ESDS-RFC-006v1
Category: Recommended Standard
Updates/Obsoletes: None

Sam A. Bacharach
June 2007
OpenGIS ® WMS 1.1.1

Open Geospatial Consortium, Inc. OpenGIS ® Web Map Service

1 Status of this Memo

This is a description of an ESDS Community Standard.

Distribution of this memo and the referenced standard is unlimited.

2 Change Explanation

Version 1 - First approved version.

3 Copyright Notice

Copyright © NASA (2007). All Rights Reserved.

4 Abstract

The purpose of this memo is to nominate the OpenGIS® Web Map Service Implementation Specification (WMS) for adoption as a NASA ESDS community standard for disseminating `views` of raster and vector data (“maps”) via the World Wide Web. WMS uses HTTP and defines several operations that allow a client to discover the functions a server is capable of providing, request a specific ”map”, and, optionally, request information about individual features shown on a map.

This nomination is for version 1.1.1 of the WMS specification. Future installations of WMS should consider use of the most recent version. WMS 1.3 is identical to ISO 19128, which the International Organization for Standardization (ISO) released as an International Standard during calendar year 2005.

5 Table of Contents

6 Introduction

The Open Geospatial Consortium, Inc. (OGC) WMS specification is nominated to provide an industry-standard, non-proprietary method of disseminating geospatial information that is supported by many commercial and open source applications. Because of its broad acceptance in the market place, legacy systems can be easily adapted to comply quickly and cheaply. More than 80 companies and organizations, including several that provide open source software, offer solutions for NASA use. The value of WMS is in its simplicity: a single interface is used to issue requests to multiple servers from multiple vendors and open source and then overlay all of the responses for the user to view – 'layers' are returned with transparent backgrounds then simply displayed on top of one another by the client.

The WMS specification consists of two mandatory operations and one optional operation. The (mandatory) GetCapabilities operation lets a client query and learn the capabilities of the service provided by a compliant server. The client parses the returned document and creates a user interface display that lists the available data layers, projections, image file format, and symbolization options. To be compliant a server must return at least one of several formats, including .gif, .png, .jpg, and .svg. It may return other formats too. The (mandatory) GetMap interface lets the client request a georegistered picture (a "map") in one of the image file formats listed in the GetCapabilities reply. Finally, the (optional) GetFeatureInfo operation lets a client request any available information about a pixel location on the map. Users can use the GetFeatureInfo to identify or query the selected

objects from map layers.

Though this nomination is for version 1.1.1 of the WMS specification existing implementations of earlier WMS versions are not made obsolete and it does not require replacing existing 1.0 and 1.1 implementations. Version 1.1.1 is in widespread use within NASA as well as elsewhere. The more recent version, WMS 1.3 is the topic of RFC-005 and its use is encouraged in order to expand the operational experience of that version. All of the versions of the specification provide the same functionality using the same interfaces. The only impact of having a NASA system that has WMS 1.0, WMS 1.1, WMS 1.1.1 and even WMS 1.3, services in it, is on those clients that need to work with all of them. Commercial and open source practice for WMS clients includes the ability to negotiate versioning of the server and send the proper query to each one, which means version specificity is not a factor.

7 Motivation to Adopt the OGC WMS

Why adopt a standard that has been approved by both International Organization for Standardization (ISO) (the premier legally empowered standards body in the world) and the OGC (the leading provider of geospatial implementation specifications)? Two opportunities: reduced costs for NASA and the user, and broader use of NASA data. NASA is under continual pressure to operate more efficiently and at the same time engender greater use of its products and services. A recently published Return on Investment study funded by NASA demonstrates the cost effectiveness of not only using standards, but the WMS Specification itself. That study is found at Appendix C. The service orientation that WMS and the overall OGC Service Oriented Architecture (SOA) bring will, over time, relieve NASA of the need to have a custom viewer for each of its data sources. It will simplify the process of adapting legacy systems to an interoperable architecture and also relieve NASA of the high integration costs of having custom interfaces on each of its new systems. One viewer or a set of viewers for communities of practice will be able to display and fuse data from services that did not even exist when they were provided. Integration via shared, industry-standard interfaces, will be easier and often done automatically by software after a new service is simply registered. The use of WMS will also enable citizens to access and exploit the NASA data from their existing desktop and browser software, driving costs down, and use and benefits up.

WMS is an important step into the world of SOA as defined in the Federal Enterprise Architecture. It provides an inexpensive, straight forward way for NASA to evaluate the applicability of a SOA. WMS and other OGC specifications (listed in Appendix B) will extend the reach and use of NASA data to other government departments and the general citizenship, and will lower barriers between internal scientific uses and external applications.

8 OpenGIS ® Web Map Service Implementation Specification 1.1.1

The specification is included as Attachment A. It can also be found at
http://portal.opengeospatial.org/files/?artifact_id=1081&version=1&format=pdf

9 References

Normative References:

Attachment A - OpenGIS ® Web Map Service Implementation Specification 1.1.1

Informative References:

http://www.opengeospatial.org/specs/?page=abstract

10 Authors' Address

Sam A. Bacharach
35 Main Street
Wayland, MA 01778
USA
Tel: 508-655-5858; fax 703-352-7361
email: sbacharach@opengeospatial.org

11 Appendix A - Glossary of Acronyms

2D	Two Dimensional
AOS	Application Objects Specification
API	Application Programming Interface
AVIRIS	Airborne Visible/Infrared Imaging Spectrometer
BNF	Backus-Naur Form
DAAC	Digital Active Archive Center
ESDS	Earth Science Data Systems
ESE	Earth Science Enterprise
ESIP	Earth Science Information Partner
GC	Grid Coverage
GIS	Geographic Information System
GLOBE	Global Learning and Observations to Benefit the Environment
GML	Geography Markup Language
GO-1	Geographic Objects - Version 1
GSFC	Goddard Space Flight Center
HTTP	Hyper Text Transfer Protocol
JPL	Jet Propulsion Laboratory
OGC™	Open Geospatial Consortium, Inc.
OGCRM	OGC Reference Model
ORM	OGC Reference Model
OpenGIS ®	Registered Trademark of the OGC.
OWS	OpenGIS Web Services
PO	Physical Oceanography
ROI	Return On Investment
SFS	OpenGIS ® Simple Feature Implementation Specification
SLD	OpenGIS ® Styled Layer Descriptor Implementation Specification
SOA	Service Oriented Architecture
SQL	Structured Query Language
SVS	Scientific Visualization System
TDB	Technical Document Baseline
URL	Uniform Resource Locator
WCS	OpenGIS ® Web Coverage Service Implementation Specification

WFS	OpenGIS ® Web Feature Service Implementation Specification
WMC	OpenGIS ® Web Map Context Documents Implementation Specification
WMS	OpenGIS ® Web Map Service Implementation Specification See Attachment A
XML	eXtensible Markup Language

12 Appendix B. OGC's Work

The full collection of OGC specifications is included to illustrate the depth of the work performed by the consortium and to enable an understanding of the 'architecture' of which WMS is a part. As an international consortium OGC has produced a number of specifications that are related which allows a user to assemble them in new and innovative ways. The architecture itself and the overall scope of OGC are described in the OGC Reference Model. The OGCRM is available at the URL listed in Section 9 References . The existence of this architecture and multiple products that support it insures that NASA will enjoy a wide selection of products and options to implement and expand an SOA.

Title	Version	Date	Description
OpenGIS ® Catalog Services Implementation Specification	2.0	2004-08-02	Defines a common interface that enables diverse but conformant applications to perform discovery, browse and query operations against distributed and potentially heterogeneous catalog servers.
OpenGIS ® Coordinate Transformation Services Implementation Specification	1.0	2001-01-	12 Provides interfaces for general positioning, coordinate systems, and coordinate transformations.
OpenGIS ® Filter Encoding Implementation Specification	1.1	2005-05-03	This document defines an XML encoding for filter expressions based on the BNF definition of the OpenGIS Common Catalog Query Language as described in the OpenGIS Catalog Interface Implementation Specification, Version 1.0 [2].
OpenGIS ® Geography Markup Language Encoding Specification	3.1.1	2005-05-03	The Geography Markup Language (GML) is an XML encoding for the transport and storage of geographic information, including both the geometry and properties of geographic features.
OpenGIS ® GO-1 Application Objects (AOS) Implementation Specification	1.0.0	2005-05-04	The GO-1 Application Objects specification defines a set of core packages that support a small set of Geometries, a basic set of renderable Graphics that correspond to those Geometries, 2D device abstractions (displays, mouse, keyboard, etc.), and supporting classes. Implementation of these APIs will support the needs of many users of geospatial and graphic information. These APIs support the rendering of geospatial datasets, provide fine-grained symbolization of geometries, and support dynamic, event and user driven animation of geo-registered graphics.
OpenGIS ® Grid Coverages (GC) Implementation Specification	1.0	2001-01-12	This specification was designed to promote interoperability between software implementations by data vendors and software vendors providing grid analysis and processing capabilities.
OpenGIS ® OGC Web Services Common Implementation Specification (Common)	1.0	2005-05-03	This document specifies many of the aspects that are, or should be, common to all or multiple OWS interface Implementation Specifications. Those specifications currently include the Web Map Service (WMS), Web Feature Service (WFS), and Web Coverage Service

Title	Version	Date	Description
			(WCS). These common aspects include: operation request and response contents; parameters included in operation requests and responses; and encoding of operation requests and responses.
OpenGIS ® Simple Features - SQL (SFS)	1.1	1999-05-05	The Simple Feature Specification application programming interfaces (APIs) provide for publishing, storage, access, and simple operations on Simple Features (point, line, polygon, multi-point, etc).
OpenGIS ® Styled Layer Descriptor (SLD) Implementation Specification	1.0	2002-08-19	The SLD is an encoding for how the Web Map Server (WMS 1.0 & 1.1) specification can be extended to allow user-defined symbolization of feature data.
OpenGIS ® Web Coverage Service (WCS) Implementation Specification	1.0	2003-10-16	Extends the Web Map Server (WMS) interface to allow access to geospatial "coverages" that represent values or properties of geographic locations, rather than WMS generated maps (pictures).
OpenGIS ® Web Feature Service (WFS) Implementation Specification	1.1	2005-05-03	The OGC Web Feature Service (WFS) interface is a collection of operations (implemented as messages carried over HTTP) for retrieving and manipulating geographic features. An implementation of the OGC WFS IS allows a client to retrieve and update geospatial data encoded in Geography Markup Language (GML) from one or more Web Feature Services.
OpenGIS ® Web Map Context Documents (WMC) Implementation Specification	1.1	2005-05-03	This document is a companion specification to the OGC Web Map Service Interface Implementation Specification The present Context specification states how a specific grouping of one or more maps from one or more map servers can be described in a portable, platform-independent format for storage in a repository or for transmission between clients. This description is known as a "Web Map Context Document," or simply a "Context." Presently, context documents are primarily designed for WMS bindings.
OpenGIS ® Web Map Service (WMS) Implementation Specification	1.3	2004-08-02	Provides three operations protocols (GetCapabilities, GetMap, and GetFeatureInfo) in support of the creation and display of registered and superimposed map-like views of information that come simultaneously from multiple sources that are both remote and heterogeneous.
OpenGIS ® Reference Model (ORM)	0.1.2	2003-03-04	The ORM describes a framework for the ongoing work of the OpenGIS Consortium and our specifications and implementing interoperable solutions and applications for geospatial services, data, and applications.
OpenGIS ® Technical Document Baseline (TDB)	1.3	2004-04-22	Spreadsheet of OGC Technical Document Baseline (in update to reflect 05 changes)

Your attention is called to the Styled Layer Descriptor specification which offers a standard way to vary the symbology of a WMS ‘map’ and the WMS Context Document which provides a method

to not only ‘save a session’, but pass that session to another user who can then recreate it on their client. Links to both documents are provided in Appendix D, Related OGC Specifications.

13 Appendix C: NASA ROI study

Dated April 2005 by Booz Allen Hamilton

http://gio.gsfc.nasa.gov/docs/ROI%20Study.pdf

This study examines the use of the WMS specification in NASA and concludes that initial costs are higher than using proprietary interfaces, but that life cycle costs are dramatically reduced and flexibility and extensibility are dramatically increased. There is a reasonable expectation that the implementation premium will disappear as more experience is gained with using the specification.

14 Appendix D: Related Open GIS specifications

OGC Reference Model
http://portal.opengeospatial.org/files/?artifact_id=3836

OpenGIS ® Styled Layer Descriptor Implementation Specification
https://portal.opengeospatial.org/files/?artifact_id=1188

OpenGIS ® Web Map Context Implementation Specification
https://portal.opengeospatial.org/files/?artifact_id=8618

Attachment A. OpenGIS ® Web Map Service Implementation Specification 1.1.1

Open GIS Consortium Inc.

Date: 2002-01-16

Reference number of this OpenGIS® project document: OGC 01-068r3

Version: 1.1.1

Category: OpenGIS® Implementation Specification

Status: Adopted Specification

Editor: Jeff de La Beaujardière

Web Map Service Implementation Specification

Document type: **OpenGIS® Publicly Available Standard**
Document stage: **Adopted Specification**
Document language: **English**

WARNING: *The Open GIS Consortium (OGC) releases this specification to the public without warranty. It is subject to change without notice. This specification is currently under active revision by the OGC Technical Committee*

Requests for clarification and/or revision can be made by contacting the OGC at revisions@opengis.org.

