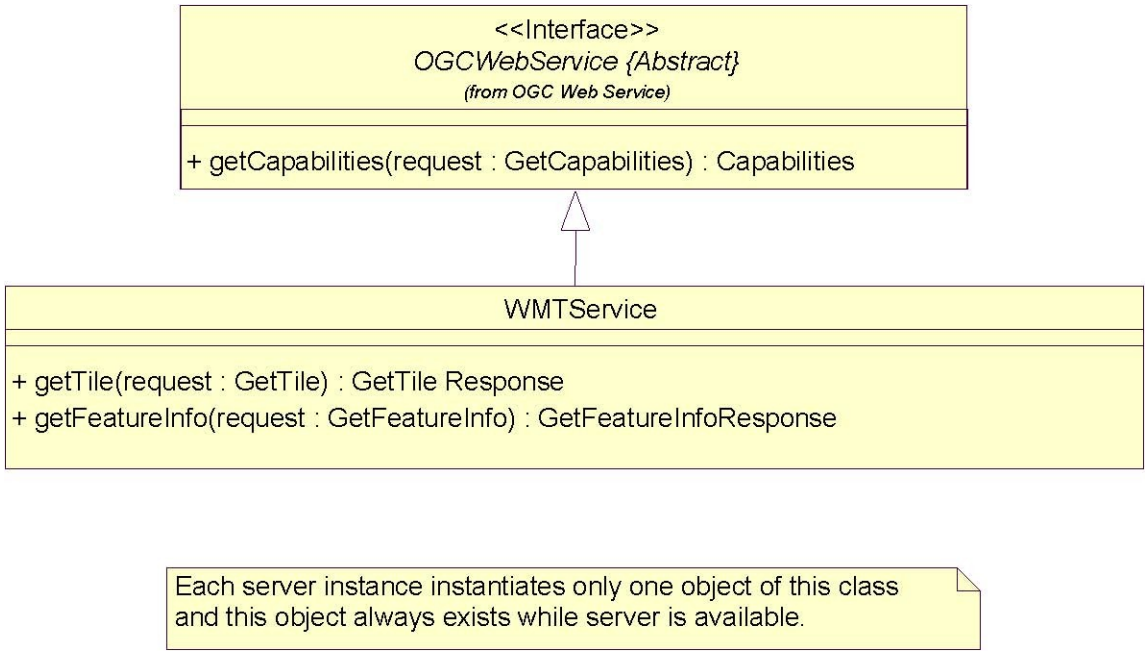

图 C.1 — WMTS 接口的 UM 图

这三个操作每个都分别使用一个请求和响应数据类型，每个数据类型由一或多个UML类定义。下面各章节将给出WMTS接口更完整的UML模型，增加了定义操作请求与响应数据类型的UML类。

C.2 UML 包

WMTS接口的UML模型被分为6个包在后面各章节中进行说明。这6个针对WMTS的包用到了其他6个非WMTS的包，分别是OWS Web Service, OWS Operations Metadata, OWS Service Identification, OWS Service Provider, 以及ISO 19115 子集。下图显示了各包之间的依赖关系。

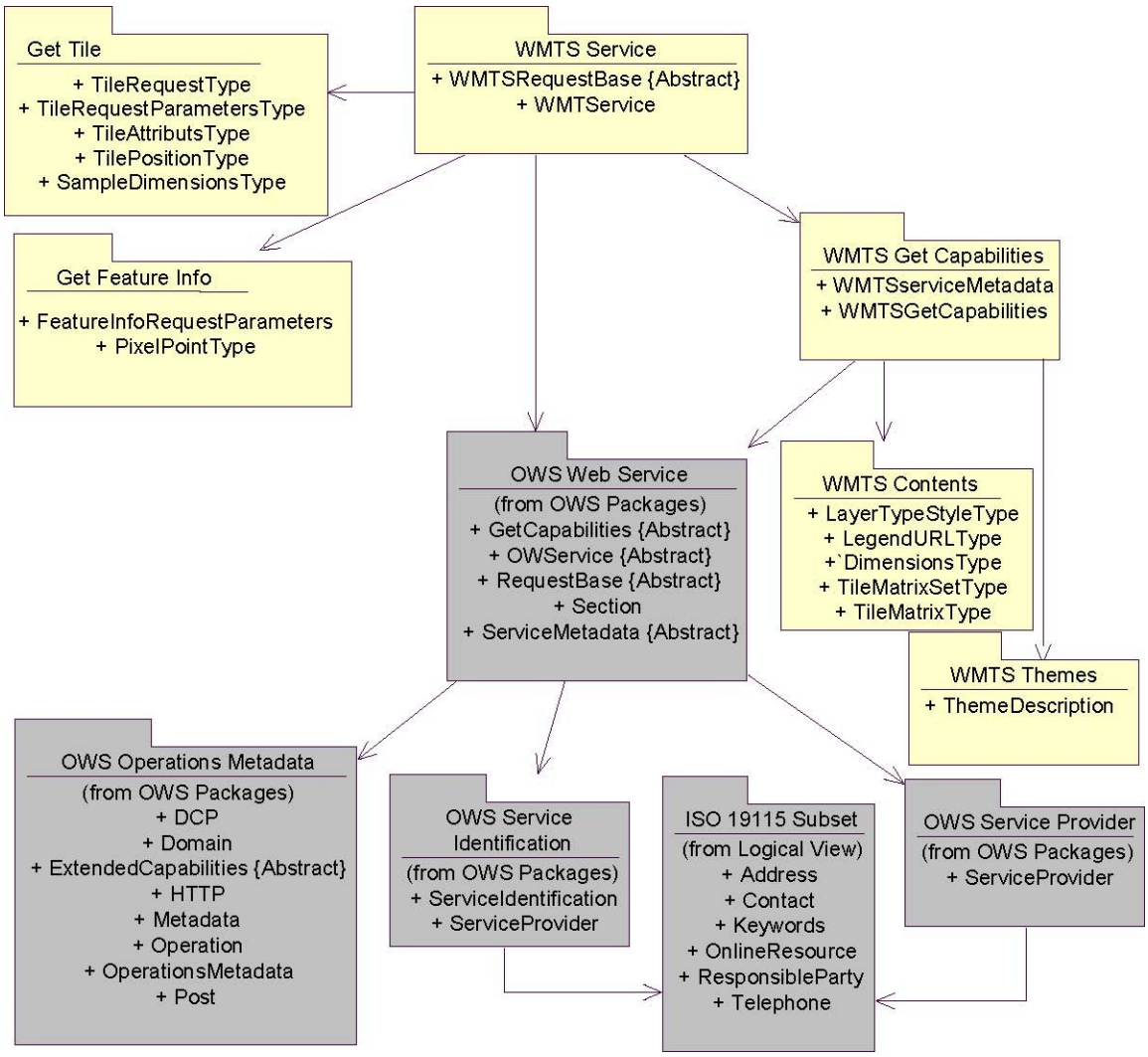

Figure C.2 — WMTS interface package diagram

Each of the three WMTS-specific packages shown in Figure C.2 is described in the following subclauses. The OWS Web Service, OWS Operations Metadata, OWS Service Identification, OWS Service Provider, and ISO 19115 Subset packages are described in the Annex C of OWS Common [OGC 06-121r3].

C.3 WMTS Service package

The WMTS Service package is shown in the class diagram in Figure C.3. This diagram does not show the classes used by the WMTS operation requests and responses, which are shown (with part of this package) in the WMTS Get Capabilities, Get Tile and Get Feature Info packages. This diagram also shows RequestBase used classes from the OWS Web Service package, which is common to all OGC Web Services, plus one used class from the WMTS package.

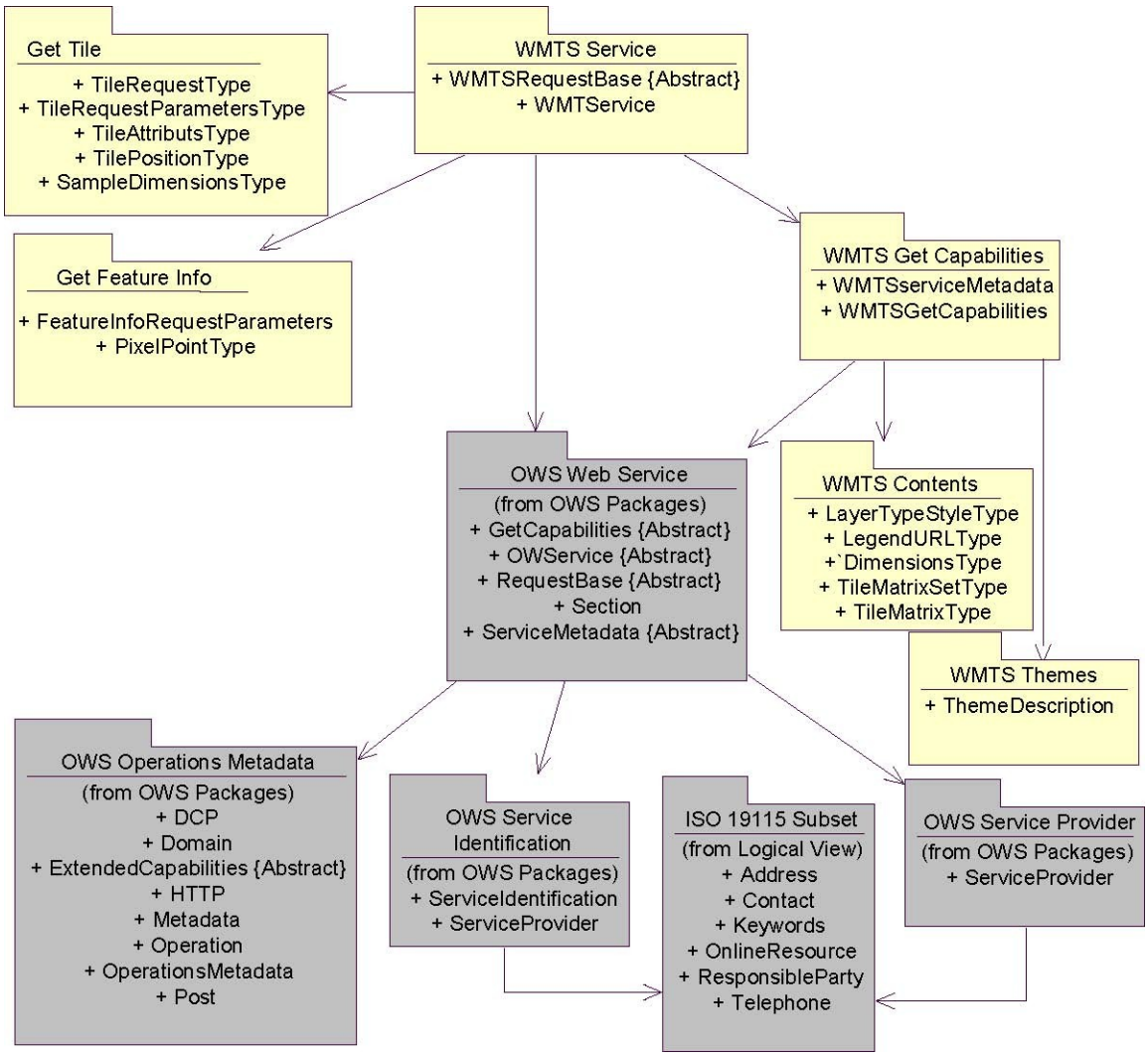

图 C.2 — WMTS 接口包图

图C.2所示的这三个针对WMTS接口的包将在后续章节中说明。其他包，包括OWS Web Service, OWS Operations Metadata, OWS Service Identification, OWS Service Provider, 和ISO 19115 子集等在OWS通用实现规范[OGC 06-121r3]的附录C中说明。

C.3 WMTS服务包

WMTS服务接口包如图C.3所示。该图没有表示WMTS操作请求和相应使用的类，它们在WMTS Get Capabilities, Get Tile 和Get Feature Info包中表示。与所有OGC的Web服务一样，该图同时显示了RequestBase使用的OWS Web Service包中的类，加上一个用到的WMTS包中的类。

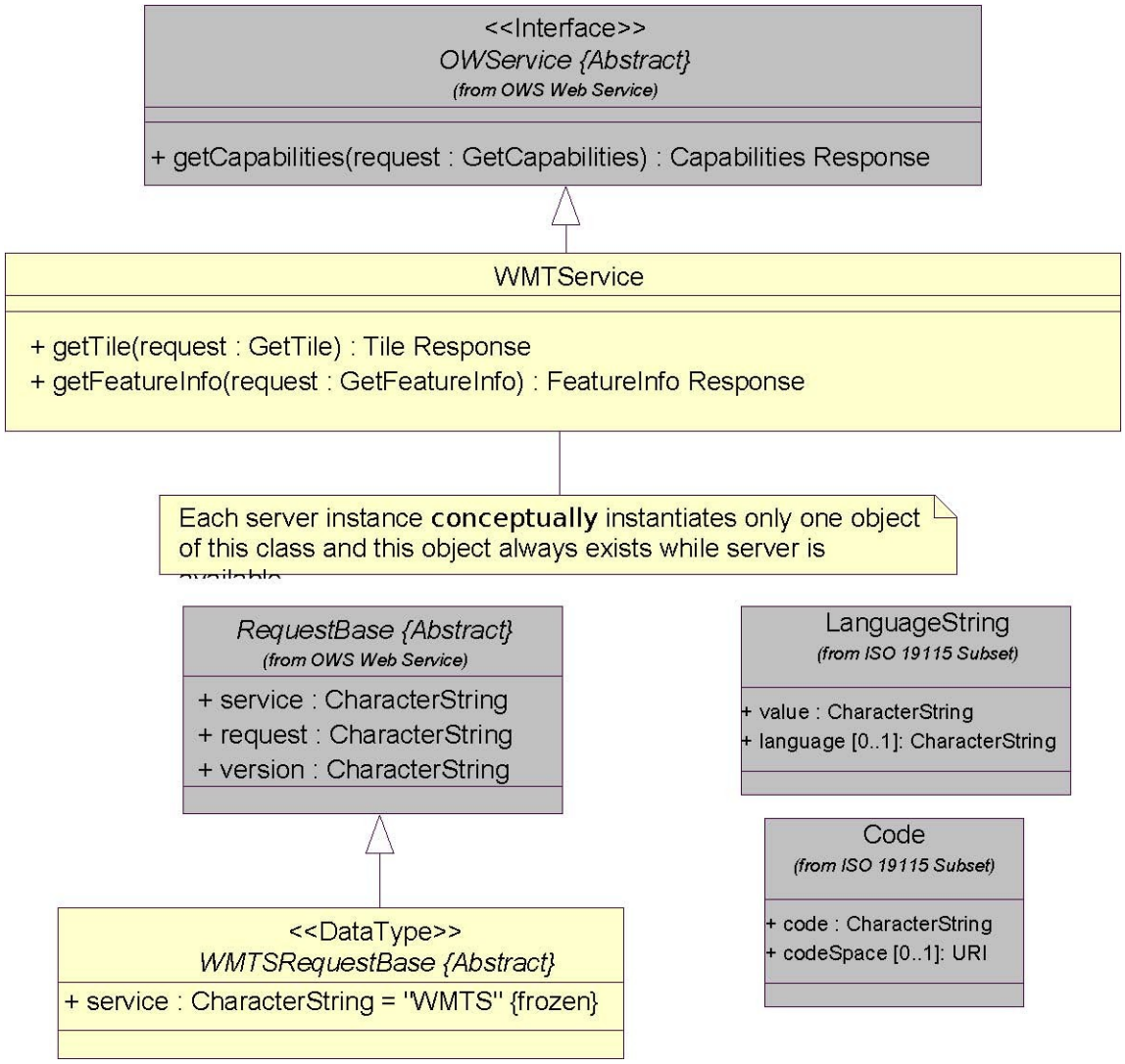

Figure C.3 — WMTS Service package class diagram

C.4 WMTS Get Capabilities, Contents and Themes packages

The WMTS Get Capabilities package is shown in the class diagram in Figure C.4, C.5 and C.6. This diagram also shows several classes from the OWS package. The classes introduced by this package are further defined by Table 2 through Table 16 in this document.