Copyright 1999, 2000, 2001 BBN Technologies
Copyright 1999, 2000, 2001 Cadcorp Ltd.
Copyright 1999, 2000, 2001 CubeWerx Inc.
Copyright 1999, 2000, 2001 IONIC Software s.a.
Copyright 1999, 2000, 2001 Laser-Scan Limited
Copyright 1999, 2000, 2001 SICAD Geomatics GmbH & Co. oHG
Copyright 1999, 2000, 2001 Social Change Online Pty Ltd
Copyright 1999, 2000, 2001 US Army Engineer Research and Development Center

The companies listed above have granted the Open GIS Consortium, Inc. (OGC) a nonexclusive, royalty-free, paid up, worldwide license to copy and distribute this document and to modify this document and distribute copies of the modified version.

This document does not represent a commitment to implement any portion of this specification in any company's products.

OGC's Legal, IPR and Copyright Statements are found at http://www.opengis.org/legal/ipr.htm

NOTICE

Permission to use, copy, and distribute this document in any medium for any purpose and without fee or royalty is hereby granted, provided that you include the above list of copyright holders and the entire text of this NOTICE.

We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of OGC documents is granted pursuant to this license. However, if additional requirements (as documented in the Copyright FAQ at http://www.opengis.org/legal/ipr_faq.htm) are satisfied, the right to create modifications or derivatives is sometimes granted by the OGC to individuals complying with those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

RESTRICTED RIGHTS LEGEND. Use, duplication, or disclosure by government is subject to restrictions as set forth in subdivision (c)(1)(ii) of the Right in Technical Data and Computer Software Clause at DFARS 252.227.7013

OpenGIS® is a trademark or registered trademark of Open GIS Consortium, Inc. in the United States and in other countries.

Contents

© OGC 2001 – All rights reserved

© OGC 2001 – All rights reserved

i. Preface

This document is primarily a correction and clarification of the OpenGIS Web Map Service Interface Implementation Specification version 1.1.0 [4], hereinafter "WMS 1.1.0." Substantive differences between the present specification and its predecessor are summarized in the Foreword and are called out in the text where appropriate.

Web Mapping within the OGC was first described in "WWW Mapping Framework" [5]. The first OGC consensus position of the WWW Mapping Special Interest Group, a core task force of the OGC, is described in "User Interaction with Geospatial Data" [2]. From these documents, as well as from "A Web Mapping Scenario" [7], an OGC-sponsored initiative was begun. That initiative, known as the Web Mapping Testbed (WMT), was first described in a Request For Technology (RFT) [10] and then in a Request for Quotation (RFQ) [11].

The WMT Phase I process culminated in the OpenGIS Web Map Service Interface Implementation Specification version 1.0.0 [6], hereinafter "WMS 1.0.0." That first version supported basic interoperability of simple map servers and clients, but did not fully address access to Simple Features, Coverages, data with temporal or other dimensions, and other types of geoprocessing services. Many of these elements were addressed in the follow-on Web Mapping Testbed phase 2 (WMT2) and the Geospatial Fusion Services Testbed. WMS 1.1.0 was a result of WMT2.

ii. Submitting Organizations

The OGC Web Map Service Revision Working Group submits this Implementation Specification to the OGC Technical Committee as a Revision to Web Map Service Interface Implementation Specification version 1.1.0.

iii. Submission Contact Points

All questions regarding this submission should be directed to the Editor or to the WWW Mapping SIG chair:

Jeff de La Beaujardière (Editor)
NASA Goddard Space Flight Center

© OGC 2001 – All rights reserved

Code 933
Greenbelt MD 20771 USA
+1 301 286 1569
delabeau@iniki.gsfc.nasa.gov

Allan Doyle (WWW Mapping SIG Chair)
International Interfaces, Inc.
948 Great Plain Ave. PMB-182
Needham, MA 02492 USA
+1 781 433 2695
adoyle@intl-interfaces.com

Additional Contributors

Craig Bruce
CubeWerx
csbruce@cubewerx.com

Adrian Cuthbert
adrian.cuthbert@bigfoot.com

Allan Doyle
International Interfaces
adoyle@intl-interfaces.com

John Evans,
GST, Inc./NASA GSFC
john.evans@gsfc.nasa.gov

George Percivall
GST, Inc./NASA GSFC
percivall@gsfc.nasa.gov

Arliss Whiteside
BAE SYSTEMS Mission Solutions
Arliss.Whiteside@baesystems.com

© OGC 2001 – All rights reserved

iv. Revision History

Date	Release	Editor	Description
2000-04-19	1.0.0	Allan Doyle	First WMS Implementation Specification (OGC document #00-028)
2001-06-21	1.1.0	Jeff de La Beaujardière	Revised edition (OGC document #01-047r2)
2001-12-12	1.1.1	Jeff de La Beaujardière	Minor revision (OGC document #01-068r3)

v. Changes to the OpenGIS Abstract Specification

The OpenGIS© Abstract Specification requires the following change to accommodate this OpenGIS© standard:

- The brief description of the Web Map Service presently found in Abstract Specification Topic 12, "Service Architecture," should be augmented.

The needed material is expected to emerge in part from the Architecture thread of the OGC Web Services testbed.

© OGC 2001 – All rights reserved

Foreword

Attention is drawn to the possibility that some of the elements of this part of OGC 01-068r3 may be the subject of patent rights. Open GIS Consortium Inc. shall not be held responsible for identifying any or all such patent rights.

This edition cancels and replaces the previous edition (OGC 01-047r2), which has been technically revised.

Summary of Changes from Version 1.1.0

1. The text in Section 6.5.5.1 regarding the EPSG:4326 spatial reference system has been revised in response to concerns raised by the OGC Coordinate Transformation working group using new text provided by that group. In that Section and elsewhere the phrase "coordinate system" has been replaced with "coordinate reference system" in keeping with the usage in other OGC documents.

2. An optional and recommended change has been made to the use of <SRS> elements in Capabilities XML. WMS 1.1.0 allowed a whitespace-separated list of Spatial Reference System identifiers inside a single <SRS>. This revision allows a sequence of SRS elements, each containing a single identifier, and deprecates the whitespace-separated list encoding.

3. The use of the suffix "Z" in ISO 8601:1988(E) time strings in UTC has been made mandatory instead of recommended. Annex B now more clearly states where it has extended ISO 8601.

4. Section 6.5.5.1 has been clarified regarding the order of values in the BBOX request parameter.

5. The former Section 7.1.5 has been renumbered 7.1.4. Section 7.1.4.4 ("Layers and Styles") has been rewritten for clarity. A new Section 7.1.4.5 ("Layer Properties") has been added. Some informative material that was previously found only the Capabilities DTD has been copied into this specification document.

6. Table 7, "Inheritance of Layer Properties," has been substantially revised for clarity. Text has been added, and material previously in the Comments column has been moved to appropriate subsections in Section 7.1.4.5.

7. For the use of the Styled Layer Descriptor specification, three new optional operations are named, but not otherwise specified, in this document (GetLegendGraphic, GetStyles, PutStyles).

© OGC 2001 – All rights reserved

8. The use of reserved characters in HTTP GET URLs has been clarified. This change inserts a new Section 6.2.1 and Table 1, renumbering later portions accordingly.

9. The implicit permission for servers to reference private copy of DTD in the Capabilities XML has been made explicit (Section 7.1.4).

10. Text has been added to 7.2.3.7 ("FORMAT") regarding acceptable and recommended output formats for GetMap requests. This section has also been moved to appear before the section on output width and height.

11. The fact that the XML format for reporting exceptions is required has been clarified. (Section 7.2.3.11).

12. Exception Codes defined by this document are now summarized in Appendix A.3.

13. Mention of the optional test layer WMT_GRATICULE (former Section 7.1.4.7) has been deleted, as it was found in practice that it was error-prone and of little use in diagnosing alignment errors in other layers.

14. The discussion regarding maps that span the anti-meridian and whose X axis is longitude has been made more permissive (Section 6.5.6).

15. The sample GetMap request using a default style has been corrected. (Introduction). An error regarding the list of styles in a GetMap request has been corrected (Section 7.2.3.4).

16. The role of each GetMap request parameter has been clarified in Section 7.2.3, and the name of each sub-clause therein has been shortened. The role of each GetFeatureInfo request parameter has been clarified in Section 7.3.3.

17. Text in Section 7.3.3.7 concerning the default value of FEATURE_COUNT which contradicted the information in Table 8 has been corrected to match Table 8, clearly making the default value be 1 rather than arbitrary.

18. In Section 6.4.1 ("Parameter Ordering and Case"), the text about unknown parameters in requests has been loosened (from "shall ignore" to "shall not require").

19. Text has been added to Section v concerning UML and the OGC Abstract Specification.

20. Annex E ("Automatic Projections") has been added.

21. Annex F ("Future Work") has been added for informational purposes.

22. Text has been added to Annex D concerning the OGC Conformance Testing Program. Mention of ISO 19105 has been removed from Clause 2.

© OGC 2001 – All rights reserved

23. The list of terms and definitions (Section 4) has been augmented, and has been reformatted according to ISO practice.

24. The material previously in the Introduction has been moved to the Scope clause, and the Introduction has been shortened to less than one page to conform to ISO practice.

25. The list of Contributors to this document has been augmented and moved to Section iii ("Submission Contact Points").

26. The declaration and citing of normative references has been modified to better conform to ISO practice. The list of normative references has been augmented to reflect implicit mentions in the text. The names of authors of several references have been corrected.

27. The sample XML (informative) in Annex A.2 has been corrected to match the Service Name required by Section 7.5.1.2.

28. The normative verb "must" has been replaced by "shall" to conform to ISO practice ("shall" means something is required by the standard, "must" that something is required by law).

Normative Annexes

Annexes A, B, C,. D and E are normative, except that Subsections A.2 and A.4 are informative. Annex F is informative.

© OGC 2001 – All rights reserved

Introduction

A Web Map Service (WMS) produces maps of georeferenced data. We define a "map" as a visual representation of geodata; a map is not the data itself. This specification defines three WMS operations: **GetCapabilities** returns service-level metadata, which is a description of the service's information content and acceptable request parameters; **GetMap** returns a map image whose geospatial and dimensional parameters are well-defined; **GetFeatureInfo** (optional) returns information about particular features shown on a map.

This specification defines a syntax for World Wide Web (WWW) Uniform Resource Locators (URLs) that invoke each of these operations. Also, an Extensible Markup Language (XML) encoding is defined for service-level metadata.

When requesting a map, a client may specify the information to be shown on the map (one or more "Layers"), possibly the "Styles" of those Layers, what portion of the Earth is to be mapped (a "Bounding Box"), the projected or geographic coordinate reference system to be used (the "Spatial Reference System," or SRS), the desired output format, the output size (Width and Height), and background transparency and color.

When two or more maps are produced with the same Bounding Box, Spatial Reference System, and output size, the results can be accurately layered to produce a composite map. The use of image formats that support transparent backgrounds allows the lower Layers to be visible. Furthermore, individual map Layers can be requested from different Servers. The WMS specification thus enables the creation of a network of distributed Map Servers from which Clients can build customized maps.

A particular WMS provider in a distributed WMS network need only be the steward of its own data collection. This stands in contrast to vertically-integrated web mapping sites that gather in one place all of the data to be made accessible by their own private interface.

© OGC 2001 – All rights reserved

Web Map Service Implementation Specification

1 Scope

This OpenGIS® Standard specifies the behavior of a service that produces georeferenced maps. This standard specifies operations to retrieve a description of the maps offered by a service instance, to retrieve a map, and to query a server about features displayed on a map.

This OpenGIS® Standard is applicable to pictorial renderings of maps in a graphical format. This standard is *not* applicable to retrieval of actual feature data or coverage data values.

A Web Map Service produces maps of georeferenced data. We define a "map" as a visual representation of geodata; a map is not the data itself. These maps are generally rendered in a pictorial format such as PNG, GIF or JPEG, or occasionally as vector-based graphical elements in Scalable Vector Graphics (SVG) or Web Computer Graphics Metafile (WebCGM) formats. This specification standardizes the way in which maps are requested by clients and the way that servers describe their data holdings. This document defines three operations, the first two of which are required of every WMS.

GetCapabilities (required): Obtain service-level metadata, which is a machine-readable (and human-readable) description of the WMS's information content and acceptable request parameters.

GetMap (required): Obtain a map image whose geospatial and dimensional parameters are well-defined.

GetFeatureInfo (optional): Ask for information about particular features shown on a map.

A standard web browser can ask a Web Map Service to perform these operations simply by submitting requests in the form of Uniform Resource Locators (URLs) [IETF RFC 2396]. The content of such URLs depends on which of the tasks is requested. All URLs include a specification version number and a request type parameter. In addition, when invoking GetMap a WMS Client can specify the information to be shown on the map (one or more "Layers"), possibly the "Styles" of those Layers, what portion of the Earth is to be mapped (a "Bounding Box"), the projected or geographic coordinate reference system to be used (the "Spatial Reference System," or SRS), the desired output format, the output size (Width and Height), and background transparency and color. When

invoking GetFeatureInfo the Client indicates what map is being queried and which location on the map is of interest.

When two or more maps are produced with the same Bounding Box, Spatial Reference System, and output size, the results can be accurately layered to produce a composite map. The use of image formats that support transparent backgrounds (e.g., GIF or PNG) allows the lower Layers to be visible. Furthermore, individual map Layers can be requested from different Servers. The WMS GetMap operation thus enables the creation of a network of distributed Map Servers from which Clients can build customized maps.

A particular WMS provider in a distributed WMS network need only be the steward of its own data collection. This stands in contrast to vertically-integrated web mapping sites that gather in one place all of the data to be made accessible by their own private interface.

Because each WMS is independent, a WMS must be able to provide a machine-readable description of its capabilities. This "Service Metadata" enables Clients to formulate valid requests and enables the construction of searchable catalogs that can direct clients to particular WMSes.