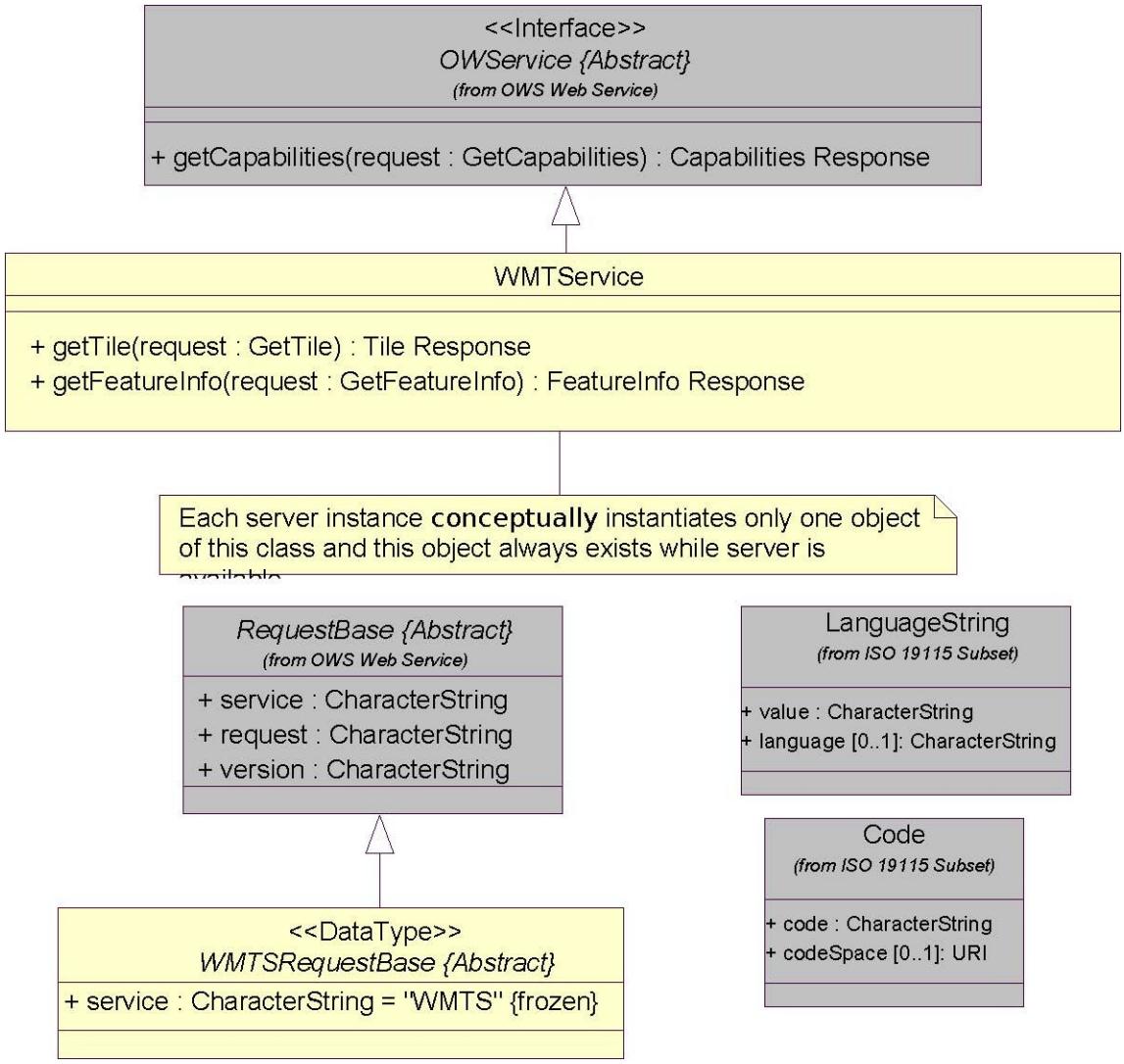

图 C.3 — WMTS 服务包的类图

C.4 WMTS GetCapabilities, Contents 和 Themes 包

图C.4，C.5和C.6表示了WMTS Get Capabilities包。该图还表示了OWS包中的几个类。该包引入的类在本文档的表2至表16进行了详细定义。

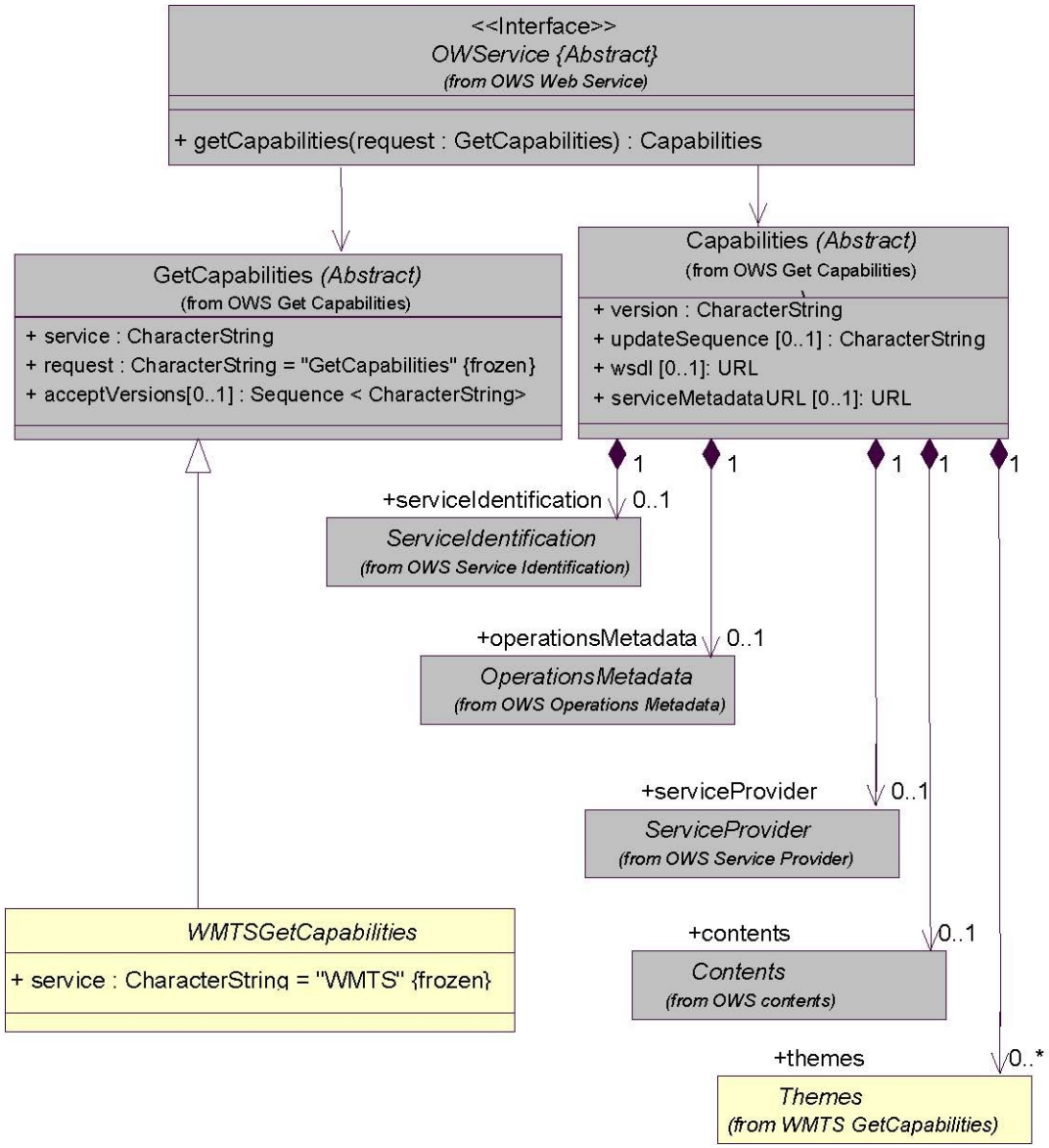

Figure C.4 — Get Capabilities package class diagram, part 1

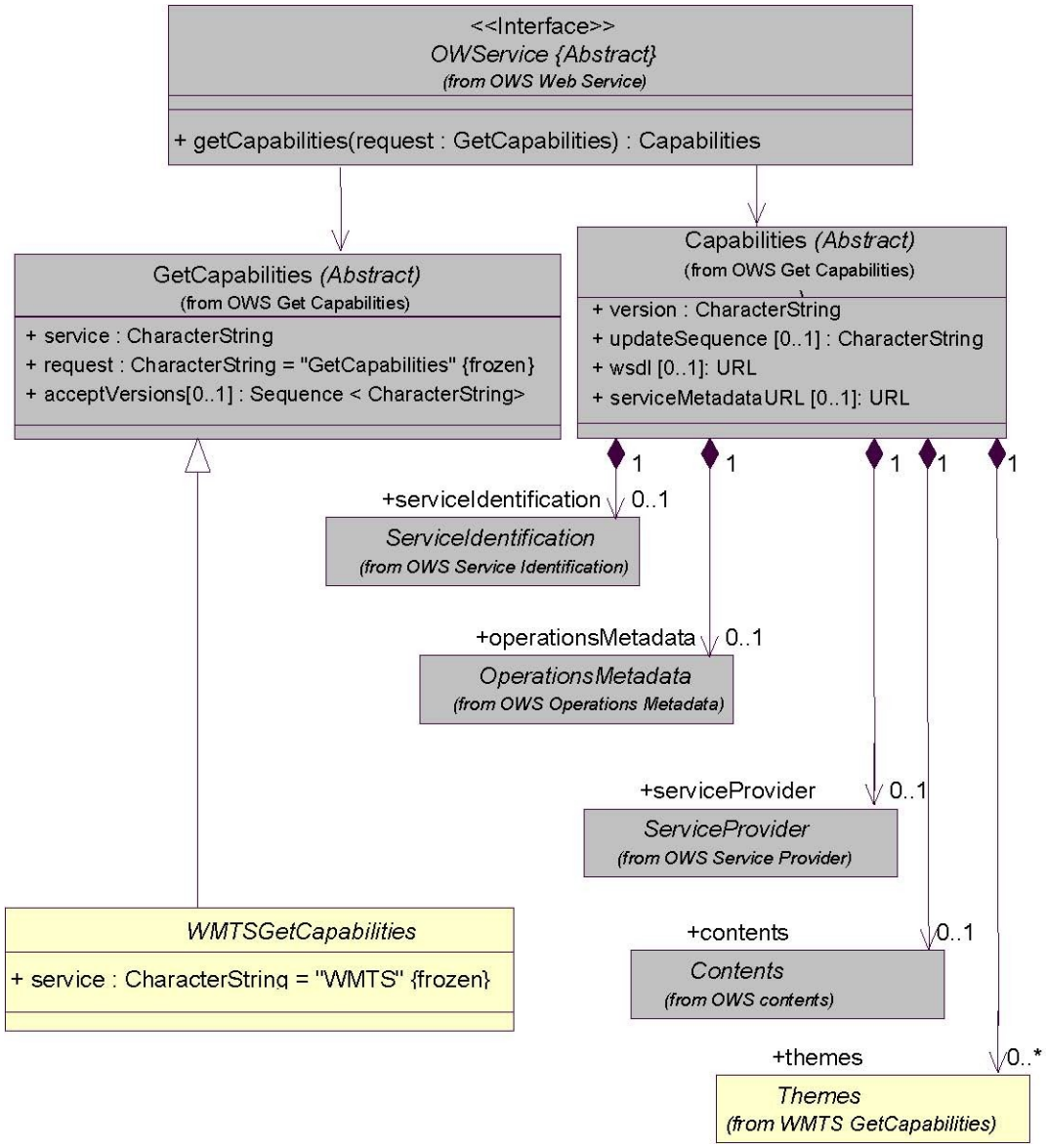

图 C.4 — Get Capabilities 包的类图,第 1 部分

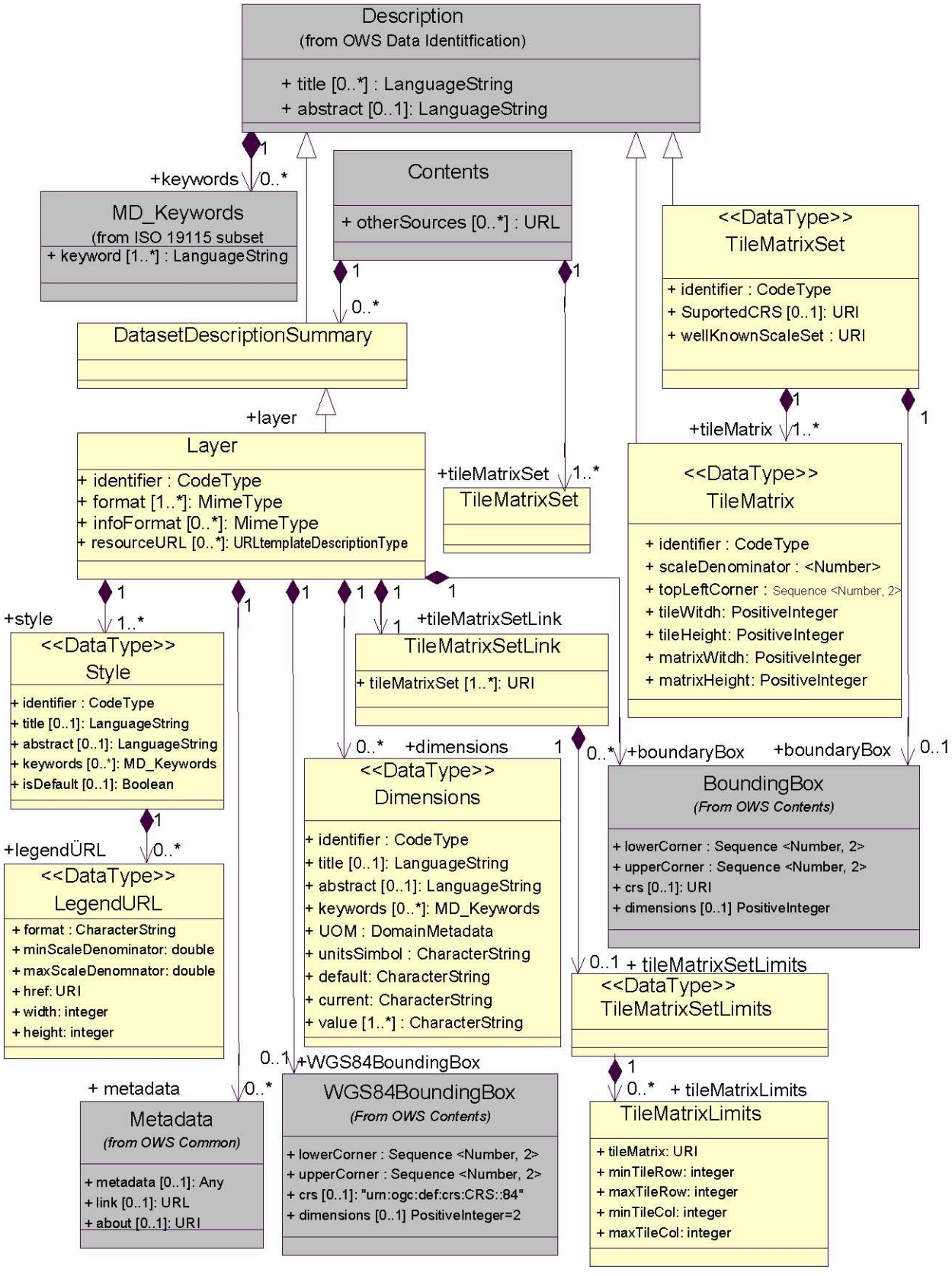

Figure C.5 — Get Capabilities package class diagram, part 2

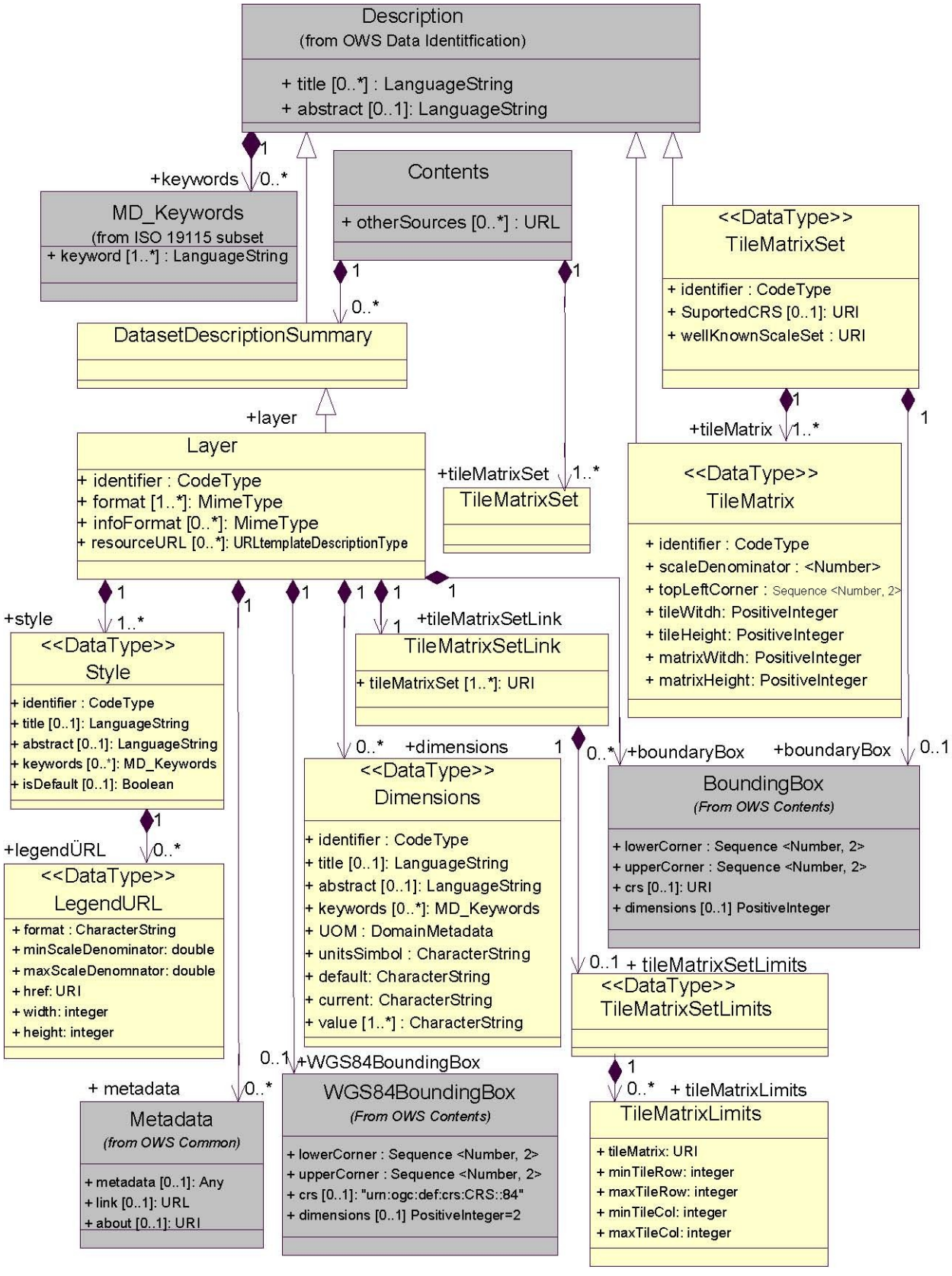

Figure C.5 — Get Capabilities Get Capabilities 包的类图,第 2 部分

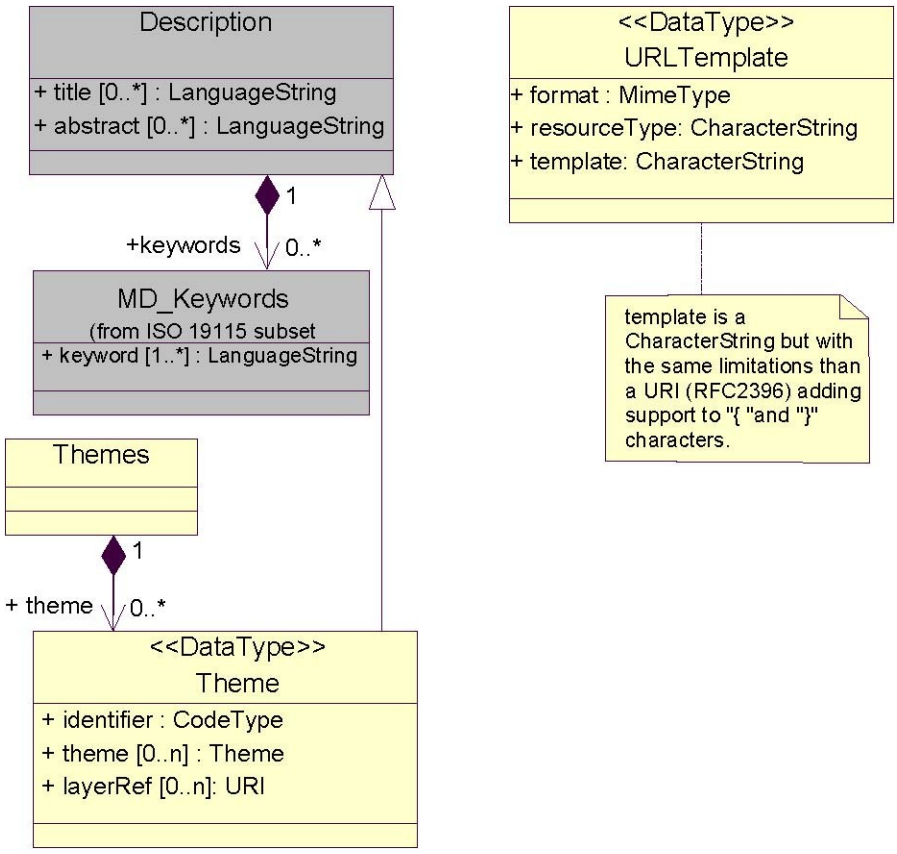

Figure C.6 — Get Capabilities package class diagram, part 3

C.5 WMTS GetTile package

The WMTS Get Tile package is shown in the class diagram in Figure C.7. This diagram also shows several classes from the OWS package. The classes introduced by this package are further defined by Table 22 in this document.