A WMS may optionally allow the GetFeatureInfo operation. If it does, its maps are said to be "queryable," and a Client can request information about features on a map by adding to the map URL additional parameters specifying a location (as an X, Y offset from the upper left corner) and the number of nearby features about which to return information.

Cascading Map Servers

A "Cascading Map Server" is a WMS that behaves like a client of other WMSes and behaves like a WMS to other clients. For example, a Cascading Map Server can aggregate the contents of several distinct map servers into one service. Furthermore, a Cascading Map Server can perform additional functions such as output format conversion or coordinate transformation on behalf of other servers.

Styled Layer Descriptors

This specification applies to a Web Map Service that publishes its ability to produce maps rather than its ability to access specific data holdings. A basic WMS classifies its georeferenced information holdings into "Layers" and offers a finite number of predefined "Styles" in which to display those layers.

The behavior of a Web Map Service can be extended to allow user-defined symbolization of feature data instead of named Layers and Styles. The Styled Layer Descriptor (SLD) specification [3] describes this extension. In brief, an SLD-enabled WMS retrieves features from a Web Feature Service [8] and applies explicit styling information provided by the user in order to render a map.

© OGC 2001 – All rights reserved

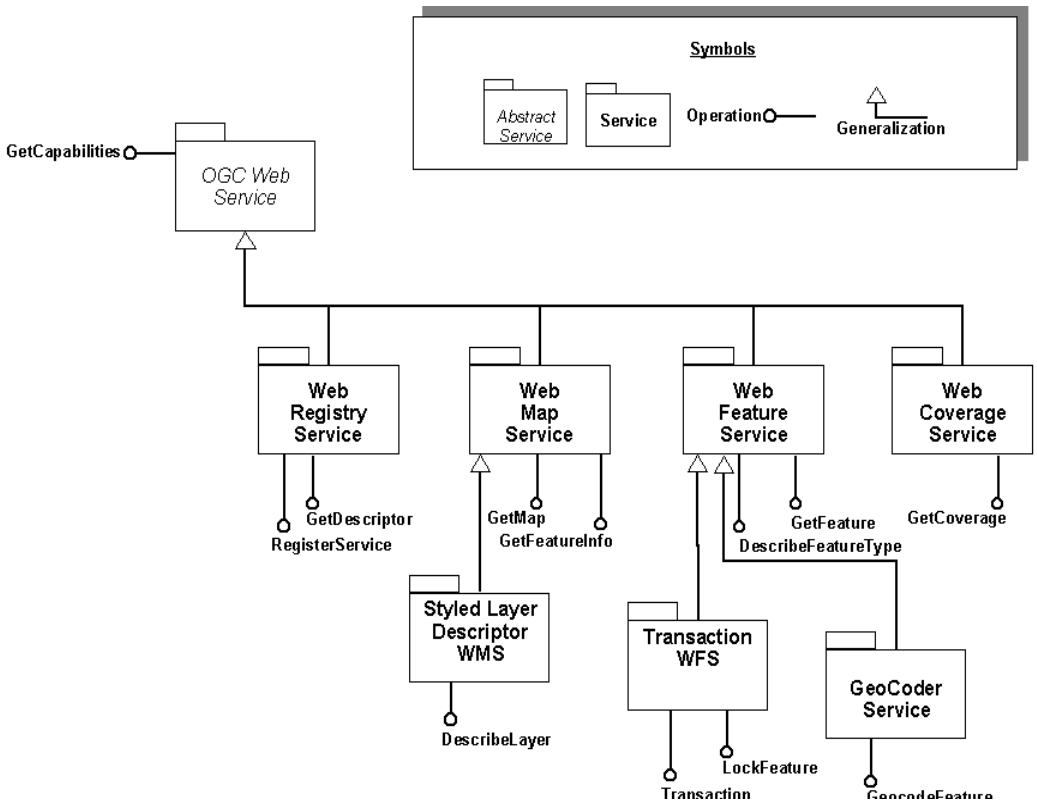
An SLD WMS adds the following additional operations that are not available on a basic WMS:

- DescribeLayer
- GetLegendGraphic
- GetStyles
- PutStyles

Relation to other OGC Web Services

The OGC Web Services (OWS) suite includes three principal types of georeferenced information access services: Web Map Server (WMS), Web Coverage Server (WCS), and Web Feature Server (WFS). In addition, there are services such as GeoParser and GeoCoder that return spatially referenced results. Figure 1 is an architecture diagram showing conceptually how some of the OGC Web Services are related, and naming some (not all) of the operations they define.

Figure 1 — OGC Web Services Architecture diagram



© OGC 2001 – All rights reserved

Web Mapping Examples

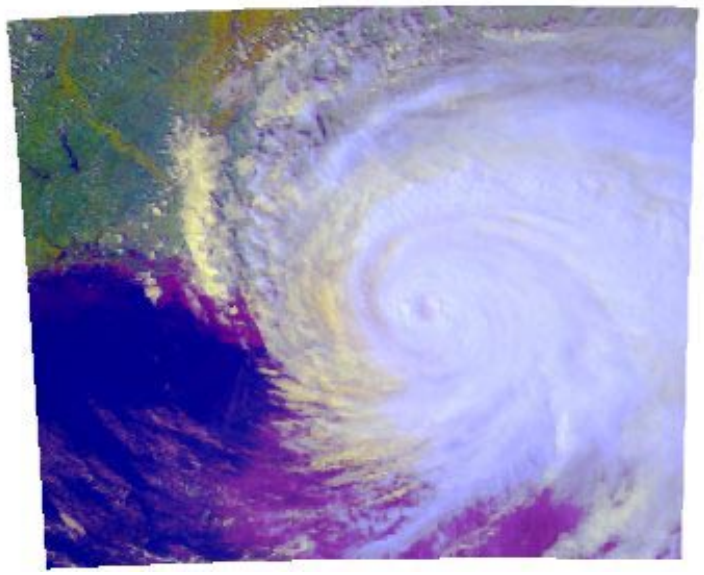
We conclude this Introduction with some illustrative URLs and their resulting maps.

Example 1: One Server, One Layer, Default Style

The following hypothetical URL requests the US National Oceanographic and Atmospheric Administration hurricane image shown in Figure 2:

```
http://a-map-co.com/mapserver.cgi?VERSION=1.1.0&REQUEST=GetMap&
SRS=EPSG:4326&BBOX=-97.105,24.913,78.794,36.358&
WIDTH=560&HEIGHT=350&LAYERS=AVHRR-09-27&STYLES=&
FORMAT=image/png&BGCOLOR=0xFFFFFF&TRANSPARENT=TRUE&
EXCEPTIONS=application/vnd.ogc.se_inimage
```

Figure 2 — NOAA Hurricane Image of the Gulf of Mexico



Example 2: One Server, Three Layers, Named Styles

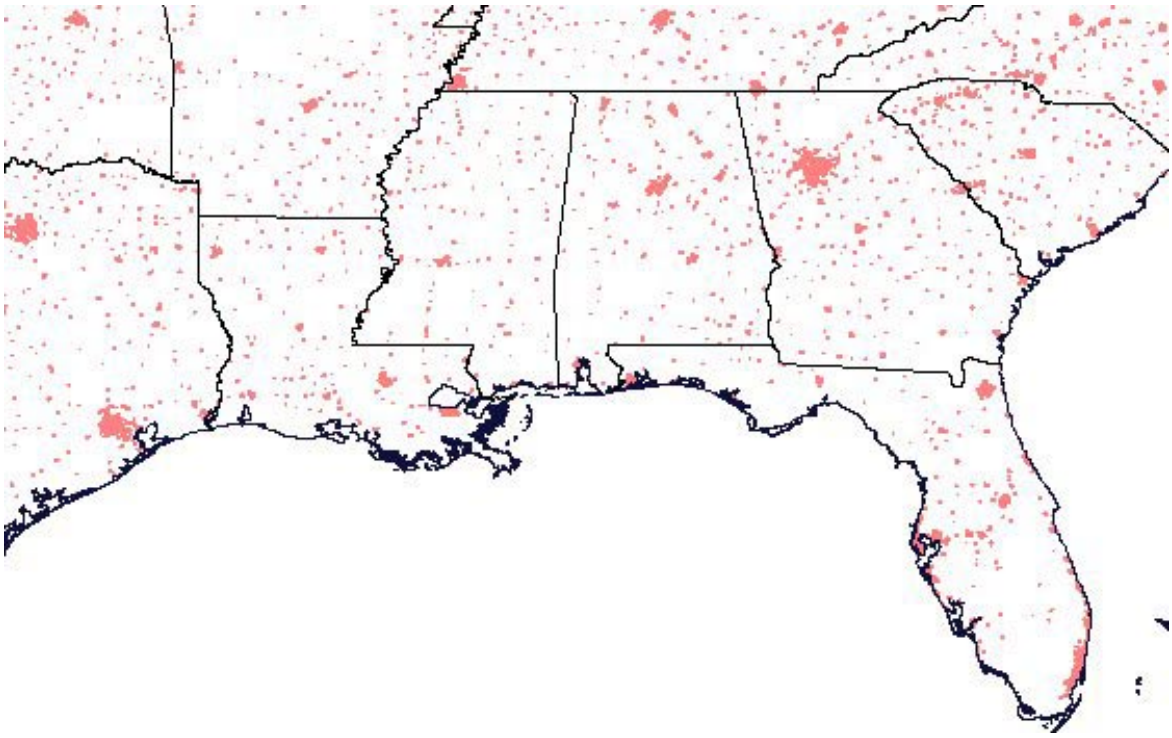
The following hypothetical URL requests three layers--built-up areas, coastlines, and political boundaries --to produce the map shown in Figure 3:

```
http://b-maps.com/map.cgi?VERSION=1.1.0&REQUEST=GetMap&
SRS=EPSG:4326&BBOX=-97.105,24.913,78.794,36.358&
WIDTH=560&HEIGHT=350&LAYERS=BUILTUPA_1M,COASTL_1M,POLBNDL_1M&
```


© OGC 2001 – All rights reserved

```
STYLES=0XFF8080,0X101040,BLACK&FORMAT=image/png&BGCOLOR=0xFFFFFF&
TRANSPARENT=TRUE&EXCEPTIONS=application/vnd.ogc.se_inimage
```

Figure 3 — Political, Coastline, and Populated Areas, Southeastern United States



Notice that in both of these URLs the spatial information is identical:

```
SRS=EPSG:4326&BBOX=-97.105,24.913,78.794,36.358
&WIDTH=560&HEIGHT=350.
```

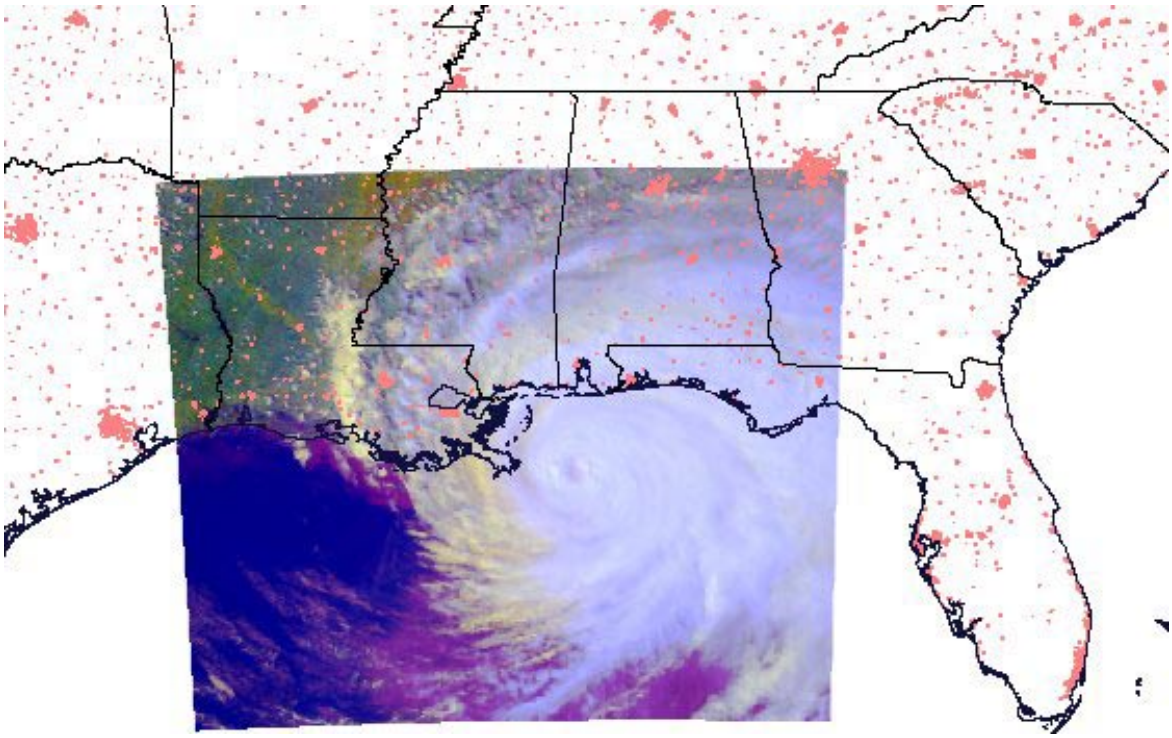
The second map, for which a transparent background was requested (TRANSPARENT=TRUE), can therefore be precisely overlaid on the first.

Example 3: Two Servers, Four Layers

Figure 4 shows the result of overlaying Figure 2 on Figure 3 to produce a composite map from two separate Map Servers.