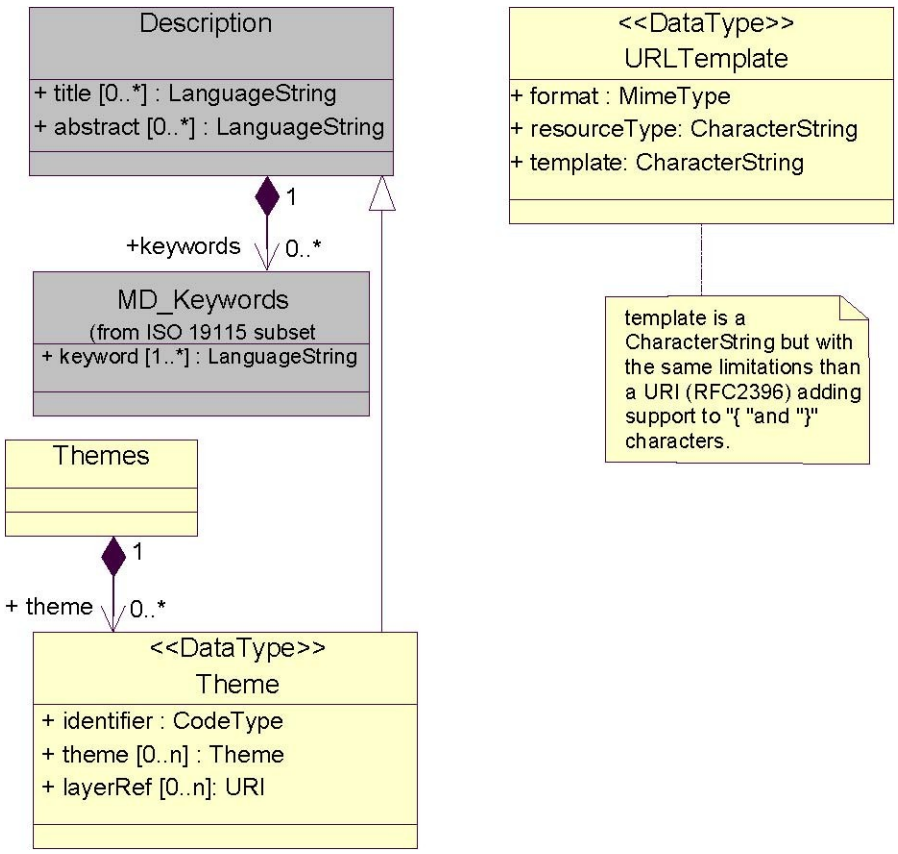

图 C.6 —Get Capabilities 包的类图,第 3 部分

C.5 WMTS GetTile 包

图C.7表示了WMTS Get Tile包。该图还表示了OWS包中的几个类。该包引入的类在本文档的表22进行了详细定义。

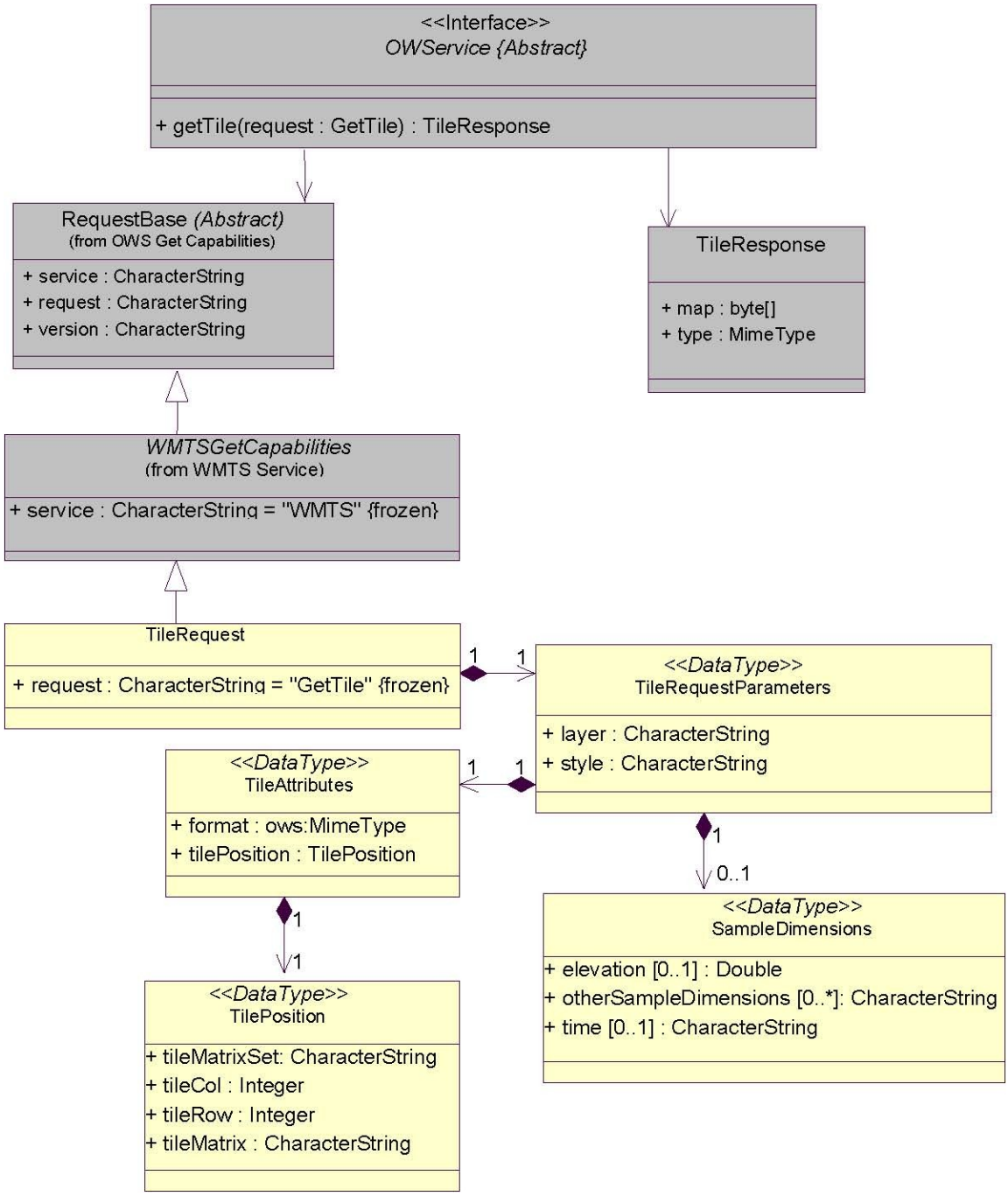

Figure C.7 — Get Tile package class diagram

Table C.1 — Mapping of UML TileRequest attributes and HTTP GetTile request parameters

UML attribute	HTTP request parameters
service	`Service`
request	`Request`
version	`Version`
layer	`Layer`
style	`Style`
tileAttributes	`TileMatrixSet, TileMatrix, TileRow, TileCol, Format`
sampleDimensions	`Time, Elevation, SampleDimensions`

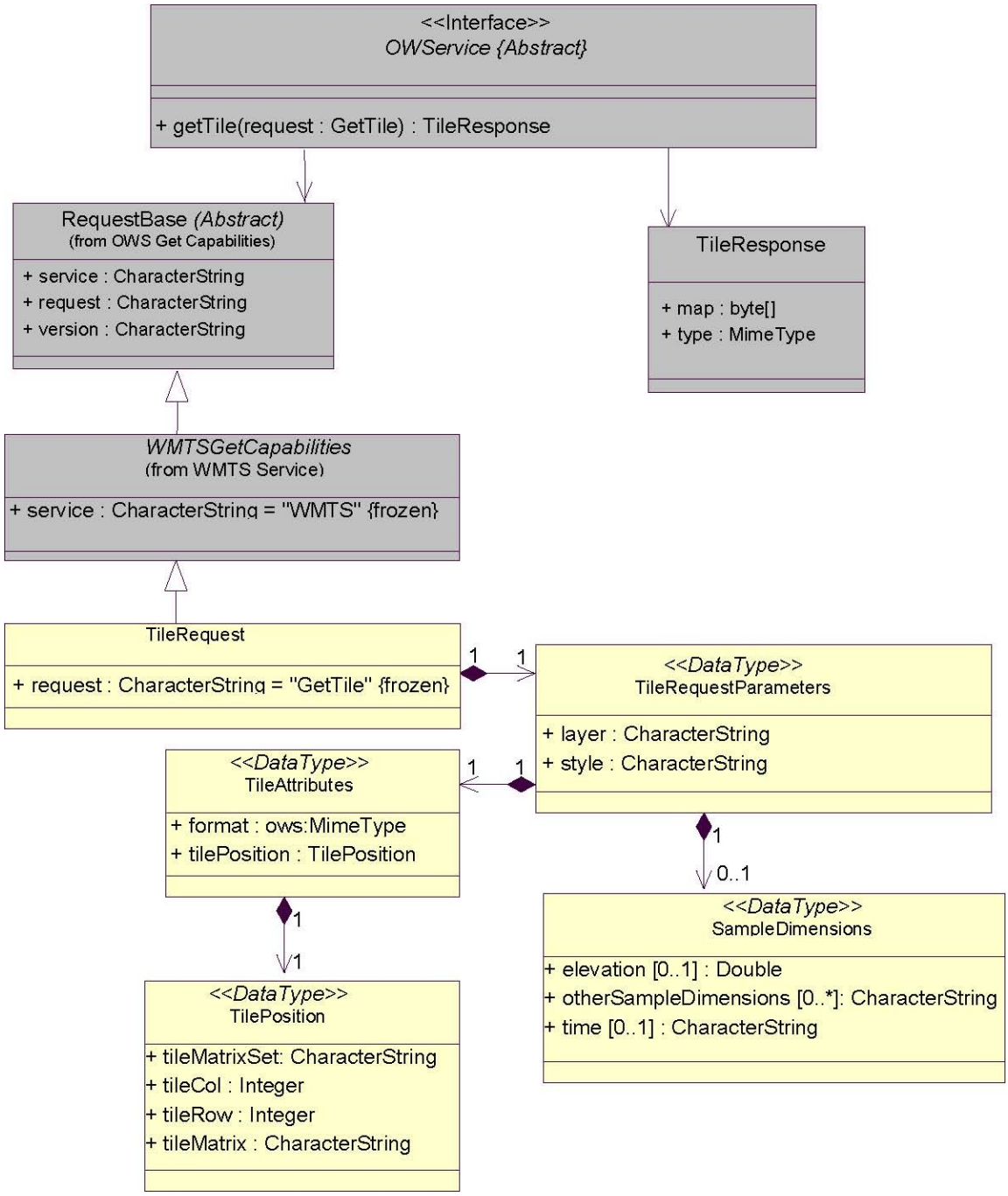

图 C.7 — Get Tile 包的类图

表 C.1 —UML TileRequest 属性与 HTTP GetTile 请求之间的对应关系

UML 属性	HTTP请求参数
service	Service
request	Request
version	Version
layer	Layer
style	Style
tileAttributes	TileMatrixSet, TileMatrix, TileRow, TileCol, Format
sampleDimensions	Time, Elevation, SampleDimensions

C.6 WMTS GetFeatureInfo package

The WMTS Get Feature Info package is shown in the class diagram in Figure C.8. This diagram also shows several classes from the OWS and WMTS packages. The classes introduced by this package are further defined by Table 25 in this document.

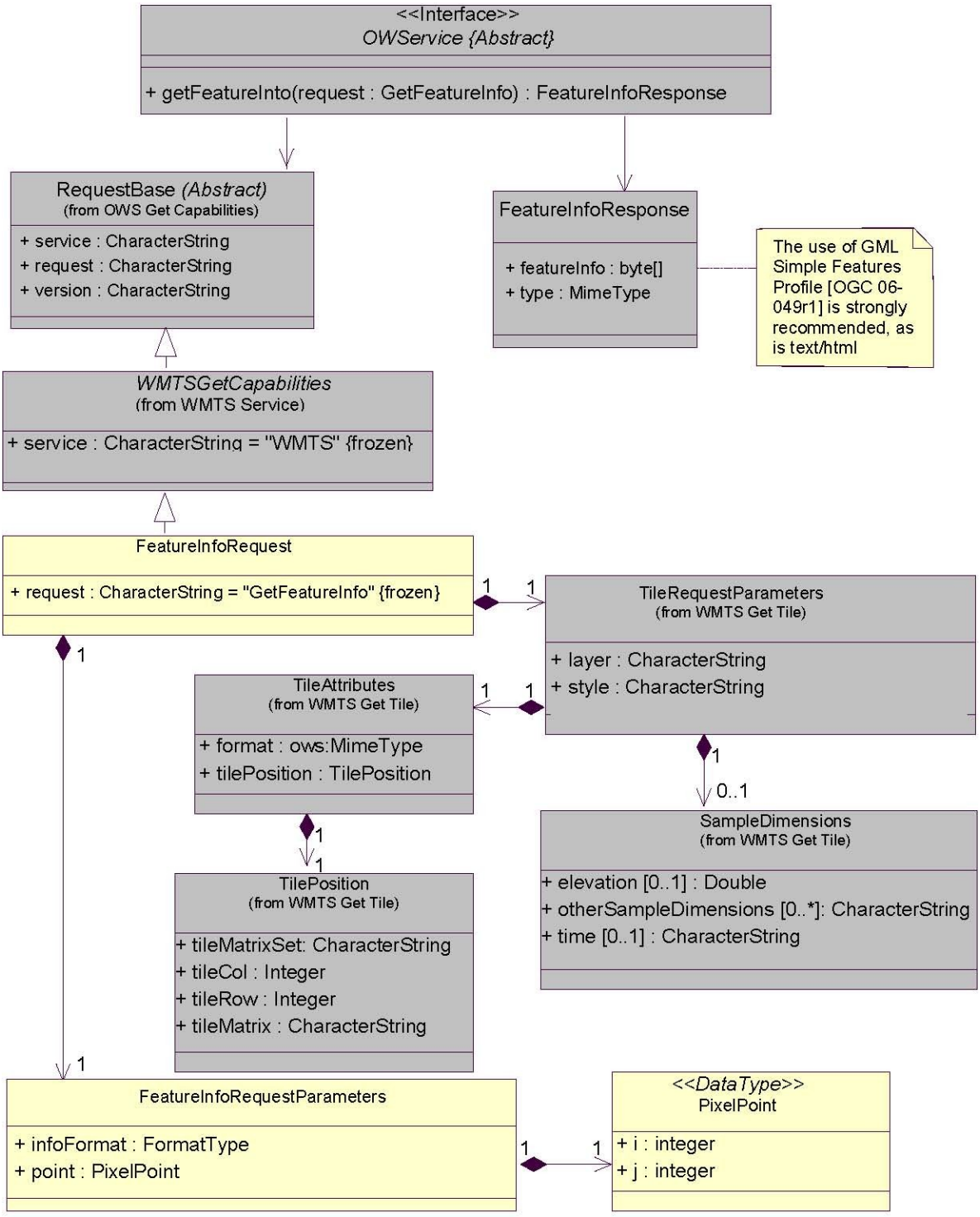

Figure C.8 — Get Feature Info package class diagram

C.6 WMTS GetFeatureInfo 包

图C.8表示了WMTS Get Feature Info包。该图还表示了OWS包中的几个类。该包引入的类在本文档的表25进行了详细定义。

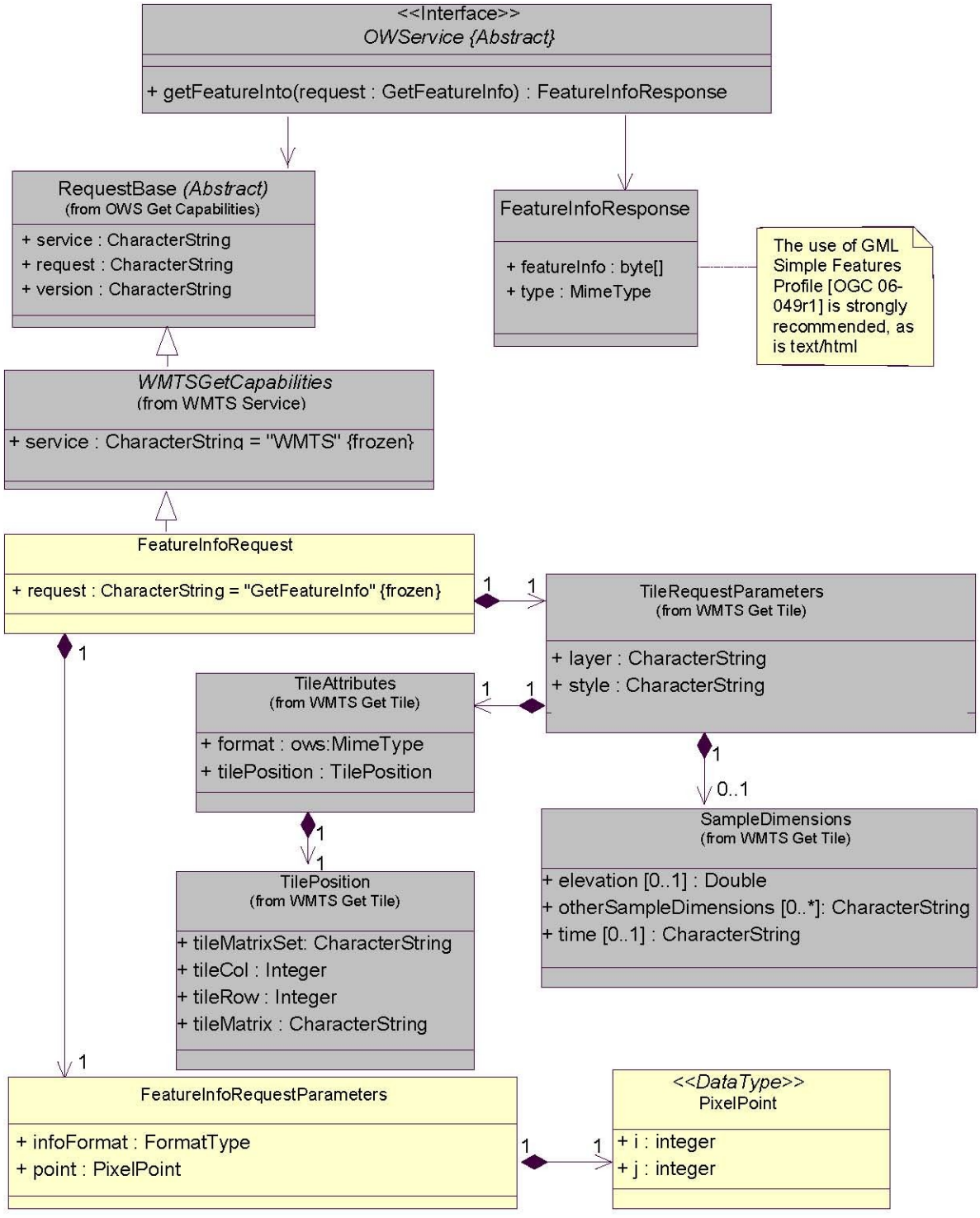

图 C.8 — Get Feature Info 包的类图

Table C.2 — Mapping of UML FeatureInfoRequest attributes and HTTP GetFeatureInfo request parameters

UML attribute	HTTP request parameters
service	`Service`
request	`Request`
version	`Version`
layer	`Layer`
style	`Style`
tileAttributes	`TileMatrixSet, TileMatrix, TileRow, TileCol, Format`
sampleDimensions	`Time, Elevation, SampleDimensions`
infoFormat	`InfoFormat`
point	`J, I`

表C.2 —UML FeatureInfoRequest属性与HTTP GetFeatureInfo请求之间的对应关系

UML 属性	HTTP请求参数
service	Service
request	Request
version	Version
layer	Layer
style	Style
tileAttributes	TileMatrixSet, TileMatrix, TileRow, TileCol, Format
sampleDimensions	Time, Elevation, SampleDimensions
infoFormat	InfoFormat
point	J, I

Annex D (informative) Example XML documents

D.1 Introduction

This annex provides more example XML documents than given in the body of this document. Particularly it includes an XML example for RESTful approach.

D.2 ServiceMetadata response example

This is a complete example of ServiceMetadata response with RESTful and KVP support.

```
<?xml version="1.0" encoding="UTF-8"?>
<Capabilities xmlns="http://www.opengis.net/wmts/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:gml="http://www.opengis.net/gml"
           xsi:schemaLocation="http://www.opengis.net/wmts/1.0
http://schemas.opengis.net/wmts/1.0.0/wmtsGetCapabilities_response.xsd"
version="1.0.0">
 <ows:ServiceIdentification>
  <ows:Title>Web Map Tile Service</ows:Title>
  <ows:Abstract>Service that contrains the map access interface to some
TileMatrixSets</ows:Abstract>
  <ows:Keywords>
   <ows:Keyword>tile</ows:Keyword>
   <ows:Keyword>tile matrix set</ows:Keyword>
   <ows:Keyword>map</ows:Keyword>
  </ows:Keywords>
  <ows:ServiceType>OGC WMTS</ows:ServiceType>
  <ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
  <ows:Fees>none</ows:Fees>
  <ows:AccessConstraints>none</ows:AccessConstraints>
 </ows:ServiceIdentification>
 <ows:ServiceProvider>
  <ows:ProviderName>MiraMon</ows:ProviderName>
  <ows:ProviderSite xlink:href="http://www.creaf.uab.cat/miramon"/>
  <ows:ServiceContact>
   <ows:IndividualName>Joan Maso Pau</ows:IndividualName>
   <ows:PositionName>Senior Software Engineer</ows:PositionName>
   <ows:ContactInfo>
    <ows:Phone>
     <ows:Voice>+34 93 581 1312</ows:Voice>
     <ows:Facsimile>+34 93 581 4151</ows:Facsimile>
    </ows:Phone>
    <ows:Address>
     <ows:DeliveryPoint>Fac Ciencies UAB</ows:DeliveryPoint>
     <ows:City>Bellaterra</ows:City>
     <ows:AdministrativeArea>Barcelona</ows:AdministrativeArea>
     <ows:PostalCode>08193</ows:PostalCode>
     <ows:Country>Spain</ows:Country>
     <ows:ElectronicMailAddress>joan.maso@uab.cat</ows:ElectronicMailAddress>
    </ows:Address>
   </ows:ContactInfo>
```

附录D（资料性）XML文档实例

D.1 引言

本附录提供了比标准正文中更多的XML文档实例，特别是包含了一个REST风格下的XML实例。

D.2 ServiceMetadata响应实例

以下是一个完整的REST风格、支持KVP的服务元数据文档实例。

```
<?xml version="1.0" encoding="UTF-8"?>
<Capabilities xmlns="http://www.opengis.net/wmts/1.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:gml="http://www.opengis.net/gml"
           xsi:schemaLocation="http://www.opengis.net/wmts/1.0
http://schemas.opengis.net/wmts/1.0.0/wmtsGetCapabilities_response.xsd"
version="1.0.0">
 <ows:ServiceIdentification>
  <ows:Title>Web Map Tile Service</ows:Title>
  <ows:Abstract>Service that contrains the map access interface to some
TileMatrixSets</ows:Abstract>
  <ows:Keywords>
   <ows:Keyword>tile</ows:Keyword>
   <ows:Keyword>tile matrix set</ows:Keyword>
   <ows:Keyword>map</ows:Keyword>
  </ows:Keywords>
  <ows:ServiceType>OGC WMTS</ows:ServiceType>
  <ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
  <ows:Fees>none</ows:Fees>
  <ows:AccessConstraints>none</ows:AccessConstraints>
 </ows:ServiceIdentification>
 <ows:ServiceProvider>
  <ows:ProviderName>MiraMon</ows:ProviderName>
  <ows:ProviderSite xlink:href="http://www.creaf.uab.cat/miramon"/>
  <ows:ServiceContact>
   <ows:IndividualName>Joan Maso Pau</ows:IndividualName>
   <ows:PositionName>Senior Software Engineer</ows:PositionName>
   <ows:ContactInfo>
    <ows:Phone>
     <ows:Voice>+34 93 581 1312</ows:Voice>
     <ows:Facsimile>+34 93 581 4151</ows:Facsimile>
    </ows:Phone>
    <ows:Address>
     <ows:DeliveryPoint>Fac Ciencies UAB</ows:DeliveryPoint>
     <ows:City>Bellaterra</ows:City>
     <ows:AdministrativeArea>Barcelona</ows:AdministrativeArea>
     <ows:PostalCode>08193</ows:PostalCode>
     <ows:Country>Spain</ows:Country>
     <ows:ElectronicMailAddress>joan.maso@uab.cat</ows:ElectronicMailAddress>
    </ows:Address>
   </ows:ContactInfo>
```

```
  </ows:ServiceContact>
 </ows:ServiceProvider>
 <ows:OperationsMetadata>
  <ows:Operation name="GetCapabilities">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/cgi-bin/MiraMon5_0.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Get>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
  <ows:Operation name="GetTile">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/cgi-bin/MiraMon5_0.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Get>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
 </ows:OperationsMetadata>
 <Contents>
  <Layer>
   <ows:Title>Blue Marble Next Generation</ows:Title>
   <ows:Abstract>Blue Marble Next Generation NASA Product</ows:Abstract>
   <ows:WGS84BoundingBox>
    <ows:LowerCorner>-180 -90</ows:LowerCorner>
    <ows:UpperCorner>180 90</ows:UpperCorner>
   </ows:WGS84BoundingBox>
   <ows:Identifier>BlueMarbleNextGeneration</ows:Identifier>
   <Style isDefault="true">
    <ows:Identifier>Default</ows:Identifier>
   </Style>
   <Format>image/jpeg</Format>
   <TileMatrixSetLink>
    <TileMatrixSet>BigWorldPixel</TileMatrixSet>
   </TileMatrixSetLink>
   <ResourceURL format="image/png" resourceType="tile"
template="http://www.maps.bob/wmts/BlueMarbleNextGeneration/
default/BigWorldPixel/{TileMatrix}/{TileRow}/{TileCol}.png"/>
  </Layer>
  <TileMatrixSet>
   <ows:Identifier>BigWorldPixel</ows:Identifier>
   <ows:SupportedCRS>urn:ogc:def:crs:OGC:1.3:CRS84</ows:SupportedCRS>

<WellKnownScaleSet>urn:ogc:def:wkss:OGC:1.0:GlobalCRS84Pixel</WellKnownScaleSe
t>
   <TileMatrix>
    <ows:Identifier>10000m</ows:Identifier>
    <ScaleDenominator>33130800.83133142</ScaleDenominator>
```

```
  </ows:ServiceContact>
 </ows:ServiceProvider>
 <ows:OperationsMetadata>
  <ows:Operation name="GetCapabilities">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/cgi-bin/MiraMon5_0.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Get>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
  <ows:Operation name="GetTile">
   <ows:DCP>
    <ows:HTTP>
     <ows:Get xlink:href="http://www.maps.bob/cgi-bin/MiraMon5_0.cgi?">
      <ows:Constraint name="GetEncoding">
       <ows:AllowedValues>
        <ows:Value>KVP</ows:Value>
       </ows:AllowedValues>
      </ows:Constraint>
     </ows:Get>
    </ows:HTTP>
   </ows:DCP>
  </ows:Operation>
 </ows:OperationsMetadata>
 <Contents>
  <Layer>
   <ows:Title>Blue Marble Next Generation</ows:Title>
   <ows:Abstract>Blue Marble Next Generation NASA Product</ows:Abstract>
   <ows:WGS84BoundingBox>
    <ows:LowerCorner>-180 -90</ows:LowerCorner>
    <ows:UpperCorner>180 90</ows:UpperCorner>
   </ows:WGS84BoundingBox>
   <ows:Identifier>BlueMarbleNextGeneration</ows:Identifier>
   <Style isDefault="true">
    <ows:Identifier>Default</ows:Identifier>
   </Style>
   <Format>image/jpeg</Format>
   <TileMatrixSetLink>
    <TileMatrixSet>BigWorldPixel</TileMatrixSet>
   </TileMatrixSetLink>
   <ResourceURL format="image/png" resourceType="tile"
template="http://www.maps.bob/wmts/BlueMarbleNextGeneration/
default/BigWorldPixel/{TileMatrix}/{TileRow}/{TileCol}.png"/>
  </Layer>
  <TileMatrixSet>
   <ows:Identifier>BigWorldPixel</ows:Identifier>
   <ows:SupportedCRS>urn:ogc:def:crs:OGC:1.3:CRS84</ows:SupportedCRS>

<WellKnownScaleSet>urn:ogc:def:wkss:OGC:1.0:GlobalCRS84Pixel</WellKnownScaleSe
t>
   <TileMatrix>
    <ows:Identifier>10000m</ows:Identifier>
    <ScaleDenominator>33130800.83133142</ScaleDenominator>
```

```
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>7</MatrixWidth>
    <MatrixHeight>5</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>20000m</ows:Identifier>
    <ScaleDenominator>66261601.66266284</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>4</MatrixWidth>
    <MatrixHeight>3</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>40000m</ows:Identifier>
    <ScaleDenominator>132523203.3253257</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>2</MatrixWidth>
    <MatrixHeight>2</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>60000m</ows:Identifier>
    <ScaleDenominator>198784804.9879885</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>1</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>120000m</ows:Identifier>
    <ScaleDenominator>397569609.9759771</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>1</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>240000m</ows:Identifier>
    <ScaleDenominator>795139219.9519541</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>1</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
  </TileMatrixSet>
 </Contents>
 <ServiceMetadataURL
xlink:href="http://www.maps.bob/wmts/1.0.0/WMTSCapabilities.xml"/>
</Capabilities>
```

```
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>7</MatrixWidth>
    <MatrixHeight>5</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>20000m</ows:Identifier>
    <ScaleDenominator>66261601.66266284</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>4</MatrixWidth>
    <MatrixHeight>3</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>40000m</ows:Identifier>
    <ScaleDenominator>132523203.3253257</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>2</MatrixWidth>
    <MatrixHeight>2</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>60000m</ows:Identifier>
    <ScaleDenominator>198784804.9879885</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>1</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>120000m</ows:Identifier>
    <ScaleDenominator>397569609.9759771</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>1</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
   <TileMatrix>
    <ows:Identifier>240000m</ows:Identifier>
    <ScaleDenominator>795139219.9519541</ScaleDenominator>
    <TopLeftCorner>-180 90</TopLeftCorner>
    <TileWidth>640</TileWidth>
    <TileHeight>480</TileHeight>
    <MatrixWidth>1</MatrixWidth>
    <MatrixHeight>1</MatrixHeight>
   </TileMatrix>
  </TileMatrixSet>
 </Contents>
 <ServiceMetadataURL
xlink:href="http://www.maps.bob/wmts/1.0.0/WMTSCapabilities.xml"/>
</Capabilities>
```

Annex E (informative) Well-known scale sets

The following well-known scale sets are defined in this standard. To be conformant to these well-known scale sets, a WMTS server SHALL allow responses from the largest scale denominator on the following tables and all intermediate scale denominators down to the most detailed scale resolution of that data; it is therefore not required to support the smallest scale denominators in order to be conformant to a well-known scale set. Cell sizes (pixel size in terrain units) have been calculated assuming 0.28 mm pixel size and the WGS84 equatorial earth diameter.

URN identifiers for each well-known scale set follow the OGC 07-092r3 best practices document.

E.1 GlobalCRS84Scale (urn:ogc:def:wkss:OGC:1.0:GlobalCRS84Scale)

This well-known scale set has been defined for global cartographic products. Rounded scales have been chosen for intuitive cartographic representation of vector data. Scale denominator is only accurate near the equator.