© OGC 2001 – All rights reserved

Figure 4 — Combined Hurricane Image and Population Map



2 Conformance

Conformance with this specification shall be checked using all the relevant tests specified in Annex D (normative).

3 Normative references

The following normative documents contain provisions that, through reference in this text, constitute provisions of this specification. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this specification are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies.

CGI, *The Common Gateway Interface,* National Center for Supercomputing Applications, <http://hoohoo.ncsa.uiuc.edu/cgi/>

EPSG, *European Petroleum Survey Group Geodesy Parameters*, Lott, R., Ravanas, B., Cain, J., Girbig, J.-P., and Nicolai, R., eds., <http://www.epsg.org/>

© OGC 2001 – All rights reserved

FGDC-STD-001-1988, *Content Standard for Digital Geospatial Metadata (version 2),* US Federal Geographic Data Committee, <http://www.fgdc.org/metadata/contstan.html>

IETF RFC 2045 (November 1996), *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*, Freed, N. and Borenstein N., eds., <http://www.ietf.org/rfc/rfc2045.txt>

IETF RFC 2119 (March 1997), *Key words for use in RFCs to Indicate Requirement Levels*, Bradner, S., ed., <ftp://ftp.isi.edu/in-notes/rfc2119.txt>.

IETF RFC 2616 (June 1999), *Hypertext Transfer Protocol – HTTP/1.1*, Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and Berners-Lee, T., eds., <http://www.ietf.org/rfc/rfc2616.txt>

IETF RFC 2396 (August 1998), *Uniform Resource Identifiers (URI): Generic Syntax*, Berners-Lee, T., Fielding, N., and Masinter, L., eds., <http://www.ietf.org/rfc/rfc2396.txt>

ISO 8601:1988(E), *Data elements and interchange formats - Information interchange - Representation of dates and times*.

ISO 19115, *Geographic information — Metadata*

OGC AS 12 (September 2001), *The OpenGIS Abstract Specification Topic 12: OpenGIS Service Architecture (Version 4.2)*, Kottman, C. (ed.), <http://www.opengis.org/techno/specs.htm>

UCUM, *Unified Code for Units of Measure,* Schadow, G. and McDonald, C. J. (eds.), <http://aurora.rg.iupui.edu/~schadow/units/UCUM/>

XML 1.0 (October 2000), *Extensible Markup Language (XML) 1.0* (*2nd edition)*, World Wide Web Consortium Recommendation, Bray, T., Paoli, J., Sperberg-McQueen, C.M., and Maler, E., eds., <http://www.w3.org/TR/2000/REC-xml>

4 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

4.1
operation
specification of a transformation or query that an object may be called to execute [OGC AS 12]

4.2
interface
named set of **operations** that characterize the behavior of an entity [OGC AS 12]

© OGC 2001 – All rights reserved

4.3
service
distinct part of the functionality that is provided by an entity through **interfaces** [OGC AS 12]

4.4
service instance
server
actual implementation of a **service**

4.5
client
software component that can invoke an **operation** from a **server**

4.6
request
invocation of an **operation** by a **client**

4.7
response
result of an **operation** returned from a **server** to a **client**

4.8
map
pictorial representation of geographic data

4.9
spatial reference system
a projected or geographic coordinate reference system

4.10
capabilities XML
service-level metadata describing the **operations** and content available at a **service instance**.

5 Conventions

5.1 Normative verbs

In the sections labeled as normative, the key words "**required**", "**shall**", "**shall not**", "**should**", "**should not**", "**recommended**", "**may**", and "**optional**" in this document are to be interpreted as described in [IETF RFC 2119].

The verb "**deprecate**" provides notice that the referenced portion of the specification is being retained for backwards compatibility with earlier versions but may be removed from a future version of the specification without further notice.

© OGC 2001 – All rights reserved

5.2 Abbreviated Terms

CGI	Common Gateway Interface
DCP	Distributed Computing Platform
DTD	Document Type Definition
EPSG	European Petroleum Survey Group
GIF	Graphics Interchange Format
GIS	Geographic Information System
GML	Geography Markup Language
HTTP	Hypertext Transfer Protocol
IETF	Internet Engineering Task Force
JPEG	Joint Photographic Experts Group
MIME	Multipurpose Internet Mail Extensions
OGC	Open GIS Consortium
OWS	OGC Web Service
PNG	Portable Network Graphics
RFC	Request for Comments
SLD	Styled Layer Descriptor
SVG	Scalable Vector Graphics
URL	Uniform Resource Locator
WebCGM	Web Computer Graphics Metafile
WCS	Web Coverage Service
WFS	Web Feature Service
WMS	Web Map Service
XML	Extensible Markup Language

6 Basic Service Elements

This clause specifies aspects of Web Map Server behavior (more generally, of OGC Web Service behavior) that are independent of particular operations or are common to several operations.

6.1 Version Numbering and Negotiation

6.1.1 Version Number Form

The published specification version number contains three positive integers, separated by decimal points, in the form "x.y.z". The numbers "y" and "z" will never exceed 99. Each OWS specification is numbered independently.

6.1.2 Version Changes

A particular specification's version number **shall** be changed with each revision. The number **shall** increase monotonically and **shall** comprise no more than three integers separated by decimal points, with the first integer being the most significant. There may

© OGC 2001 – All rights reserved

be gaps in the numerical sequence. Some numbers may denote experimental or interim versions. Service instances and their clients need not support all defined versions, but **shall** obey the negotiation rules below.

6.1.3 Appearance in Requests and in Service Metadata

The version number appears in at least two places: in the Capabilities XML describing a service, and in the parameter list of client requests to that service. The version number used in a client's request of a particular service instance **shall** be equal to a version number which that instance has declared it supports (except during negotiation as described below). A service instance may support several versions, whose values clients may discover according to the negotiation rules.

6.1.4 Version Number Negotiation

An OWS Client may negotiate with a Service Instance to determine a mutually agreeable specification version. Negotiation is performed using the GetCapabilities operation (described in Section 7.1) according to the following rules.

All Capabilities XML **shall** include a protocol version number. In response to a GetCapabilities request containing a version number, an OGC Web Service **shall** either respond with output that conforms to that version of the specification, **or** negotiate a mutually agreeable version if the requested version is not implemented on the server. If no version number is specified in the request, the server **shall** respond with the highest version it understands and label the response accordingly.

Version number negotiation occurs as follows:

1) If the server implements the requested version number, the server **shall** send that version.

2a) If a version unknown to the server is requested, the server **shall** send the highest version less than the requested version.

2b) If the client request is for a version lower than any of those known to the server, then the server **shall** send the lowest version it knows.

3a) If the client does not understand the new version number sent by the server, it **may** either cease communicating with the server **or** send a new request with a new version number that the client does understand but which is less than that sent by the server (if the server had responded with a lower version).

3b) If the server had responded with a higher version (because the request was for a version lower than any known to the server), and the client does not understand the proposed higher version, then the client **may** send a new request with a version number higher than that sent by the server.

© OGC 2001 – All rights reserved

The process is repeated until a mutually understood version is reached, or until the client determines that it will not or cannot communicate with that particular server.

Example 1: Server understands versions 1, 2, 4, 5 and 8. Client understands versions 1, 3, 4, 6, and 7. Client requests version 7. Server responds with version 5. Client requests version 4. Server responds with version 4, which the client understands, and the negotiation ends successfully.

Example 2: Server understands versions 4, 5 and 8. Client understands version 3. Client requests version 3. Server responds with version 4. Client does not understand that version or any higher version, so negotiation fails and client ceases communication with that server.

6.2 General HTTP Request Rules

At present, the only distributed computing platform (DCP) explicitly supported by OGC Web Services is the World Wide Web itself, or more specifically Internet hosts implementing the Hypertext Transfer Protocol (HTTP) [IETF RFC 2616]. Thus the Online Resource of each operation supported by a service instance is an HTTP Uniform Resource Locator (URL). The URL may be different for each operation, or the same, at the discretion of the service provider. Each URL **shall** conform to the description in [IETF RFC 2616] (Section 3.2.2 "HTTP URL") but is otherwise implementation-dependent; only the parameters comprising the service request itself are mandated by the OGC Web Services specifications.

HTTP supports two request methods: GET and POST. One or both of these methods may be defined for a particular OGC Web Service type and offered by a service instance, and the use of the Online Resource URL differs in each case. The basic WMS specification only defines HTTP GET for invoking operations. (A Styled Layer Descriptor WMS [3] defines HTTP POST for some operations.)

6.2.1 Reserved characters in HTTP GET URLs

The URL specification [IETF RFC 2396] reserves particular characters as significant and requires that these be escaped when they might conflict with their defined usage. The present WMS specification explicitly reserves several of these characters for use in the query portion of HTTP GET requests. When the characters "?", "&", "=", "/", ":" and "," appear in one of the roles defined in Table 1, they are to appear literally in the URL. When such characters appear elsewhere (for example, in the value of a parameter), they are to be encoded as defined in [IETF RFC 2396].

Table 1 — Reserved Characters in HTTP GET Query

Character	Reserved Usage
?	Separator indicating start of query string.
&	Separator between parameters in query string.

© OGC 2001 – All rights reserved

=	Separator between name and value of parameter.
/	Separator between MIME type and subtype in format parameter value.
:	Separator between Namespace and Identifier in SRS parameter value.
,	Separator between individual values in list-oriented parameters.

6.2.2 HTTP GET

An Online Resource URL intended for HTTP GET requests is in fact only a URL prefix to which additional parameters are appended in order to construct a valid Operation request. A URL prefix is defined as an opaque string including the protocol, hostname, optional port number, path, a question mark '?', and, **optionally**, one or more server-specific parameters ending in an ampersand '&'. The prefix uniquely identifies the particular service instance. A client appends the necessary request parameters as name/value pairs in the form "name=value&". The resulting URL **shall** be valid according to the HTTP Common Gateway Interface standard [CGI], which mandates the presence of '?' before the sequence of query parameters and the '&' between each parameter.

The URL prefix **shall** end in either a '?' (in the absence of additional server-specific parameters) or a '&'. In practice, however, Clients **should** be prepared to add a necessary trailing '?' or '&' before appending the Operation parameters defined in this specification in order to construct a valid request URL.

Table 2 summarizes the components of an operation request URL.

Table 2 — A general OGC Web Service Request

URL Component	Description
http://host[:port]/path?{name[=value]&}	URL prefix of service operation. [ ] denotes 0 or 1 occurrence of an optional part; {} denotes 0 or more occurrences. The prefix is entirely at the discretion of the service provider.
name=value&	One or more standard request parameter name/value pairs defined by an OGC Web Service. The actual list of required and optional parameters is mandated for each operation by the appropriate OWS specification.

6.2.3 HTTP POST

An Online Resource URL intended for HTTP POST requests is a complete and valid URL to which Clients transmit request parameters in the body of the POST request. A WMS **shall not** require additional parameters to be appended to the URL in order to construct a valid target for the Operation request.

© OGC 2001 – All rights reserved

Operation requests using HTTP POST have not yet been defined for the basic Web Map Service.

6.3 General HTTP Response Rules

Upon receiving a valid request, the service **shall** send a response corresponding exactly to the request as detailed in the appropriate specification. Only in the case of Version Negotiation (described above) may the server offer a differing result.

Upon receiving an invalid request, the service **shall** issue a Service Exception as described in Section 6.7.

NOTE: As a practical matter, in the WWW environment a client should be prepared to receive either a valid result, or nothing, or any other result. This is because the client may itself have formed a non-conforming request that inadvertently triggered a reply by something other than an OGC Web Service, because the Service itself may be non-conforming, etc.

Response objects **shall** be accompanied by the appropriate Multipurpose Internet Mail Extensions (MIME) type [IETF RFC 2045] for that object. Allowable types for operation responses and service exceptions are discussed below.

Response objects **should** be accompanied by other HTTP entity headers as appropriate and to the extent possible. In particular, the Expires and Last-Modified headers provide important information for caching; Content-Length may be used by clients to know when data transmission is complete and to efficiently allocate space for results, and Content-Encoding or Content-Transfer-Encoding may be necessary for proper interpretation of the results.

6.4 Request Parameter Rules

6.4.1 Parameter Ordering and Case

Parameter names **shall not** be case sensitive, but parameter values **shall** be case sensitive. In this document, parameter names are typically shown in uppercase for typographical clarity, not as a requirement.

Parameters in a request **may** be specified in any order.

An OGC Web Service **shall** be prepared to encounter parameters that are not part of this specification. In terms of producing results per this specification, an OGC Web Service **shall not** require such parameters.

6.4.2 Parameter Lists

Parameters consisting of lists (for example, the LAYERS and STYLES in WMS GetMap) **shall** use the comma (",") as the separator between items in the list. Additional white space **shall not** be used to delimit list items. If a parameter value includes a space or comma, it **shall** be escaped using the URL encoding rules [IETF RFC 2396].

© OGC 2001 – All rights reserved

Individual entries in a list **may** be empty, as represented by two successive commas (",,").

6.5 Common Request Parameters

6.5.1 VERSION

The VERSION parameter specifies the protocol version number. The format of the version number, and version negotiation, are described in Section 6.1.

6.5.2 REQUEST

The REQUEST parameter indicates which service operation is being invoked. The value **shall** be the name of one of the operations offered by the OGC Web Service Instance.

6.5.3 FORMAT

The FORMAT parameter specifies the output format of the response to an operation.

An OGC Web Service **may** offer only a subset of the formats known for that type of operation, but the server **shall** advertise in its Capabilities XML those formats it does support and **shall** accept requests for any format it advertises. A Service Instance **may** optionally offer a new format not previously offered by other instances, with the recognition that clients are not required to accept or process an unknown format. If a request contains a Format not offered by a particular server, the server **shall** throw a Service Exception (with code "InvalidFormat").

A Client **may** accept only a subset of the formats known for that type of operation. If a Client and Service do not support any mutually agreeable formats, the Client may, at its discretion, cease communicating with that service, or search for an intermediary service provider that performs format conversion, or allow the user to choose other disposition methods (e.g., saving to local storage or passing to helper application).

Formats are expressed in both Capabilities XML and in operation requests using MIME types. Each Operation has a distinct list of supported formats. Some formats may be offered by several operations, and are then duplicated as needed in each list.

Generally, OGC Web Service MIME types are chosen from among those in common use on the Internet [9]. However, additional OGC-specific types have been adopted to distinguish among different types of XML-formatted content (the generic XML MIME types being text/xml and application/xml), as listed in Table 3.

Table 3 — OGC-Specific MIME Types

MIME Type	Document Content
application/vnd.ogc.wms_xml	WMS Capabilities XML
application/vnd.ogc.gml	Geography Markup Language XML [1]
application/vnd.ogc.se_xml	Service Exception XML

© OGC 2001 – All rights reserved

application/vnd.ogc.se_inimage	Image overwritten with Exception message.
application/vnd.ogc.se_blank	Blank image because Exception occurred.

6.5.4 EXCEPTIONS

The EXCEPTIONS parameter states the format in which to report errors. See Section 6.7 on Service Exceptions, below.

6.5.5 Spatial Reference System

The Spatial Reference System (SRS) is a text parameter that names a horizontal coordinate reference system code. The name includes a namespace prefix, a colon, a numeric identifier, and possibly a comma followed by additional parameters. This specification defines two namespaces, EPSG and AUTO, which are discussed below.

NOTE: The use of the term SRS is in keeping with WMS 1.0.0. A more modern definition uses Coordinate Reference System (CRS) when referring to spatial referencing by coordinates, and SRS when referring to spatial referencing by addresses or indexes.

OGC Web Services are **not** required to support all possible SRSes, but **shall** advertise in their Capabilities XML those projections which they do offer and **shall** accept requests for all advertised projections. If a request contains an SRS not offered by a particular server, the server **shall** throw a Service Exception (code = "InvalidSRS").

Clients are **not** required to support all possible SRSes. If a Client and Service do not support any mutually agreeable SRS, the Client may, at its discretion, cease communicating with that service, or search for an intermediary service provider that performs coordinate transformations, or allow the user to choose other disposition methods.

6.5.5.1 EPSG Namespace for SRS

The EPSG namespace makes use of the European Petroleum Survey Group tables [EPSG], which define numeric identifiers (the EPSG "CRS code," corresponding to the field "COORD_REF_SYS_CODE" in the EPSG database) for many common projections and which associate projection or coordinate metadata (such as measurement units or central meridian) for each identifier. An SRS name in the EPSG namespace includes only the prefix and the identifier, not any additional parameters. This format is used both as the value of the SRS parameter in a service request and as the value of an <SRS> element in the Capabilities XML.

When the SRS parameter specifies a Geographic Coordinate Reference System, e.g., "EPSG:4326", the returned image is implicitly projected using a pseudo-Plate Carrée projection that plots Longitude along the X-axis and Latitude along the Y-axis. The BBOX request parameter (Section 7.2.3.6) values for such a coordinate reference system **shall** be specified in the order minimum longitude, minimum latitude, maximum

© OGC 2001 – All rights reserved

longitude, maximum latitude. The BBOX parameter values **shall** use the coordinate reference system units.

Some Projected Coordinate Reference Systems, e.g., "EPSG:30800" ("RT38 2.5 gon W", used in Sweden), have axes order other than X=East, Y=North. The BBOX request parameter values for such a coordinate system **shall** be specified in the order minimum Easting, minimum Northing, maximum Easting, maximum Northing. The BBOX parameters **shall** use the coordinate reference system units.

The BBOX parameters **shall** be specified using decimal or floating-point notation. In particular, sexagesimal degrees **shall** be specified as decimal degrees.

6.5.5.2 AUTO Namespace for SRS

The AUTO namespace is used for "automatic" projections; that is, for a class of projections that include an arbitrary center of projection. An SRS request parameter specifying an automatic projection includes the AUTO namespace prefix, a numeric projection identifier from the AUTO namespace, a numeric identifier from the EPSG [EPSG] namespace indicating what units are to be used for bounding boxes in that SRS, and values for the central longitude and latitude in decimal degrees:

```
AUTO:auto_proj_id,epsg_units_id,lon0,lat0
```

Valid projection identifiers are defined by this specification in Annex E.

In a request for a georeferenced map or data, the complete AUTO SRS is specified including longitude, latitude, and units. In Capabilities XML, the longitude, latitude, and units variables are omitted because they are chosen by the client in a request for an AUTO SRS.

Example: A service instance indicates that it supports Auto Orthographic projection by including the element "`<SRS>AUTO:42003</SRS>`" in its Capabilities XML. A client may issue a GetMap request for a map in this projection, with bounding box in meters, centered at 100 degrees West longitude and 45 degrees North latitude, using the parameter "`SRS=AUTO:42003,9001,-100,45`".

6.5.5.3 Undefined SRS

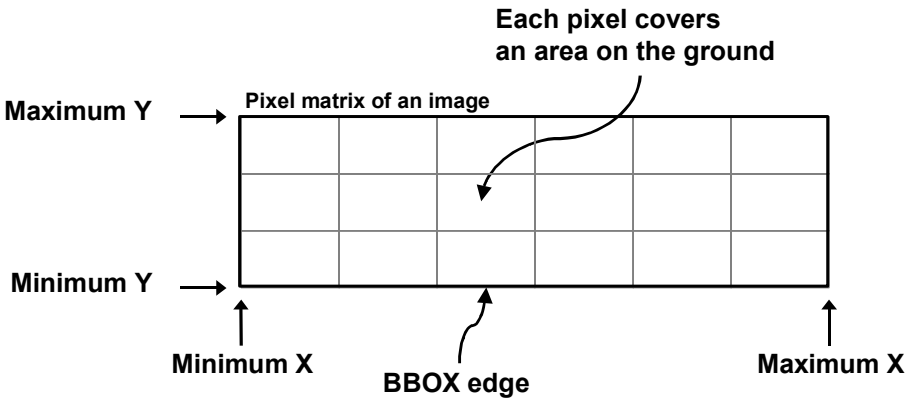
A Server may offer geographic information whose precise spatial reference is undefined. For example, a digitized collection of hand-drawn historical maps may represent an area of the Earth but not employ a modern coordinate system. In such case, the value "NONE" (case-insensitive) **shall** be used when declaring the SRS of such a collection or object. Clients **should not** attempt to overlay information whose SRS=none with other information.

© OGC 2001 – All rights reserved

6.5.6 Bounding Box

The Bounding Box (BBOX) is a set of four comma-separated decimal, scientific notation, or integer values (if integers are provided where floating point is needed, the decimal point is assumed at the end of the number). These values specify the minimum X, minimum Y, maximum X, and maximum Y ranges, in that order, expressed in units of the SRS of the request, such that a rectangular area is defined in those units. When the SRS is a Platte Carrée projection of longitude and latitude coordinates, X refers to the longitudinal axis and Y to the latitudinal axis. The four bounding box values indicate the outside edges of a rectangle, as in Figure 5: minimum X is the left edge, maximum X the right, minimum Y the bottom, and maximum Y the top. The relation of the Bounding Box to the image pixel matrix is shown in the figure: the bounding box goes around the "outside" of the pixels of the image rather than through the centers of the border pixels. In this context, individual pixels have an area.

Figure 5 — Bounding Box representation



A Bounding Box **should not** have zero area.

If a request contains an invalid Bounding Box (e.g., one whose minimum X is greater than or equal to the maximum X, or whose minimum Y is greater than or equal to the maximum Y) the server **shall** throw an exception.

© OGC 2001 – All rights reserved

If a request contains a Bounding Box whose area does not overlap at all with the BoundingBox advertised in the Capabilities XML for the requested geodata object, the server **should** return empty content (e.g., a blank map, empty coverage file, null feature set) for that element. Any elements that are partly or entirely contained in the Bounding Box **should** be returned in the appropriate format.

If the Bounding Box values are not defined for the given SRS (e.g., latitudes greater than 90 degrees in EPSG:4326), the server **should** return empty content for areas outside the valid range of the SRS.

In the particular case of longitude, the following behavior **may** apply regarding the anti-meridian at 180 degrees of longitude. There is a legitimate desire for maps that span the anti-meridian (for example, a map centered on the Pacific Ocean). However, a strict interpretation of the previous paragraph suggests that areas beyond 180 degrees should be shown as empty content; this corresponds to the STRICT constraint below. A server **may** choose to relax this behavior by instead applying the LOOSE constraint below. If `minx` is the west-most longitude in degrees and `maxx` is the east-most, then:

STRICT longitude constraint (default):

```
-180 <= minx < maxx <= 180
```

LOOSE longitude constraint (optional):

```
-180 <= minx < maxx < minx + 360 < 540
```

EXAMPLES: minx,maxx values and the corresponding scope of the bounding box:
-180,180 = Earth centered at Greenwich
0,360 = Earth with Greenwich at left edge
120,250 = Pacific Ocean

6.5.7 Time Dimension

Some geospatial information may be available at multiple times (for example, an hourly weather map). An OGC Web Service may announce available times in its Capabilities XML, and some operations include a parameter for requesting a particular time. The format of a time string is specified in Annex C. Depending on the context, time values may appear as a single value, a list of values, or an interval, as specified in Annex D. When providing temporal information, Servers **should** declare a default value in Capabilities XML unless there is compelling reason to behave otherwise, and Servers **shall** respond with the default value if one has been declared and the Client request does not include a value.

© OGC 2001 – All rights reserved

6.5.8 Elevation Dimension

Some geospatial information may be available at multiple elevations (for example, ozone concentrations at different heights in the atmosphere). An OWS may announce available elevations in its Capabilities XML, and some operations include a parameter for requesting a particular elevation. A single elevation value is an integer or real number whose units are declared by naming an EPSG datum. Depending on the context, elevation values may appear as a single value, a list of values, or an interval, as specified in Annex D. When providing elevation information, Servers **should** declare a default value in Capabilities XML unless there is compelling reason to behave otherwise, and Servers **shall** respond with the default value if one has been declared and the Client request does not include a value.

6.5.9 Other Sample Dimensions

Some geospatial information may be available at other dimensions (for example, satellite images in different wavelength bands). The dimensions other than the four space-time dimensions are referred to as "sample dimensions". An OWS may announce available sample dimensions in its Capabilities XML, and some operations include a mechanism for including dimensional parameters. Each sample dimension has a Name and one or more valid values. The declaration and use of sample dimensions are specified in Annex D.

6.5.10 Additional Request Parameters

Most service requests require additional parameters (beyond REQUEST) to unambiguously state what result to construct. Each OGC Web Service specification defines the required and optional parameters for its operation(s).

6.5.11 Vendor-Specific Parameters

Finally, the requests allow for optional vendor-specific parameters (VSPs) that will enhance the results of a request. Typically, these are used for private testing of non-standard functionality prior to possible standardization. A generic client is **not** required or expected to make use of these VSPs.

An OGC Web Service **shall** produce a valid result even if VSPs are missing or malformed (i.e., the Service **shall** supply a default value), or if VSPs are supplied that are not known to the Service (i.e., the Service **shall** ignore unknown request parameters).

An OGC Web Service **may** choose not to advertise some or all of its VSPs. If VSPs are included in the Capabilities XML, then they **shall** be defined within an internal DTD section of that XML document. (An internal DTD comprises declarations enclosed in square brackets [] within the <DOCTYPE> element of the XML [see ref. 9].) In the absence of such parameters, the internal DTD is absent.

Clients **may** read the internal DTD and formulate requests using any VSPs advertised therein.

© OGC 2001 – All rights reserved

Vendors **should** choose vendor-specific parameter names with care to avoid clashes with standard parameters.

6.6 Service Result

The return value of a valid Service request **shall** correspond to the type requested in the FORMAT parameter. In an HTTP environment, the Content-type header of the response **shall** be exactly the MIME type given in the request.

Several OGC-specific MIME types have been defined in Table 3 for various XML document types (all of which would traditionally be labeled "text/xml"). To be compliant with this Specification a server **shall** return the appropriate OGC MIME type if defined, and the client **shall** be able to accept it, but it is **recommended** that the client also be prepared to accept the MIME type "text/xml" and deduce the specific content type through other means.

6.7 Service Exceptions

Upon receiving a request that is invalid according to the rules of the Distributed Computing Platform (DCP) in use, the service **may** issue an exception of a type valid in that DCP. For example: in the HTTP DCP, if the URL prefix is incorrect an HTTP 404 status code [IETF RFC 2616] is sent.

Upon receiving a request that is invalid according to the relevant OGC Web Services specification, the service **shall** issue a Service Exception Report as defined here and in Annex A.3. The Report is meant to describe to the client application or its human user the reason(s) that the request is invalid.

The EXCEPTIONS parameter in a request indicates the format in which the Client wishes to be notified of Service Exceptions. The only value of the EXCEPTIONS parameter that is defined for all OGC Web Services is "`application/vnd.ogc.se_xml`", which means "Service Exception XML." Particular services may define other formats; this specification defines additional Exception formats for Web Map Servers in Section 7.2.3.11.

NOTE: A client should also be prepared for other returned values and types since there is a possibility that the Service instance is poorly behaved or that a request was directed at a non-compliant OGC Web Service.