Table E.1 — Definition of Well-known scale set GlobalCRS84Scale

CRS	Scale Denominator	Pixel Size (degrees)
urn:ogc:def:crs:OGC:1.3:CRS84	500 106	1.25764139776733
	250 106	0.628820698883665
	100 106	0.251528279553466
	50 106	0.125764139776733
	25 106	6.28820698883665 10-2
	10 106	2.51528279553466 10-2
	5 106	1.25764139776733 10-2
	2.5 106	6.28820698883665 10-3
	1 106	2.51528279553466 10-3
	500 103	1.25764139776733 10-3
	250 103	6.28820698883665 10-4
	100 103	2.51528279553466 10-4
	50 103	1.25764139776733 10-4
	25 103	6.28820698883665 10-5
	10 103	2.51528279553466 10-5
	5 103	1.25764139776733 10-5
	2.5 103	6.28820698883665 10-6
	1 103	2.51528279553466 10-6
	500	1.25764139776733 10-6
	250	6.28820698883665 10-7
	100	2.51528279553466 10-7

附录E（资料性）知名比例尺系列

本标准定义了以下知名比例尺系列。为了与这些知名比例尺系列一致，WMTS服务器应可对从以下各表中最大比例尺分母和中间比例尺分母到数据的最小比例尺分母之间的所有比例尺作出响应。为了与某一个知名比例尺系列一致，不要求支持最小比例尺分母。网格大小（单位地面长度对应的像素大小）是在像素大小假定为0.28mm和WGS84下的赤道直径计算得出的。

每个知名比例尺系列的URN标识符是根据OGC 07-092r3最佳实践文档确定的。

E.1 GlobalCRS84Scale (urn:ogc:def:wkss:OGC:1.0:GlobalCRS84Scale)

本知名比例尺系列是针对全球性制图产品定义的。选择了整数比例尺是为了矢量数据直观的地图表达。比例尺分母只是在赤道附近是准确的。

表 E.1 — 知名比例尺系列 GlobalCRS84Scale 的定义

CRS	比例尺分母	像素大小 (度)
urn:ogc:def:crs:OGC:1.3:CRS84	500 106	1.25764139776733
	250 106	0.628820698883665
	100 106	0.251528279553466
	50 106	0.125764139776733
	25 106	6.28820698883665 10-2
	10 106	2.51528279553466 10-2
	5 106	1.25764139776733 10-2
	2.5 106	6.28820698883665 10-3
	1 106	2.51528279553466 10-3
	500 103	1.25764139776733 10-3
	250 103	6.28820698883665 10-4
	100 103	2.51528279553466 10-4
	50 103	1.25764139776733 10-4
	25 103	6.28820698883665 10-5
	10 103	2.51528279553466 10-5
	5 103	1.25764139776733 10-5
	2.5 103	6.28820698883665 10-6
	1 103	2.51528279553466 10-6
	500	1.25764139776733 10-6
	250	6.28820698883665 10-7
	100	2.51528279553466 10-7

E.2 GlobalCRS84Pixel (urn:ogc:def:wkss:OGC:1.0:GlobalCRS84Pixel)

This well-known scale set has been defined for global cartographic products. Rounded pixel sizes have been chosen for intuitive cartographic representation of raster data. Some values have been chosen to coincide with original pixel size of commonly used global products like STRM (1" and 3"), GTOPO (30") or ETOPO (2' and 5'). Scale denominator and approximated pixel size in meters are only accurate near the equator.

Table E.2 — Definition of Well-known scale set GlobalCRS84Pixel

CRS	Scale Denominator	Pixel Size (degrees)	Approx. Pixel Size (m)
urn:ogc:def:crs:OGC:1.3:CRS84	795139219.9519541	2	240000
	397569609.9759771	1	120000
	198784804.9879885	0.5 (30')	60000
	132523203.3253257	0.333333333333333 (20')	40000
	66261601.66266284	0.166666666666667 (10')	20000
	33130800.83133142	$8.333333333333333 \cdot 10^{-2}$ (5')	10000
	13252320.33253257	$3.333333333333333 \cdot 10^{-2}$ (2')	4000
	6626160.166266284	$1.666666666666667 \cdot 10^{-2}$ (1')	2000
	3313080.083133142	$8.333333333333333 \cdot 10^{-3}$ (30")	1000
	1656540.041566571	$4.166666666666667 \cdot 10^{-3}$ (15")	500
	552180.0138555236	$1.388888888888889 \cdot 10^{-3}$ (5")	166
	331308.0083133142	$8.333333333333333 \cdot 10^{-4}$ (3")	100
	110436.0027711047	$2.777777777777778 \cdot 10^{-4}$ (1")	33
	55218.00138555237	$1.388888888888889 \cdot 10^{-4}$ (0.5")	16
	33130.80083133142	$8.333333333333333 \cdot 10^{-5}$ (0.3")	10
	11043.60027711047	$2.777777777777778 \cdot 10^{-5}$ (0.1")	3
	3313.080083133142	$8.333333333333333 \cdot 10^{-6}$ (0.03")	1
	1104.360027711047	$2.777777777777778 \cdot 10^{-6}$ (0.01")	0.33

E.3 GoogleCRS84Quad (urn:ogc:def:wkss:OGC:1.0:GoogleCRS84Quad)

This well-known scale set has been defined to allow quadtree pyramids in CRS84. Level 0 allows representing the whole world in a single 256x256 pixels (where the first 64 and last 64 lines of the tile are left blank). The next level represents the whole world in 2x2 tiles of 256x256 pixels and so on in powers of 2. Scale denominator is only accurate near the equator.

E.2 GlobalCRS84Pixel (urn:ogc:def:wkss:OGC:1.0:GlobalCRS84Pixel)

本知名比例尺系列是针对全球性制图产品定义的。选择取整的像素大小是为了栅格数据直观的地图表达。选择有些值是为了与诸如STRM (1" 和 3"), GTOPO (30") 或ETOPO (2'和5')等一些广泛使用的全球产品的原始像素大小一致。 比例尺分母和以米为单位的像素大概大小只是在赤道附近是准确的。

表 E.2 —知名比例尺系列 GlobalCRS84Pixel 的定义

CRS	比例尺分母	像素大小 (度)	大致的像素大小 (m)
urn:ogc:def:crs:OGC:1.3:CRS84	795139219.9519541	2	240000
	397569609.9759771	1	120000
	198784804.9879885	0.5 (30')	60000
	132523203.3253257	0.333333333333333 (20')	40000
	66261601.66266284	0.166666666666667 (10')	20000
	33130800.83133142	$8.333333333333333\ 10^{-2}$ (5')	10000
	13252320.33253257	$3.333333333333333\ 10^{-2}$ (2')	4000
	6626160.166266284	$1.666666666666667\ 10^{-2}$ (1')	2000
	3313080.083133142	$8.333333333333333\ 10^{-3}$ (30")	1000
	1656540.041566571	$4.166666666666667\ 10^{-3}$ (15")	500
	552180.0138555236	$1.388888888888889\ 10^{-3}$ (5")	166
	331308.0083133142	$8.333333333333333\ 10^{-4}$ (3")	100
	110436.0027711047	$2.777777777777778\ 10^{-4}$ (1")	33
	55218.00138555237	$1.388888888888889\ 10^{-4}$ (0.5")	16
	33130.80083133142	$8.333333333333333\ 10^{-5}$ (0.3")	10
	11043.60027711047	$2.777777777777778\ 10^{-5}$ (0.1")	3
	3313.080083133142	$8.333333333333333\ 10^{-6}$ (0.03")	1
	1104.360027711047	$2.777777777777778\ 10^{-6}$ (0.01")	0.33

E.3 GoogleCRS84Quad (urn:ogc:def:wkss:OGC:1.0:GoogleCRS84Quad)

本知名比例尺系列是针对CRS84坐标系下使用四叉树金字塔结构设计的。第0级把全球用一个256x256像素的图块表示（其中前后各64行为空白）。下一个级别把全球表示成2x2 个256x256像素的图块，以此类推，逐级乘以2次方。比例尺分母只是在赤道附近是准确的。

Table E.3 — Definition of Well-known scale set GoogleCRS84Quad

CRS	Scale Denominator	Pixel Size (degrees)
urn:ogc:def:crs:OGC:1.3:CRS84	559082264.0287178	1.40625000000000
	279541132.0143589	0.703125000000000
	139770566.0071794	0.351562500000000
	69885283.00358972	0.175781250000000
	34942641.50179486	8.78906250000000 10-2
	17471320.75089743	4.39453125000000 10-2
	8735660.375448715	2.19726562500000 10-2
	4367830.187724357	1.09863281250000 10-2
	2183915.093862179	5.49316406250000 10-3
	1091957.546931089	2.74658203125000 10-3
	545978.7734655447	1.37329101562500 10-3
	272989.3867327723	6.86645507812500 10-4
	136494.6933663862	3.43322753906250 10-4
	68247.34668319309	1.71661376953125 10-4
	34123.67334159654	8.58306884765625 10-5
	17061.83667079827	4.29153442382812 10-5
	8530.918335399136	2.14576721191406 10-5
	4265.459167699568	1.07288360595703 10-5
	2132.729583849784	5.36441802978516 10-6

E.4 GoogleMapsCompatible (urn:ogc:def:wkss:OGC:1.0:GoogleMapsCompatible)

This well-known scale set has been defined to be compatible with Google Maps and Microsoft Live Map projections and zoom levels. Level 0 allows representing the whole world in a single 256x256 pixels. The next level represents the whole world in 2x2 tiles of 256x256 pixels and so on in powers of 2. Scale denominator is only accurate near the equator.

表 E.3 — 知名比例尺系列 GoogleCRS84Quad 的定义

CRS	比例尺分母	像素大小(度)
urn:ogc:def:crs:OGC:1.3:CRS84	559082264.0287178	1.40625000000000
	279541132.0143589	0.703125000000000
	139770566.0071794	0.351562500000000
	69885283.00358972	0.175781250000000
	34942641.50179486	8.78906250000000 10-2
	17471320.75089743	4.39453125000000 10-2
	8735660.375448715	2.19726562500000 10-2
	4367830.187724357	1.09863281250000 10-2
	2183915.093862179	5.49316406250000 10-3
	1091957.546931089	2.74658203125000 10-3
	545978.7734655447	1.37329101562500 10-3
	272989.3867327723	6.86645507812500 10-4
	136494.6933663862	3.43322753906250 10-4
	68247.34668319309	1.71661376953125 10-4
	34123.67334159654	8.58306884765625 10-5
	17061.83667079827	4.29153442382812 10-5
	8530.918335399136	2.14576721191406 10-5
	4265.459167699568	1.07288360595703 10-5
	2132.729583849784	5.36441802978516 10-6

E.4 GoogleMapsCompatible (urn:ogc:def:wkss:OGC:1.0:GoogleMapsCompatible)

本知名比例尺系列是为了与Google Maps和Microsoft Live Map的投影和缩放级别兼容而定义的。第0级把全球用一个256x256像素的图块表示（其中前后各64行为空白）。下一个级别把全球表示成2x2个256x256像素的图块，以此类推，逐级乘以2次方。比例尺分母只是在赤道附近是准确的。

Table E.4 — Definition of Well-known scale set GoogleMapsCompatible

CRS	Zoom level name	Scale Denominator	Pixel Size (m)
urn:ogc:def:crs:EPSG:6.18:3:3857 WGS 84 / Pseudo-Mercator http://www.epsg-registry.org/export.htm?gml=urn:ogc:def:crs:EPSG::3857	0	559082264.0287178	156543.0339280410
	1	279541132.0143589	78271.51696402048
	2	139770566.0071794	39135.75848201023
	3	69885283.00358972	19567.87924100512
	4	34942641.50179486	9783.939620502561
	5	17471320.75089743	4891.969810251280
	6	8735660.375448715	2445.984905125640
	7	4367830.187724357	1222.992452562820
	8	2183915.093862179	611.4962262814100
	9	1091957.546931089	305.7481131407048
	10	545978.7734655447	152.8740565703525
	11	272989.3867327723	76.43702828517624
	12	136494.6933663862	38.21851414258813
	13	68247.34668319309	19.10925707129406
	14	34123.67334159654	9.554628535647032
	15	17061.83667079827	4.777314267823516
	16	8530.918335399136	2.388657133911758
	17	4265.459167699568	1.194328566955879
	18	2132.729583849784	0.5971642834779395

表 E.4 — 知名比例尺系列 GoogleMapsCompatible 的定义

CRS	缩放级别	比例尺分母	像素大小 (m)
urn:ogc:def:crs:EPSG:6.18:3:3857 WGS 84 / Pseudo-Mercator http://www.epsg-registry.org/export.htm?gml=urn:ogc:def:crs:EPSG::3857	0	559082264.0287178	156543.0339280410
	1	279541132.0143589	78271.51696402048
	2	139770566.0071794	39135.75848201023
	3	69885283.00358972	19567.87924100512
	4	34942641.50179486	9783.939620502561
	5	17471320.75089743	4891.969810251280
	6	8735660.375448715	2445.984905125640
	7	4367830.187724357	1222.992452562820
	8	2183915.093862179	611.4962262814100
	9	1091957.546931089	305.7481131407048
	10	545978.7734655447	152.8740565703525
	11	272989.3867327723	76.43702828517624
	12	136494.6933663862	38.21851414258813
	13	68247.34668319309	19.10925707129406
	14	34123.67334159654	9.554628535647032
	15	17061.83667079827	4.777314267823516
	16	8530.918335399136	2.388657133911758
	17	4265.459167699568	1.194328566955879
	18	2132.729583849784	0.5971642834779395

Annex F (normative) WSDL description of the service

This Annex is normative for services that support SOAP messages. It provides an abstract WSDL description for a generic WMTS service and guidance on how to create a concrete WSDL description for a particular WMTS server instance.

A WSDL document is typically used in combination with SOAP encoding but this annex describes WSDL documents that deal with KVP and SOAP encodings of the procedure oriented architectural style. WSDL documents (especially in version 1.1) seem not to be appropriate to describe resource oriented architectural style.

F.1 General

The Web Services Description Language (WSDL) is an XML language for describing the computational characteristics of a web service: interface signatures, protocol bindings and network endpoints. Version 1.1 was authored jointly by Microsoft and IBM and published as a W3C Note in March 2001. Furthermore, WSDL 1.1 is recommended by WS-I Basic Profile 1.2 and the current drafts of WS-I Basic Profile 2.0, which will be used in conjunction with WSDL 1.1. WSDL 2.0 (formerly 1.2 but renamed because of its substantial differences from 1.1) has already been promoted to a W3C Recommendation but is not widely supported yet by standards and tools.