Service Exception Report XML **shall** be valid according to the Service Exception DTD in Annex A.3. In an HTTP environment, the MIME type of the returned XML **shall** be "`application/vnd.ogc.se_xml`". Individual error messages appear as <ServiceException> elements within the <ServiceExceptionReport>. The messages can be formatted either as chunks of plain text or, if included in a character data (CDATA) section, as XML-like text containing angle brackets ("<" and ">"), as shown in the example Service Exception Report in Annex A.4.

Service Exceptions **may** include exception codes as indicated in Annex A.3. Services **shall not** use these codes for meanings other than those specified. This specification

© OGC 2001 – All rights reserved

defines several exception codes; The specific codes and semantics of allowed exceptions may be extended by other OGC Web Service implementation specifications. Clients **may** use these codes to automate responses to Service Exceptions.

7 Web Map Service Operations

The three operations defined for a Web Map Service are GetCapabilities, GetMap, and GetFeatureInfo. This section specifies the implementation and use of these WMS operations in the Hypertext Transfer Protocol (HTTP) Distributed Computing Platform (DCP). Future versions may apply to other DCPs.

NOTE: As discussed in the Introduction, an SLD-enabled WMS also offers additional operations, which are specified in reference [3].

7.1 GetCapabilities (required)

7.1.1 General

The purpose of the GetCapabilities operation is described in the Basic Service Elements section, above. In the particular case of a Web Map Service, the response of a GetCapabilities request is general information about the service itself and specific information about the available maps.

7.1.2 GetCapabilities Request Overview

The general form of a GetCapabilities request is defined in the Basic Service Elements section. When making this request of a WMS, which may offer other OGC Web Services as well, it is necessary to indicate that the client seeks information about the WMS in particular. Thus, the SERVICE parameter of the request **shall** have the value "WMS" as shown in Table 4 below.

Table 4 — The parameters of a GetCapabilities request URL

Request Parameter	Required/ Optional	Description
VERSION=version	O	Request version
SERVICE=WMS	R	Service type
REQUEST=GetCapabilities	R	Request name
UPDATESEQUENCE=string	O	Sequence number or string for cache control

© OGC 2001 – All rights reserved

7.1.3 Request Parameters

7.1.3.1 VERSION

The **optional** VERSION parameter, and its use in version negotiation, is specified in the Basic Service Elements section.

In WMS version 1.0.0, the name of this parameter was "WMTVER". That name is now deprecated, but for backwards compatibility and version negotiation a post-1.0.0 server **shall** accept either form without issuing a Service Exception. In the case that VERSION and WMTVER are both given, VERSION takes precedence.

7.1.3.2 SERVICE

The **required** SERVICE parameter indicates which of the available service types at a particular service instance is being invoked. This parameter allows the same URL prefix to offer Capabilities XML for multiple OGC Web Services.

When invoking GetCapabilities on a WMS that implements this version of the specification or a later one, the service_name value "WMS" **shall** be used.

When invoking GetCapabilities on a WMS that implements version 1.0.6 or earlier, the SERVICE parameter **should not** be used by Clients and **may** be ignored by Servers. An earlier server **shall not** issue an Exception upon encountering this parameter (or any other unknown parameter), as specified in Section 6.2.5.1.4 of WMS 1.0.0 [6] and Section 6.4.1 of the present specification.

7.1.3.3 REQUEST

This nature of the **required** REQUEST parameter is specified in the Basic Service Elements section. To invoke the GetCapabilities operation, the value "GetCapabilities" **shall** be used. In WMS version 1.0.0, the value of this parameter was "capabilities". That value is now deprecated, but for backwards compatibility a post-1.0.0 server **shall** accept either form without issuing a Service Exception. When a client is initially contacting a WMS whose version it does not know the Client **should** be prepared to recover if REQUEST=GetCapabilities fails and **may** send REQUEST=capabilities.

7.1.3.4 UPDATESEQUENCE

The **optional** UPDATESEQUENCE parameter is for maintaining cache consistency. Its value can be an integer, a timestamp in [ISO 8601:1988(E)] format (see Appendix B), or any other number or string. The server **may** include an UpdateSequence value in its Capabilities XML. If present, this value **should** be increased when changes are made to the Capabilities (e.g., when new maps are added to the service). The server is the sole judge of lexical ordering sequence. The client **may** include this parameter in its GetCapabilities request. The response of the server based on the presence and relative value of UpdateSequence in the client request and the server metadata **shall** be according to Table 5:

© OGC 2001 – All rights reserved

Table 5 — Use of UpdateSequence Parameter

Client Request UpdateSequence Value	Server Metadata UpdateSequence Value	Server Response
none	any	most recent Capabilities XML
any	none	most recent Capabilities XML
equal	equal	Exception: code=CurrentUpdateSequence
lower	higher	most recent Capabilities XML
higher	lower	Exception: code=InvalidUpdateSequence

7.1.4 GetCapabilities Response

The Basic Service Elements section specifies general rules about the GetCapabilities response.

In the particular case of a Web Map Service complying with this version of the standard, the Extensible Markup Language (XML) [XML 1.0] response **shall** be valid according to the XML Document Type Definition (DTD) in Annex A.1 of this document. The DTD specifies the required and optional content of the response and how the content is formatted.

A server's Capabilities XML **may** reference an exact copy of the DTD in Annex A.1 instead of the master copy at the URL stated in the Annex. The DTD copy **shall** be located at a fully-qualified and accessible URL to permit XML validating software to retrieve it.

A server **may** comply with other published or experimental versions, in which case it **shall** support Version Negotiation as described in the Basic Service Elements section. A DTD for version 1.0.0 was published as an annex to that version of the WMS specification. Other DTDs are archived at <http://www.digitalearth.gov/wmt/xml/>.

7.1.4.1 Names vs. Titles

A number of elements have both a <Name> and a <Title>. Typically, the Name is a single word used for machine-to-machine communication while the Title is for the benefit of humans. For example, a dataset might have the Title "Maximum Atmospheric Temperature" and be requested using the Name "ATMAX".

7.1.4.2 General Service Metadata

The first part of the Capabilities XML is a <Service> element providing general metadata for the service as a whole. It **shall** include a Name, Title, and Online Resource URL. Optionally, Abstract, Keyword List, Contact Information, Fees, and Access Constraints **may** be provided. The meaning of most of these elements is defined in [ISO 19115].

The Service Name **shall** be "OGC:WMS" in the case of a Web Map Service.

© OGC 2001 – All rights reserved

The Service Title is at the discretion of the provider, and **should** be brief yet descriptive enough to identify this server in a menu with other servers.

The Abstract element allows a descriptive narrative providing more information about the enclosing object.

The OnlineResource element within the Service element can be used, for example, to point to the web site of the service provider. There are other OnlineResource elements used for the URL prefix of each supported operation.

A list of keywords or keyword phrases **should** be included to help catalog searching. Currently, no controlled vocabulary has been defined.

Contact Information **should** be included.

The reserved word "none" (case-insensitive) **shall** be used if there are no fees or access constraints, as follows: <Fees>none</Fees>, <AccessConstraints>none</AccessConstraints>. When constraints are imposed, no precise syntax has been defined for the place-holder elements.

7.1.4.3 Capability Metadata

The <Capability> element of the Capabilities XML names the actual operations that are supported by the service instance, the output formats offered for those operations, and the URL prefix for each operation. The XML DTD includes placeholders for Distributed Computing Platforms other than HTTP, and request methods other that HTTP GET, but currently only HTTP GET is defined for a basic WMS.

Ignorable vendor-specific elements **may** be included as discussed in Section 6.5.11. An SLD WMS would also include a <UserDefinedSymbolization> element and URLs for HTTP POST requests.

7.1.4.4 Layers and Styles

The most critical part of the WMS Capabilities XML is the Layers and Styles it defines.

Each available map is advertised by a <Layer> element in the Capabilities XML. A single parent Layer encloses any number of additional layers, which may be hierarchically nested as desired. Some properties defined in a parent layer are inherited by the children it encloses. These inherited properties may be either redefined or added to by the child. Section 7.1.4.7 defines whether or how each property is inherited.

A Map Server **shall** include at least one <Layer> element for each map layer offered. If desired, layers **may** be repeated in different categories when relevant.

No controlled vocabulary has been defined, so at present Layer and Style Names, Titles and Keywords are arbitrary.

© OGC 2001 – All rights reserved

7.1.4.5 Layer Properties

The <Layer> element can enclose child elements providing metadata about the Layer. The values of some of these elements are inherited as defined in Section 7.1.4.7. Their meanings are defined in this Section.

7.1.4.5.1 Title

A <Title> is **required** for all layers; it is a human-readable string for presentation in a menu. The Title is not inherited by child Layers.

7.1.4.5.2 Name

If, and only if, a layer has a <Name>, then it is a map layer that can be requested by using that Name in the LAYERS parameter of a GetMap request. If the layer has a Title but no Name, then that layer is only a category title for all the layers nested within. A Map Server that advertises a Layer containing a Name element **shall** be able to accept that Name as the value of LAYERS argument in a GetMap request and return the corresponding map. A Client **shall not** attempt to request a layer that has a Title but no Name.

A server **shall** throw an exception (code="LayerNotDefined") if an invalid layer is requested.

A containing category itself **may** include a Name by which all of the nested layers can be requested at once. For example, a parent layer "Roads" may have children "Interstates" and "State Highways" and allow the user to request either child individually or both together.

The Name is not inherited by child Layers.

7.1.4.5.3 Abstract and KeywordList

The **optional** <Abstract> and <KeywordList> elements are **recommended**. Abstract is a narrative description of the map layer. KeywordList contains zero or more Keywords to aid in catalog searches. The Abstract and KeywordList elements are not inherited by child Layers.

7.1.4.5.4 Style

Zero or more Styles **may** be advertised for a Layer or collection of layers using <Style> elements, each of which **shall** have <Name> and <Title> elements. The style's Name is used in the Map request STYLES parameter. The Title is a human-readable string. If only a single style is available, that style is known as the "default" style and need not be advertised by the server.

A Style **may** contain several other elements in the Capabilities XML DTD in Annex A.1. In particular, <Abstract> provides a narrative description while <LegendURL> contains

© OGC 2001 – All rights reserved

the location of an image of a map legend appropriate to the enclosing Style. A <Format> element in LegendURL indicates the MIME type of the logo image, and the attributes `width` and `height` state the size of the image in pixels.

Style declarations are inherited by child Layers. A child **shall not** redefine a Style with the same Name as one inherited from a parent. A child **may** define a new Style with a new Name that is not available for the parent Layer.

7.1.4.5.5 SRS

Every Layer is available in one or more spatial reference systems (or in an undefined SRS; see 6.5.5.3).

Every Layer **shall** have at least one <SRS> element that is either stated explicitly or inherited from a parent Layer (Section 7.1.4.6). The root <Layer> element **shall** include a sequence of zero or more SRS elements listing all SRSes that are common to all subsidiary layers. Use a single SRS element with empty content (like so: "`<SRS></SRS>`") if there is no common SRS. Layers may optionally add to the global SRS list, or to the list inherited from a parent layer. Any duplication **shall** be ignored by clients.

When a Layer is available in several Spatial Reference Systems, there are two ways to encode the list of SRS values. The first of these is new in this version of the specification, the second is deprecated but still included for backwards compatibility.

1. **Optional, recommended:** Multiple single-valued <SRS> elements: a list of SRS values is represented as a sequence of <SRS> elements, each of which contains only a single SRS name. Example: <SRS>EPSG:1234</SRS> <SRS>EPSG:5678</SRS>.

2. **Deprecated:** Single list-valued <SRS> element: a list of SRS values is represented as a whitespace-separated list of SRS names contained within a single <SRS> element. Example: <SRS>EPSG:1234 EPSG:5678</SRS>.

WMS 1.1.1 Clients **shall** be prepared to handle either encoding.

NOTE: Change from version 1.1.0: Only the second, deprecated encoding was previously defined.

7.1.4.5.6 LatLonBoundingBox

Every Layer **shall** have exactly one <LatLonBoundingBox> element that is either stated explicitly or inherited from a parent Layer. LatLonBoundingBox states the minimum bounding rectangle of the map data in the EPSG:4326 geographic coordinate system (see Section 6.5.5.1). The LatLonBoundingBox attributes `minx`, `miny`, `maxx`, `maxy` indicate the edges of an enclosing rectangle in decimal degrees as in Figure 5. LatLonBoundingBox **shall** be supplied regardless of what SRS the map server may support, but it **may** be approximate if EPSG:4326 is not supported. Its purpose is to facilitate geographic searches without requiring coordinate transformations by the search engine.

© OGC 2001 – All rights reserved

The LatLonBoundingBox metadata element, and the BoundingBox element defined in the next section, have the following relationship to the BBOX request parameter defined in Section 7.2.3.6. The bounding box metadata in Capabilities XML specify the minimum enclosing rectangle for the layer as a whole. The BBOX request parameter, on the other hand, specifies which rectangular area is to be drawn on the map. The BBOX rectangle may or may not overlap, contain, or be contained within the BoundingBox rectangle.

7.1.4.5.7 BoundingBox

Layers **may** have zero or more <BoundingBox> elements that are either stated explicitly or inherited from a parent Layer. Each BoundingBox states the bounding rectangle of the map data in a particular spatial reference system; the attribute `SRS` indicates which SRS applies. If the data area is shaped irregularly then the BoundingBox gives the minimum enclosing rectangle. The attributes `minx`, `miny`, `maxx`, `maxy` indicate the edges of the bounding box in units of the specified SRS as in Figure 5. **Optional** `resx` and `resy` attributes indicate the spatial resolution of the data in those same units.

NOTE: <LatLonBoundingBox> (Section 7.1.4.5.6) is effectively a BoundingBox in which the attribute SRS="EPSG:4326" is implicit, but LatLonBoundingBox does not include resx and resy attributes. A separate BoundingBox element explicitly naming EPSG:4326 **may** be provided by the server in order, for example, to provide resolution information.

A Layer **may** have multiple BoundingBox element, but each one **shall** state a different SRS. A Layer inherits any BoundingBox values defined by its parents. A BoundingBox inherited from the parent Layer for a particular SRS is replaced by any declaration for the same SRS in the child Layer. A BoundingBox in the child for a new SRS not already declared by the parent is added to the list of bounding boxes for the child Layer. A single Layer element **shall not** contain more than one BoundingBox for the same SRS.

NOTE: There is no provision for describing disjoint bounding boxes. For example, consider a dataset which covers two areas separated by some distance. The server cannot provide two separate bounding boxes in the same Layer using the same SRS to separately describe those areas. To handle this type of situation, the server may either define a single larger bounding box which encloses both areas, or may define two separate Layers that each have distinct Name and BoundingBox values.

A server which has the ability to transform data to different SRSes **may** choose not to provide an explicit BoundingBox for every possible SRS available for each Layer. The server **should** provide BoundingBox information for at least the native SRS of the Layer.

7.1.4.5.8 ScaleHint

Layers **may** include a <ScaleHint> element that suggests minimum and maximum scales for which it is appropriate to display this layer. Because WMS output is destined for output devices of arbitrary size and resolution, the usual definition of scale as the ratio of map size to real-world size is not appropriate here. The following definition of Scale Hint is **recommended**. Consider a hypothetical map with a given Bounding Box, width and height. The central pixel of that map (or the pixel just to the northwest of center) will have some size, which can be expressed as the ground distance in meters of the southwest to northeast diagonal of that pixel. The two values in ScaleHint are the minimum and

© OGC 2001 – All rights reserved

maximum recommended values of that diagonal. It is recognized that this definition is not geodetically precise, but at the same time the hope is that by including it conventions will develop that can be later specified more clearly.

ScaleHint is inherited by child Layers. A ScaleHint declaration in the child replaces the any declaration inherited from the parent.

7.1.4.5.9 Dimension and Extent

The **optional** <Dimension> and <Extent> elements enclose metadata for multi-dimensional data. See Annex C.

Dimension declarations are inherited from parent Layers. Any new Dimension declarations in the child are added to the list inherited from the parent. A child **shall not** redefine a Dimension with the same `name` attribute as one that was inherited.

Extent declarations are inherited from parent Layers. Any Extent declarations in the child with the same `name` attribute as one inherited from the parent replaces the value declared by the parent. A Layer **shall not** declare an Extent unless a Dimension with the same `name` has been declared or inherited earlier in the Capabilities XML.

7.1.4.5.10 MetadataURL

A Map Server **should** use one or more <MetadataURL> elements to offer detailed, standardized metadata about the data underneath a particular layer. The type attribute indicates the standard to which the metadata complies. Two types are defined at present: the value 'TC211' refers to [ISO 19115]; the value 'FGDC' refers to [FGDC-STD-001-1988]. The MetadataURL element **shall not** be used to reference metadata in a non-standardized metadata format; see DataURL (Section 7.1.4.5.14) instead. The enclosed <Format> element indicates the file format MIME type of the metadata record.

MetadataURL elements are not inherited by child Layers.

7.1.4.5.11 Attribution

The **optional** <Attribution> element provides a way to identify the source of the map data used in a Layer or collection of Layers. Attribution encloses several optional elements: <OnlineResource> states the data provider's URL; <Title> is a human-readable string naming the data provider; <LogoURL> is the URL of a logo image. Client applications **may** choose to display one or more of these items. A <Format> element in LogoURL indicates the MIME type of the logo image, and the attributes `width` and `height` state the size of the image in pixels.

The Attribution element is inherited by child layers. Any redefinition by a child replaces the inherited value.

© OGC 2001 – All rights reserved

7.1.4.5.12 Identifier and AuthorityURL

A Map Server **may** use zero or more <Identifier> elements to list ID numbers or labels defined by a particular Authority. The text content of the Identifier element is the ID value. The `authority` attribute of the Identifier element corresponds to the `name` attribute of a separate <AuthorityURL> element. AuthorityURL encloses an <OnlineResource> element which states the URL of a document defining the meaning of the Identifier values.

NOTE: The semantics of how an authority defines the meaning of an identifier have not yet been precisely defined. The Identifier and AuthorityURL elements are provided as a convenience for service instances who wish to indicate the correspondence between the WMS Layers they offer and a classification of those Layers defined by the organization that operates the service.

EXAMPLE: The Global Change Master Directory (gcmd.gsfc.nasa.gov) defines a DIF_ID label for every dataset that it has cataloged. A WMS that renders one of these datasets may associate a Layer with its DIF_ID in the following manner:

```
<AuthorityURL name="gcmd"><OnlineResource xlink:href="some_url" ... /></AuthorityURL>
<Identifier authority="gcmd">id_value</Identifier>
```

AuthorityURL is inherited by subsidiary layers. A child Layer **shall not** define an AuthorityURL with the same `name` attribute as one inherited from a parent. The Identifier element is not inherited. A Layer **shall not** declare an Identifier unless a corresponding AuthorityURL has been declared or inherited earlier in the Capabilities XML.

7.1.4.5.13 FeatureListURL

A Map Server **may** use a <FeatureListURL> element to point to a list of the features represented in a Layer. FeatureListURL is not inherited by child layers.

7.1.4.5.14 DataURL

A Map Server **may** use DataURL to offer more information about the data represented by a particular layer. While the semantics are not well-defined, as long as the results of an HTTP GET request against the DataURL are properly MIME-typed, Viewer Clients and Cascading Map Servers can make use of this. Use <MetadataURL> (Section 7.1.4.5.10) instead for a precisely defined reference to standardized metadata records.

DataURL is not inherited by child layers.

7.1.4.6 Layer Attributes

A <Layer> **may** have zero or more of the following XML attributes: queryable, cascaded, opaque, noSubsets, fixedWidth, fixedHeight. All of these attributes are **optional** and default to 0, in keeping with WMS 1.0.0. Each of these attributes can be inherited or replaced by subsidiary layers. The meaning of each attribute is summarized in Table 6 and detailed in the following subsections.

© OGC 2001 – All rights reserved

Table 6 — Layer Attributes

Attribute	Allowed Values	**Meaning** (0 is default value)
queryable	0, 1	0: not queryable. 1: queryable.
cascaded	0, positive integer	0: layer has not been retransmitted by a Cascading Map Server. n: layer has been retransmitted n times.
opaque	0, 1	0: map data represents vector features that probably do not completely fill space. 1: map data are mostly or completely opaque.
noSubsets	0, 1	0: WMS can map a subset of the full bounding box. 1: WMS can only map the entire bounding box.
fixedWidth	0, positive integer	0: WMS can resize map to arbitrary width. nonzero: map has a fixed width that cannot be changed by the WMS.
fixedHeight	0, positive integer	0: WMS can resize map to arbitrary height. nonzero: map has a fixed height that cannot be changed by the WMS.

7.1.4.6.1 Queryable layers

A Layer is said to be "queryable" if the server supports the GetFeatureInfo operation on that Layer. A server **may** support GetFeatureInfo on some of its layers but not on all. A server **shall** issue a Service Exception (code="LayerNotQueryable") if GetFeatureInfo is requested on a Layer that is not queryable.

7.1.4.6.2 Cascaded layers

A Layer is said to have been "cascaded" if it was obtained from an originating server and then included in the Capabilities XML of a different server. The second server may simply offer an additional access point for the Layer, or may add value by offering additional output formats or spatial reference systems.

If a WMS cascades the content of another WMS then it **shall** increment by 1 the value of the **cascaded** attribute for the affected layers. If that attribute is missing from the originating WMS's Capabilities XML, then the Cascading WMS **shall** insert the attribute and set it to 1.

7.1.4.6.3 Opaque vs. transparent layers

If the optional **opaque** attribute is missing or has a value of "0," then maps made from that Layer will generally have significant "no-data" areas that a client may display as transparent. Vector features such as points and lines are considered not to be opaque in this context (even though at some scales and symbol sizes a collection of features might fill the map area). A value of "1" indicates that the Layer represents an area-filling coverage of space. For example, a map that represents topography and bathymetry as

© OGC 2001 – All rights reserved

regions of differing colors will have no transparent areas. The "opaque" declaration **should** be taken as a hint to the Client to place such a Layer at the bottom of a stack of maps.

This attribute describes only the Layer's data content, not the picture format of the map response. Whether or not a Layer is listed as opaque, a server **shall** still comply with the GetMap operation specified below: that is, the server **shall** send an image with a transparent background if and only if the client requests TRANSPARENT=true and a picture FORMAT that supports transparency.

7.1.4.6.4 Subsettable and resizable layers

The Layer metadata may also include three optional attributes that indicate a map server that is less functional than a normal WMS because it is not able to extract a subset of a larger dataset and/or because it only serves maps of a fixed size and cannot resize them. For example, a WMS that houses a collection of digitized images of historical maps, or pre-computed browse images of satellite data, may not be able to subset or resize those images. However, it can still respond to GetMap requests for complete maps in the original size.

Static image collections may not have a well-defined coordinate system, in which case the server **shall** declare SRS=NONE as described in the Basic Service Elements section.

When set to a value of 1, **noSubsets** indicates that the Server is not able to crop the data or map to a geographic area smaller than its enclosing bounding box.

When present and nonzero, **fixedWidth** and **fixedHeight** indicate that the Server is not able to resize, resample or interpolate the data to a different pixel resolution than its "native" resolution.

7.1.4.7 Inheritance of Layer Properties

Table 7 summarizes how the properties of an enclosing parent layer are inherited by child layers. Properties may be not inherited at all, or inherited as-is, or replaced if the child redefines them, or inherited and added to if the child also defines them. In the latter case, any duplicated definition by the child is ignorable. In Table 7, the **number** column states the number of times each element may appear in a Layer. This column repeats information that is enforced by the DTD in Annex A.1. The meanings of the values in this column are as follows: 1: appears exactly once in each Layer; 0/1: appears either once or not at all; 0+: appears zero or more times; 1+: appears one or more times. The **Inheritance** column indicates whether or how the element is inherited by child Layers. The meanings of the values in this column are as follows: no: not inherited. add: child inherits any value(s) supplied by parent and adds any values of its own ("add" is only relevant for elements that may appear more than one time); replace: value can be inherited from parent and omitted by child, but if specified by child then the parent value is ignored.

© OGC 2001 – All rights reserved

Table 7 — Inheritance of Layer Properties

Element	Number	Inheritance
Layer	0+	no
Name	1	no
Title	1	no
Abstract	0/1	no
KeywordList	0/1	no
Style	0+	add
SRS	1+	add
LatLonBoundingBox	1	replace
BoundingBox	0+	replace
Dimension	0+	add
Extent	0+	replace
Attribution	0/1	replace
AuthorityURL	0+	add
Identifier	0+	no
MetadataURL	0+	no
DataURL	0/1	no
FeatureListURL	0/1	no
ScaleHint	0/1	replace
attributes listed in Table 6	0/1	replace

7.1.5 Output Formats

Format specifiers appear in several places in Capabilities XML: as valid output formats for an operation, as supported Exception formats, and as the format of content at external URLs. Formats are specified as MIME types such as "image/gif" or "image/png". Several OGC-specific types have been defined to distinguish various types of XML documents; these are listed in Table 3, above.

7.2 GetMap (required)

7.2.1 General

The GetMap operation is designed to produce a map, which is defined to be either a pictorial image or a set of graphical elements.

Upon receiving a Map request, a Map Server **shall** either satisfy the request or throw an exception in the format requested.

© OGC 2001 – All rights reserved

7.2.2 GetMap Request Overview

Table 8 describes the Map Request. The request is typically encoded as a URL that is invoked on the WMS using the HTTP GET operation. (The optional Styled Layer Descriptor [3] extensions describe an HTTP POST encoding for communicating with an SLD-enabled WMS.)

Table 8 — The Parameters of a GetMap Request

Request Parameter	Required/ Optional	Description
VERSION=version	R	Request version.
REQUEST=GetMap	R	Request name.
LAYERS=layer_list	R	Comma-separated list of one or more map layers. Optional if SLD parameter is present.
STYLES=style_list	R	Comma-separated list of one rendering style per requested layer. Optional if SLD parameter is present.
SRS=namespace:identifier	R	Spatial Reference System.
BBOX=minx,miny,maxx,maxy	R	Bounding box corners (lower left, upper right) in SRS units.
WIDTH=output_width	R	Width in pixels of map picture.
HEIGHT=output_height	R	Height in pixels of map picture.
FORMAT=output_format	R	Output format of map.
TRANSPARENT=TRUE|FALSE	O	Background transparency of map (default=FALSE).
BGCOLOR=color_value	O	Hexadecimal red-green-blue color value for the background color (default=0xFFFFFF).
EXCEPTIONS=exception_format	O	The format in which exceptions are to be reported by the WMS (default=SE_XML).
TIME=time	O	Time value of layer desired.
ELEVATION=elevation	O	Elevation of layer desired.
Other sample dimension(s)	O	Value of other dimensions as appropriate.
Vendor-specific parameters	O	Optional experimental parameters.
The following parameters are used only with Web Map Services that support the Styled Layer Descriptor specification [3].		
SLD=styled_layer_descriptor_URL	O	URL of Styled Layer Descriptor (as defined in SLD Specification).
WFS=web_feature_service_URL	O	URL of Web Feature Service providing features to be symbolized using SLD.