F.2 WSDL Publication

There are many ways to publish a WSDL file for a Web Service instance. The mainstream IT world has established three major ways:

1. The WSDL file can be offered on the web site of the organization that publishes the web service. This approach allows humans to find the WSDL file but is not sufficient for automatic use.

2. The WSDL file can be published through public and private registries. UDDI would be the choice for general Web Services and CSW for OGC Web Services. This approach follows the publish-find-bind pattern and thus allows humans and services to discover the WSDL file in a standardized manner.

3. The web service itself can also publish the WSDL file. AXIS and the .NET frameworks follow the convention of http://url:port/service/xx?WSDL. This is sufficient for a pragmatic approach, but fails for multiple WSDL files describing specific aspects of a Web Service as long as the base URL does not change.

Option number two can be combined with option number three by just pointing to the WSDL initially published via the WSDL pattern. Therefore, it is recommended to publish a single WSDL file as described in option three but ideally to publish it through any kind of registry.

Also, within the service metadata document, a <WSDL> element may be used to specify a reference to a WSDL resource. If the service has a SOAP binding, there SHALL be a <WSDL> element. The value of the xlink:href attribute SHALL refer to a web accessible WSDL document. The xlink:role attribute indicates the namespace of the document element and WSDL version (in our examples http://schemas.xmlsoap.org/wsdl/1.0). The xlink:show attribute has the value "none" to indicate that no specific behavior is intended. This approach enables OGC Web Services with SOAP bindings to be discovered via the GetCapabilities operation and to get additional information through the referenced WSDL file.

附录F（规范性）服务的WSDL描述

本附录对于支持SOAP消息的服务是规范性的，为通用WMTS服务提供了一个抽象WSDL描述，并就如何为特定WMTS服务器实例生成具体WSDL描述提供了指南。

WSDL文档通常与SOAP编码结合使用，但本附录仅描述处理KVP和面向过程架构风格的SOAP编码的WSDL文档。WSDL文档（特别是版本V1.1）似乎不适于描述面向资源架构风格。

F.1 一般性说明

Web服务描述语言（WSDL）是一种描述Web服务各个概念特征，如接口签名、协议绑定以及网络端点的XML语言。版本1.1是由微软和IBM联合提出并作为W3C记录于2001年3月发布。之后，WS-I Basic Profile 1.2和目前的WS-I Basic Profile 2.0草案建议使用WSDL 1.1。WSDL 2.0（之前为1.2，但由于与1.1的实质性差异改名）已被提升为W3C建议，但还未被各个标准和软件工具广泛支持。

F.2 WSDL发布

有多种为Web服务实例发布一个WSDL文件的方式。IT界已建立了3种主要方式：

1. WSDL文件通过发布服务的机构网站对外提供。该方法可以让人找到该WSDL文件，但不便于自动的方式使用该WSDL文件。

2. WSDL文件可以通过公共的或私有的注册系统发布。UDDI可以用来发布通用的Web服务，CSW可以用来发布OGC的Web服务。该方法遵循了发布-发现-绑定的模式，可以让人和服务都可以以一种标准的方式发现该WSDL文件。

3. Web服务本身也可以发布WSDL文件。AXIS和.Net框架则遵循http://url:port/service/xx?WSDL的约定来发布WSDL文件。这对于程序访问足够，但对于描述Web服务不同侧面的多个WSDL文件的发布则不行，因为基本的URL不能改变。

第2种方式可以与第3种方式结合，只需要通过把WSDL模式指向WSDL最初发布的地方。因此，建议选择第3种方式发布单个的WSDL文件，但理想方法是通过一个注册系统进行发布。

同时，在服务元数据文档中，可以使用一个<WSDL>元素指定一个WSDL资源。如果服务具有一个SOAP绑定，则应该包含一个<WSDL>元素。属性xlink:href的值应指向一个可以通过Web访问的文档。xlink:role属性应标明文档元素的名称空间和WSDL版本（例如http://schemas.xmlsoap.org/wsdl/1.0）。xlink:show元素的值为"none"，以表明不希望有特定的行为。这一方法使得OGC的SOAP绑定实现的Web服务可以通过GetCapabilities操作发现，并通过引用的WSDL文件获得其他的信息。

F.3 Abstract and concrete WSDL documents

WSDL documents are intended to be modularized through the use of import statements since they are structured in an abstract and concrete service instance part. These mechanisms permit the separation of service-specific elements from shared interface definitions; such an authoring style is recommended in the WSDL specifications, and it is advocated here. In practice, this separation means that the complete OGC service will always be described by exactly one top-level WSDL. This top-level WSDL file may import a set of WSDL files for specific parts, for instance a WSDL for the abstract part and a WSDL describing only the concrete service instance part of the service.

A concrete WSDL 1.1 document need to describe five main parts: types, messages, portTypes, bindings and services. The abstract WSDL document describes types, messages and portTypes in a generic way and can be imported in the concrete WSDL description of any particular service instance. A concrete WSDL document also has to describe the binding and service parts of the document.

F.4 Abstract WMTS WSDL document

Abstract WSDL document have a modular design that reuses application schemas that are also used by SOAP messages and are mentioned in the Annex B and in some subclauses of this document.

In addition to this document, this standard includes and abstract WSDL document and some XML Schema Documents imported by the abstract WSDL and also mentioned in the Annex B. These XML Schema Documents and the abstract WSDL are bundled in a zip file with the present document. After OGC acceptance of a Version 1.0.0 of this standard, these files will also be posted online at the URL http://schemas.opengis.net/wmts/1.0.0. In the event of a discrepancy between the bundled and online versions of the XML Schema Documents, the online files SHALL be considered authoritative.

The documents relevant only in WSDL document creation are named:

wmtsAbstract.wsdl

wmts.xsd

wmtsKVP.xsd

Also complete examples can be found on the zip file and on the portal. Some fragments of these examples are shown throughout this document.

This is WSDL schema fragment for the abstract WMTS service illustrates the ability of importing some XDS files:

```
<definitions name="WMTS">

 <types>
<xsd:schematargetNamespace="http://www.opengis.net/wmts_wsdl/1.0">
<xsd:import namespace="http://www.opengis.net/ows/1.1"
schemaLocation="../../ows/1.1.0/owsCommon.xsd"/>
<xsd:import namespace="http://www.opengis.net/wmts/1.0"
schemaLocation="wmts.xsd"/>
</xsd:schema>
 </types>
```

F.3 抽象与具体WSDL文档

WSDL文档往往通过使用import语句实现模块化，因为这些文档被组织为抽象和具体服务实例部分。这样的方法可以把与服务相关的元素从共用的接口定义中分离出来；这种方式为各个WSDL规范所推荐，这里也支持这样。具体操作中，这一分离方法意味着完整的OGC服务将始终通过仅仅一个顶层的WSDL文件描述。该顶层WSDL文件可以导入一组不同部分的WSDL文件，例如抽象部分一个WSDL文件，服务的具体服务实例部分一个WSDL文件。

一个具体的WSDL 1.1的文档需要描述5个主要部分：类型、消息、端口类型（portType）绑定和服务。抽象的WSDL文档以一种通用的方式描述类型、消息和portType，并可以被导入到描述具体服务实例的具体WSDL中。一个具体WSDL文档也必须描述文档的绑定和服务部分。

F.4 抽象WMTS WSDL文档

抽象WSDL文档具有的模块化设计，可以重用被SOAP消息使用的应用模式，这些应用模式在附录B和本标准的一些章节中被提及。

除此文档之外，本标准包括一个抽象WSDL文档和一些由抽象WSDL导入的XML模式文档，这些模式文档在附录B中也有提及。这些XML模式文档和抽象WSDL与本标准文档一起打包为一个ZIP文件。在OGC接受本标准为1.0.0之后，这些文件也将放到网上通过http://schemas.opengis.net/wmts/1.0.0访问。如果打包文件和XML模式文件的在线版本之间有差别，则在线文件应视为权威版本。

仅与WSDL文档创建相关的文档名称如下：

wmtsAbstract.wsdl

wmts.xsd

wmtsKVP.xsd

完整的实例也可以在zip文件和OGC门户上找到。本文档给出了一些这些实例的片段。

以下是抽象WMTS服务的WSDL模式片段，显示了导入其他XDS文件的功能：

```
<definitions name="WMTS">

 <types>
<xsd:schematargetNamespace="http://www.opengis.net/wmts_wsdl/1.0">
<xsd:import namespace="http://www.opengis.net/ows/1.1"
schemaLocation="../../ows/1.1.0/owsCommon.xsd"/>
<xsd:import namespace="http://www.opengis.net/wmts/1.0"
schemaLocation="wmts.xsd"/>
</xsd:schema>
 </types>
```

```
 <message name="GetTileMessage_GET">
<part name="service" type="wmts:RequestServiceType"/>
<part name="request" type="wmts:GetTileValueType"/>
<part name="version" type="wmts:VersionType"/>
<part name="layer" type="xsd:string"/>
<part name="style" type="xsd:string"/>
<part name="format" type="ows:MimeType"/>
<part name="TileMatrixSet" type="xsd:string"/>
<part name="TileMatrix" type="xsd:string"/>
<part name="TileRow" type="xsd:unsignedInt"/>
<part name="TileCol" type="xsd:unsignedInt"/>
 </message>
 <message name="GetTileMessage_POST">
<part name="request" element="wmts:GetTile"/>
 </message>
 <message name="GetTileResult_SOAP">
<part name="body" element="wmts:BinaryPayload" />
 </message>

 <portType name="WMTS_HTTP_Port_GET">
<operation name="GetTile">
 <input message="tns:GetTileMessage_GET"/>
 <output message="tns:GetTileResult"/>
 <fault name="exception" message="tns:ServiceExceptionMessage"/>
</operation>
 </portType>
 <portType name="WMTS_HTTP_Port_SOAP">
<operation name="GetTile">
 <input message="tns:GetTileMessage_POST"/>
 <output message="tns:GetTileResult_SOAP"/>
 <fault name="exception" message="tns:ServiceExceptionMessage"/>
</operation>
 </portType>
</definitions>
```

F.5 Concrete WMTS WSDL document

A concrete WSDL describes a particular server instance. First, it has to import the abstract WSDL file, then it has to use a <binding> element for each encoding the server supports and, finally, a service element has to be used and it has to contain as port elements as encodings the server supports.

A HTTP GET <binding> element SHALL reference a GET portType from the abstract part and make use of the `<http:binding verb="GET">` binding as described in the WSDL 1.1 specification. The operation element SHALL reference the corresponding operation from the abstract part and use `<http:urlEncoded/>` as input and `<mime />` element as output. The `<http:operation>` element SHALL be used according to the WSDL 1.1 and WS-I Basic.

A SOAP <binding> element SHALL reference a SOAP portType from the abstract part and make use of the `<soap:binding style="document">` binding as described in the WSDL 1.1 specification. The operation element SHALL reference the corresponding operation from the abstract part and use `<soap:body use="literal"/>` as input and output. The `<soap:operation>` element SHALL be used according to the WSDL 1.1 and WS-I Basic Profile 1.2 specifications. The soapAction attribute value SHALL follow the format template:

http://www.opengis.net/{serviceType}/requests#{operationName}

```
<message name="GetTileMessage_GET">
<part name="service" type="wmts:RequestServiceType"/>
<part name="request" type="wmts:GetTileValueType"/>
<part name="version" type="wmts:VersionType"/>
<part name="layer" type="xsd:string"/>
<part name="style" type="xsd:string"/>
<part name="format" type="ows:MimeType"/>
<part name="TileMatrixSet" type="xsd:string"/>
<part name="TileMatrix" type="xsd:string"/>
<part name="TileRow" type="xsd:unsignedInt"/>
<part name="TileCol" type="xsd:unsignedInt"/>
 </message>
 <message name="GetTileMessage_POST">
<part name="request" element="wmts:GetTile"/>
 </message>
 <message name="GetTileResult_SOAP">
<part name="body" element="wmts:BinaryPayload" />
 </message>

 <portType name="WMTS_HTTP_Port_GET">
<operation name="GetTile">
 <input message="tns:GetTileMessage_GET"/>
 <output message="tns:GetTileResult"/>
 <fault name="exception" message="tns:ServiceExceptionMessage"/>
</operation>
 </portType>
 <portType name="WMTS_HTTP_Port_SOAP">
<operation name="GetTile">
 <input message="tns:GetTileMessage_POST"/>
 <output message="tns:GetTileResult_SOAP"/>
 <fault name="exception" message="tns:ServiceExceptionMessage"/>
</operation>
 </portType>
</definitions>
```

F.5 具体WMTS WSDL文档

一个具体WSDL描述了一个特定的服务实例。首先，它必须导入抽象WSDL文件，然后必须针对每一个服务器支持的编码方法使用一个<binding>元素，最后，必须使用一个service元素，该元素必须包含一个服务器支持编码的port元素。

HTTP GET <binding>元素应指向抽象部分的一个GET portType，并遵照WSDL 1.1规范利用<http:binding verb="GET">绑定。operation元素应指向抽象部分的相应操作，并使用<http:urlEncoded/>作为输入，使用<mime/>作为输出，应根据WSDL 1.1和WS-I Basic的要求使用<http:operation>元素。

SOAP <binding>元素应引用一个抽象部分的SOAP portType，并按照WSDL 1.1规范使用<soap:binding style="document">绑定。operation元素应指向抽象部分的相应操作，并使用<soap:body use="literal"/>作为输入和输出，应根据WSDL 1.1和WS-I Basic Profile 1.2的要求使用<http:operation>元素。soapAction属性的值应符合下面的格式模板：

http://www.opengis.net/{serviceType}/requests#{operationName}

<port> elements in the <service> element SHALL reference bindings from the binding part. In a GET encoding an <http:address location=""> element SHALL reference the URL of the service (that has to be the same of the <ows:Get xlink:href=""> element attribute in the service metadata document). In a SOAP encoding an <soap:address location=""> element SHALL reference the URL of the service (that has to be the same value of the <ows:Get xlink:href=""> element attribute in the service metadata document).

F.6 Concrete WMTS WSDL document example

The following example illustrates how a concrete WSDL document imports the abstract WSDL document and adds a particular description of bindings and service for this particular server instance. The example only describes GetTile operation and describes the same server example shown in subclause 7.1.3 of this document:

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions xmlns:wmts_wsdl="http://www.opengis.net/wmts_wsdl/1.0"
 targetNamespace="http://www.opengis.net/wmts_wsdl/1.0">
<!-- import WMTS types, message and portType -->
<import namespace="http://www.opengis.net/wmts_wsdl/1.0"
 location="../wmtsAbstract.wsdl"/>
<!--Bindings --><!-- HTTP Get KVPbindings -->
<binding name="WMTS_HTTP_GET_Binding" type="wmts_wsdl:WMTS_HTTP_Port_GET">
<http:binding verb="GET"/>
<operation name="GetTile">
 <http:operation location=""/>
 <input><http:urlEncoded/> </input>
 <output><mime:content type="image/*"/></output>
 <fault name="exception">
<mime:mimeXML/>
 </fault>
</operation>
</binding>
<!-- HTTP Post SOAPbindings -->
<binding name="WMTS_SOAP_Binding" type="wmts_wsdl:WMTS_HTTP_Port_SOAP">
<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
<operation name="GetTile">
 <soap:operation
 soapAction="http://www.opengis.net/wms/requests#GetTile"/>
 <input><soap:body use="literal"/></input>
 <output><soap:body use="literal"/></output>
 <fault name="exception">
<soap:fault name="exception" use="literal"/>
 </fault>
</operation>
</binding>
<!-- Services-->
 <service name="WMTS-TiledWorld-UAB-CREAF-MiraMon">
<port name="WMTS-GET-Port" binding="wmts_wsdl:WMTS_HTTP_GET_Binding">
 <http:address location="http://www.maps.bob/maps.cgi"/>
</port>
<port name="WMTS-SOAP-Port"
binding="wmts_wsdl:WMTS_SOAP_Binding">
 <soap:address location="http://www.maps.bob/maps.cgi"/>
</port>
</service>
</definitions>
```

<service>元素中的<port>元素应引用绑定部分的绑定。在GET编码中，<http:address location="">元素应指向服务的URL（应与服务元数据文档中<ows:Get xlink:href="">元素的属性一样）。在SOAP编码中，<soap:address location="">元素应指向服务的URL（应与服务元数据文档中<ows:Get xlink:href="">元素的属性一样）

F.6 具体WMTS WSDL文档实例

以下实例显示了一个具体WSDL文档如何导入抽象WSDL文档和如何为特定服务实例增加一个绑定和服务描述。该实例仅仅描述了GetTile操作，并描述了本文档7.1.3中所示的同一个服务器实例：

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions xmlns:wmts_wsdl="http://www.opengis.net/wmts_wsdl/1.0"
 targetNamespace="http://www.opengis.net/wmts_wsdl/1.0">
<!-- import WMTS types, message and portType -->
<import namespace="http://www.opengis.net/wmts_wsdl/1.0"
 location="../wmtsAbstract.wsdl"/>
<!--Bindings --><!-- HTTP Get KVPbindings -->
<binding name="WMTS_HTTP_GET_Binding" type="wmts_wsdl:WMTS_HTTP_Port_GET">
<http:binding verb="GET"/>
<operation name="GetTile">
 <http:operation location=""/>
 <input><http:urlEncoded/> </input>
 <output><mime:content type="image/*"/></output>
 <fault name="exception">
<mime:mimeXML/>
 </fault>
</operation>
</binding>
<!-- HTTP Post SOAPbindings -->
<binding name="WMTS_SOAP_Binding" type="wmts_wsdl:WMTS_HTTP_Port_SOAP">
<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
<operation name="GetTile">
 <soap:operation
 soapAction="http://www.opengis.net/wms/requests#GetTile"/>
 <input><soap:body use="literal"/></input>
 <output><soap:body use="literal"/></output>
 <fault name="exception">
<soap:fault name="exception" use="literal"/>
 </fault>
</operation>
</binding>
<!-- Services-->
 <service name="WMTS-TiledWorld-UAB-CREAF-MiraMon">
<port name="WMTS-GET-Port" binding="wmts_wsdl:WMTS_HTTP_GET_Binding">
 <http:address location="http://www.maps.bob/maps.cgi"/>
</port>
<port name="WMTS-SOAP-Port"
binding="wmts_wsdl:WMTS_SOAP_Binding">
 <soap:address location="http://www.maps.bob/maps.cgi"/>
</port>
</service>
</definitions>
```

Annex H (informative) Pseudocode

This informative Annex provides pseudocode that illustrates how to get the tiles that cover a bounding box rectangle and how to get the CRS coordinates that bounds a tile.

H.1 From BBOX to tile indices

The following fragment of pseudocode could be used to convert from a desired bounding box (bBoxMinX, bBoxMinY, bBoxMaxX, bBoxMaxY) in CRS coordinates to a range of tile set indices. This pseudocode uses the same notation that subclause 6.1 uses. In this pseudocode we assume that bBoxMinX, bBoxMinY, bBoxMaxX, bBoxMaxY, tileMatrixMinX, tileMatrixMinY, tileMatrixMinY, tileMatrixMaxY, tileSpanX and tileSpanY are floating point variables (IEEE-754) that has accuracy issues derived from the finite precision of the representation. These accuracy issues could be amplified in a typical floor() rounding down function that could return a value ±1 than that expected. To overcome this issue this code uses a small value (epsilon) added or subtracted in a place that is not affected by CRS coordinate precision.

```
// to compensate for floating point computation inaccuracies
epsilon = 1e-6

tileMinCol = floor((bBoxMinX - tileMatrixMinX) / tileSpanX + epsilon)
tileMaxCol = floor((bBoxMaxX - tileMatrixMinX) / tileSpanX - epsilon)
tileMinRow = floor((tileMatrixMaxY - bBoxMaxY) / tileSpanY + epsilon)
tileMaxRow = floor((tileMatrixMaxY - bBoxMinY) / tileSpanY - epsilon)

// to avoid requesting out-of-range tiles
if (tileMinCol < 0) tileMinCol = 0
if (tileMaxCol >= matrixWidth) tileMaxCol = matrixWidth-1
if (tileMinRow < 0) tileMinRow = 0
if (tileMaxRow >= matrixHeight) tileMaxRow = matrixHeight-1
```

To fetch all the tiles that cover this bounding box, a client would scan through tileMinCol to tileMaxCol and tileMinRow to tileMaxRow, all inclusive. A total of (tileMaxCol- tileMinCol+1) × (tileMaxRow- tileMinRow+1) will be fetched.

H.2 From tile indices to BBOX

The following pseudocode could be used to convert from a pair of tile indices (tileCol, tileRow) to the bounding box (in CRS coordinates) of this tile defined by the upper-left corner (leftX, upperY) of the tile:

```
leftX = tileCol * tileSpanX + tileMatrixMinX
upperY = tileMatrixMaxY - tileRow * tileSpanY
```

and the lower-right corner (rightX, lowerY) of the tile:

```
rightX = (tileCol+1) * tileSpanX + tileMatrixMinX
lowerY = tileMatrixMaxY – (tileRow+1) * tileSpanY
```

附录H（资料性）伪代码

本资料性附录提供了根据范围框获取图块和获取图块范围框坐标的伪代码。

H.1 根据BBOX计算图块行列号

以下伪代码片段可用于把具有特定CRS坐标的范围框(bBoxMinX, bBoxMinY, bBoxMaxX, bBoxMaxY)转换为图块集的行列范围。该伪代码使用了6.1中的标记方法。该伪代码中，假定bBoxMinX, bBoxMinY, bBoxMaxX, bBoxMaxY, tileMatrixMinX, tileMatrixMinY, tileMatrixMinY, tileMatrixMaxY, tileSpanX 和tileSpanY均为浮点变量(IEEE-754)，它具有源于有限精度表示法的准确性问题。典型情况下，如果使用了floor()函数向下取整，这些准确性问题可能会被放大，可能返回的值为期望值±1。为了避免这一问题，本代码使用一个极小值（epsilon），在不被CRS坐标精度影响的地方加上或减去该值。

```
// 为补偿浮点计算的不准确性
epsilon = 1e-6

tileMinCol = floor((bBoxMinX - tileMatrixMinX) / tileSpanX + epsilon)
tileMaxCol = floor((bBoxMaxX - tileMatrixMinX) / tileSpanX - epsilon)
tileMinRow = floor((tileMatrixMaxY - bBoxMaxY) / tileSpanY + epsilon)
tileMaxRow = floor((tileMatrixMaxY - bBoxMinY) / tileSpanY - epsilon)

// 为避免请求越界图块
if (tileMinCol < 0) tileMinCol = 0
if (tileMaxCol >= matrixWidth) tileMaxCol = matrixWidth-1
if (tileMinRow < 0) tileMinRow = 0
if (tileMaxRow >= matrixHeight) tileMaxRow = matrixHeight-1
```

要获取所有覆盖该范围框的图块，客户端可以扫描tileMinCol到tileMaxCol，和tileMinRow到tileMaxRow的整个范围，包括界点。累计(tileMaxCol- tileMinCol+1) × (tileMaxRow- tileMinRow+1)个图块将被获取。

H.2 根据图块行列号计算BBOX

以下伪代码可用来把一对图块下标(tileCol, tileRow)转换为该图块用CRS坐标表示的范围框。图块左上角点坐标：

```
leftX = tileCol * tileSpanX + tileMatrixMinX
upperY = tileMatrixMaxY - tileRow * tileSpanY
```

图块右下角点坐标：

```
rightX = (tileCol+1) * tileSpanX + tileMatrixMinX
lowerY = tileMatrixMaxY – (tileRow+1) * tileSpanY
```

Bibliography

[1] OGC 05-007r7, OpenGIS® Web Processing Service Implementation Specification version 1.0.0, at http://portal.opengeospatial.org/files/?artifact_id=24151

[2] OGC 06-042, OpenGIS® Web Map Server Implementation Specification version 1.3.0, at http://portal.opengeospatial.org/files/?artifact_id=14416

[3] OGC 06-121r3, OpenGIS® Web Service Common Implementation Specification version 1.1.0 with Corrigendum 1, at http://portal.opengeospatial.org/files/?artifact_id=20040

[4] OGC 07-057r2, OpenGIS® Tiled WMS Discussion Paper version 0.3.0, at http://portal.opengeospatial.org/files/?artifact_id=23206

[5] OGC 07-092r3, Definition identifier URNs in OGC namespace Best Practices version 1.3, at http://portal.opengeospatial.org/files/index.php?artifact_id=30575

[6] OGC 07-146r2 OGC® KML at https://portal.opengeospatial.org/files/?artifact_id=27810

[7] OGC 07-156r1 Integration of Resource-Oriented Architecture Concepts into the OGC Reference Model at http://portal.opengeospatial.org/files/?artifact_id=29634&version=1

[8] OGC 08-142 OWS-Common WSDL Annex

[9] OGC 09-006 OWS-6-DSS Engineering Report – SOAPXML and REST in WMTS at http://portal.opengeospatial.org/files/?artifact_id=31987

[10] OGC 09-144r1, Technical Committee Policies and Procedures: MIME Media Types for GML, Version 1.0, at http://portal.opengeospatial.org/files/?artifact_id=37743

[11] OnEarth WMS tiled extension at http://onearth.jpl.nasa.gov/tiled.html

[12] OpenSearch Specification 1.1 draft 4 http://www.opensearch.org/Specifications/OpenSearch/1.1

[13] OSGeo Tile Map Service Specification version 1.0 at http://wiki.osgeo.org/wiki/Tile_Map_Service_Specification

[14] OSGeo WMS Tile Caching at http://wiki.osgeo.org/wiki/WMS_Tile_Caching

[15] RESTful Web Services. Web Services for the real World. Leonard Richardson and Sam Ruby. O'Reilly Media Inc. 2007.

[16] URI Template, draft-gregorio-uritemplate-03. Network Working Group Internet-Draft. J. Gregorio, Ed. http://tools.ietf.org/html/draft-gregorio-uritemplate-03

参考文献

[1] OGC 05-007r7, OpenGIS® Web Processing Service Implementation Specification version 1.0.0, at http://portal.opengeospatial.org/files/?artifact_id=24151

[2] OGC 06-042, OpenGIS® Web Map Server Implementation Specification version 1.3.0, at http://portal.opengeospatial.org/files/?artifact_id=14416

[3] OGC 06-121r3, OpenGIS® Web Service Common Implementation Specification version 1.1.0 with Corrigendum 1, at http://portal.opengeospatial.org/files/?artifact_id=20040

[4] OGC 07-057r2, OpenGIS® Tiled WMS Discussion Paper version 0.3.0, at http://portal.opengeospatial.org/files/?artifact_id=23206

[5] OGC 07-092r3, Definition identifier URNs in OGC namespace Best Practices version 1.3, at http://portal.opengeospatial.org/files/index.php?artifact_id=30575

[6] OGC 07-146r2 OGC® KML at https://portal.opengeospatial.org/files/?artifact_id=27810

[7] OGC 07-156r1 Integration of Resource-Oriented Architecture Concepts into the OGC Reference Model at http://portal.opengeospatial.org/files/?artifact_id=29634&version=1

[8] OGC 08-142 OWS-Common WSDL Annex

[9] OGC 09-006 OWS-6-DSS Engineering Report – SOAPXML and REST in WMTS at http://portal.opengeospatial.org/files/?artifact_id=31987

[10] OGC 09-144r1, Technical Committee Policies and Procedures: MIME Media Types for GML, Version 1.0, at http://portal.opengeospatial.org/files/?artifact_id=37743

[11] OnEarth WMS tiled extension at http://onearth.jpl.nasa.gov/tiled.html

[12] OpenSearch Specification 1.1 draft 4 http://www.opensearch.org/Specifications/OpenSearch/1.1

[13] OSGeo Tile Map Service Specification version 1.0 at http://wiki.osgeo.org/wiki/Tile_Map_Service_Specification

[14] OSGeo WMS Tile Caching at http://wiki.osgeo.org/wiki/WMS_Tile_Caching

[15] RESTful Web Services. Web Services for the real World. Leonard Richardson and Sam Ruby. O'Reilly Media Inc. 2007.

[16] URI Template, draft-gregorio-uritemplate-03. Network Working Group Internet-Draft. J. Gregorio, Ed. http://tools.ietf.org/html/draft-gregorio-uritemplate-03