© OGC 2001 – All rights reserved

7.2.3 Request Parameters

7.2.3.1 VERSION

The **required** VERSION parameter is specified in the Basic Service Elements section.

In WMS version 1.0.0, the name of this parameter was "WMTVER". That name is now deprecated, but for backwards compatibility and version negotiation a post-1.0.0 server **should** accept either form without issuing a Service Exception.

7.2.3.2 REQUEST

The nature of the **required** REQUEST parameter is specified in the Basic Service Elements section. For GetMap, the value "GetMap" **shall** be used.

In WMS version 1.0.0, the value of this parameter was "map". That value is now deprecated, but for backwards compatibility a post-1.0.0 server **should** accept either form without issuing a Service Exception.

7.2.3.3 LAYERS

The **required** LAYERS parameter lists the map layer(s) to be returned by this GetMap request. The value of the LAYERS parameter is a comma-separated list of one or more valid layer names. Allowed layer names are the character data content of any <Layer><Name> element in the Capabilities XML.

A WMS **shall** render the requested layers by drawing the leftmost in the list bottommost, the next one over that, and so on.

7.2.3.4 STYLES

The **required** STYLES parameter lists the style in which each layer is to be rendered. The value of the STYLES parameter is a comma-separated list of one or more valid style names. There is a one-to-one correspondence between the values in the LAYERS parameter and the values in the STYLES parameter. Each map in the list of LAYERS is drawn using the corresponding style in the same position in the list of STYLES. Each style Name **shall** be one that was defined in the <Name> element of a <Style> element that is either directly contained within, or inherited by, the associated <Layer> element in Capabilities XML. (In other words, the Client may not request a Layer in a Style that was only defined for a different Layer.) A server **shall** throw an exception (code=StyleNotDefined) if an unadvertised Style is requested.

A client **may** request the default Style using a null value (as in "STYLES="). If several layers are requested with a mixture of named and default styles, the STYLES parameter includes null values between commas (as in "STYLES=style1,,style2,,"). If all layers are to be shown using the default style, either the form "STYLES=" or "STYLES=,,," is valid.

© OGC 2001 – All rights reserved

7.2.3.5 SRS

The **required** SRS parameter states which Spatial Reference System applies to the values in the BBOX parameter. Spatial Reference Systems are discussed in detail in the Basic Service Elements section. The value of the SRS parameter **shall** be one of the values defined in the character data section of an <SRS> element defined or inherited by the requested layer. The same SRS applies to all layers in a single request.

If the WMS server has declared SRS=NONE for a Layer, as discussed in the Basic Service Elements section, then the Layer does not have a well-defined spatial reference system and should not be shown in conjunction with other layers. The Client **shall** specify SRS=NONE (case-insensitive) in the GetMap request and the Server may issue a Service Exception otherwise.

7.2.3.6 BBOX

The **required** BBOX parameter allows a Client to request a particular Bounding Box. Bounding Boxes are defined in the Basic Service Elements section. The value of the BBOX parameter in a GetMap request is a list of comma-separated numbers of the form "`minx,miny,maxx,maxy`".

If the WMS server has declared that a Layer is not subsettable, as described in Section 7.1.4.6.4, "Subsettable and Resizable Layers," then the Client **shall** specify exactly the declared Bounding Box values in the GetMap request and the Server may issue a Service Exception otherwise.

7.2.3.7 FORMAT

The **required** FORMAT parameter states the desired format of the response to an operation, as specified in the Basic Service Elements section. Supported values for a GetMap request on a WMS instance are listed in one or more <Format> elements in the <Request><GetMap> element of its Capabilities XML. The entire MIME type string in <Format> is used as the value of the FORMAT parameter. In an HTTP environment, the MIME type **shall** be set on the returned object using the Content-type entity header.

For a Web Map Service, allowed formats are either "picture" formats or "graphic element" formats. Picture formats include common image formats such as Graphics Interchange Format (GIF; MIME type "image/gif"), Portable Network Graphics (PNG; MIME type "image/png"), Joint Photographics Expert Group (JPEG; MIME type "image/jpeg"), all of which can be displayed by common Web browsers, and others which may require external helper applications for display. Graphic element formats (less frequently used in WMS practice) include Scalable Vector Graphics (SVG) or Web Computer Graphics Metafile (WebCGM) formats.

Use of a specific format is **not required** by this specification. However, see the discussion about transparency in Section 7.2.3.9 regarding a **recommended** format for practical use.

© OGC 2001 – All rights reserved

7.2.3.8 WIDTH, HEIGHT

The **required** WIDTH and HEIGHT parameters specify the size in integer pixels of the map image to be produced. WIDTH specifies the number of pixels to be used between the minimum and maximum X values (inclusive) in the BBOX parameter, while HEIGHT specifies the number of pixels between the minimum and maximum Y values. The returned picture, regardless of its return format, **shall** have exactly the specified width and height in pixels. In the case where the aspect ratio of the BBOX and the ratio width/height are different, the WMS **shall** stretch the returned map so that the resulting pixels could themselves be rendered in the aspect ratio of the BBOX. In other words, it should be possible using this definition to request a map for a device whose output pixels are themselves non-square, or to stretch a map into an image area of a different aspect ratio.

NOTE: Map distortions will be introduced if the aspect ratio WIDTH/HEIGHT is not commensurate with X, Y and the pixel aspect. Client developers are cautioned to minimize the possibility that users will inadvertently request or unknowingly receive distorted maps.

If a request is for a graphic element return format (e.g., SVG or WebCGM),which does not have explicit width and height, the WIDTH and HEIGHT values **shall** nevertheless be present and **may** be used by the server as helpful information but this specification currently considers this use to be experimental.

If the WMS server has declared that a Layer has fixed width and height, as described in Section 7.1.4.6.4 under "Subsettable and Resizable Layers," then the Client **shall** specify exactly those WIDTH and HEIGHT values in the GetMap request and the Server may issue a Service Exception otherwise.

7.2.3.9 TRANSPARENT

The **optional** TRANSPARENT parameter specifies whether the map background is to be made transparent or not. TRANSPARENT can take on two values, "`TRUE`" or "`FALSE`". The default value is `FALSE` if this parameter is absent from the request.

The ability to return pictures drawn with transparent pixels allows results of different Map requests to be overlaid, producing a composite map. It is **strongly recommended** that every WMS offer a format that provides transparency for layers which could sensibly be overlaid above others.

NOTE: At the time of this writing, the image/gif format provides transparency and is properly displayed by common web clients. The image/png format provides a range of transparency options but support in viewing applications is very poor. The image/jpeg format does not provide transparency at all.

When TRANSPARENT is set to `TRUE` and the `FORMAT` parameter contains a Picture format (e.g., image/gif), then a WMS **shall** return (when permitted by the requested format) a result where all of the pixels not representing features or data values in that Layer are set to a transparent value. For example, a "roads" layer would be transparent wherever no road is shown. When TRANSPARENT is set to `FALSE`, those pixels **shall** be set to the value of BGCOLOR (Section 7.2.3.10).

© OGC 2001 – All rights reserved

When the Layer has been declared as "opaque" as in Section 7.1.4.6.3, then significant portions, or the entirety, of the map may not be able to made transparent.

When the FORMAT parameter contains a Graphic Element format, the TRANSPARENT parameter **may** be included in the request but its value **shall** be ignored by the WMS.

7.2.3.10 BGCOLOR

The **optional** BGCOLOR parameter specifies the color to be used as the background of the map. The general format of BGCOLOR is a hexadecimal encoding of an RGB value where two hexadecimal characters are used for each of Red, Green, and Blue color values. The values can range between 00 and FF for each (0 and 255, base 10). The format is `0xRRGGBB`; either upper or lower case characters are allowed for RR, GG, and BB values. The "0x" prefix **shall** have a lower case 'x'. The default value is `0xFFFFFF` (corresponding to the color white) if this parameter is absent from the request.

When FORMAT is a Picture format, a WMS **shall** render its output on a background whose pixels were initially uniformly of the color encoded in BGCOLOR. When FORMAT is a Graphic Element format (which does not have an explicit background), a WMS should avoid use of the BGCOLOR value for foreground elements because they would not be visible against a background picture of the same color.

When the Layer has been declared as "opaque" as in Section 7.1.4.6.3, then significant portions, or the entirety, of the map may not show any background at all.

7.2.3.11 EXCEPTIONS

The **optional** EXCEPTIONS parameter states the manner in which errors are to be reported to the client. The default value is `application/vnd.ogc.se_xml` if this parameter is absent from the request.

A Web Map Service **shall** offer one or more of the following exception reporting formats by listing them in separate <Format> elements inside the <Exceptions> element of its Capabilities XML. The entire MIME type string in <Format> is used as the value of the EXCEPTIONS parameter. The first of these formats is required to be offered by every WMS; the others are optional.

application/vnd.ogc.se_xml (required)

Errors are reported using Service Exception XML, as specified in Section 6.7 and Annex A.3. This is the default exception format if none is specified in the request. The MIME type of the XML document containing the error message(s) **shall** be application/vnd.ogc.se_xml.

The remaining exception formats are optional. A server **may** issue an exception in the default application/vnd.ogc.se_xml format if a request specifies a different format not supported by the server.

© OGC 2001 – All rights reserved

application/vnd.ogc.se_inimage (optional)

In the case of a Picture format, error messages are graphically returned as part of the content. This would usually take the form of text containing the message being painted into the returned map.

In Graphic Element output formats, the response to this value is experimental and no specific use is required or suggested by this specification.

application/vnd.ogc.se_inimage is a pseudo MIME type; the format and MIME type of the returned content with embedded error messages is actually that given in the FORMAT parameter.

application/vnd.ogc.se_blank (optional)

In the case of a Picture format, if the EXCEPTIONS parameter is set to application/vnd.ogc.se_blank, the WMS **shall**, upon detecting an error, return an object of the type specified in FORMAT whose content is uniformly "off". In the case of an image format such as GIF or JPEG, that would be an object containing only pixels of one color (the background color if BACKGROUND is specified). In the case of a picture format supporting transparency, if TRANSPARENT=TRUE is specified the pixels **shall** all be transparent.

In Graphic Element output formats, such as vector-based formats, this specification suggests that no visible graphic elements be returned.

application/vnd.ogc.se_blank is a pseudo MIME type; the format and MIME type of the returned content with embedded error messages is actually that given in the FORMAT parameter.

7.2.3.12 TIME

The use of Time values is specified in Annexes B and C.

7.2.3.13 ELEVATION

The use of Elevation values is specified in Annex C.

7.2.3.14 Other sample dimensions

The declaration of sample dimensions is specified in Annex C.

7.2.4 Vendor-Specific Parameters

The use of Vendor-Specific Parameters (VSPs) is discussed in the Basic Service Elements section. We repeat here only that clients **may** ignore any VSP they encounter in Capabilities XML, and that servers **shall not** require the presence of VSPs in a request.

© OGC 2001 – All rights reserved

7.2.5 GetMap Response

The response to a valid GetMap request **shall** be a map of the georeferenced information layer requested, in the desired style, and having the specified spatial reference system, bounding box, size, format and transparency.

An invalid GetMap request **shall** yield an error output in the requested Exceptions format (or a network protocol error response in extreme cases).

In an HTTP environment, the MIME type of the returned value's Content-type entity header **shall** match the format of the return value.

7.3 GetFeatureInfo (optional)

7.3.1 General

GetFeatureInfo is an optional operation. It is only supported for those Layers for which the attribute queryable="1" (true) has been defined or inherited (Section 7.1.4.6.1). A Client **shall not** issue a GetFeatureInfo request for other layers. A WMS **should** respond with a properly formatted Service Exception (application/vnd.ogc.se_xml) response if it encounters that request but does not support it.

The GetFeatureInfo operation is designed to provide clients of a WMS with more information about features in the pictures of maps that were returned by previous Map requests. The canonical use case for GetFeatureInfo is that a user sees the response of a Map request and chooses a point on that map for which to obtain more information. The basic operation provides the ability for a client to specify which pixel is being asked about, which layer(s) should be investigated, and what format the information should be returned in. Because the WMS protocol is stateless, the GetFeatureInfo request indicates to the WMS what map the user is viewing by including most of the original GetMap request parameters (all but VERSION and REQUEST). From the spatial context information (BBOX, SRS, WIDTH, HEIGHT) in that GetMap request, along with the X,Y position the user chose, the WMS can (possibly) return additional information about that position.

The behavior described above is geared toward the picture case. In the graphic element case, the semantics of GetFeatureInfo are less defined. The intent is to gain experience with this version of the request and perhaps provide additional rigor in future versions of the specification.

The actual semantics of how a WMS decides what to return more information about, or what exactly to return is left up to the WMS provider.

7.3.2 GetFeatureInfo Request Overview

The parameters of a GetFeatureInfo request are listed in Table 9 below.

© OGC 2001 – All rights reserved

Table 9 — The Parameters of a GetFeatureInfo Request

Request Parameter	Required/ Optional	Description
VERSION=version	R	Request version.
REQUEST=GetFeatureInfo	R	Request name.
<map_request_copy>	R	Partial copy of the Map request parameters that generated the map for which information is desired.
QUERY_LAYERS=layer_list	R	Comma-separated list of one or more layers to be queried.
INFO_FORMAT=output_format	O	Return format of feature information (MIME type).
FEATURE_COUNT=number	O	Number of features about which to return information (default=1).
X=pixel_column	R	X coordinate in pixels of feature (measured from upper left corner=0)
Y=pixel_row	R	Y coordinate in pixels of feature (measured from upper left corner=0)
EXCEPTIONS=exception_format	O	The format in which exceptions are to be reported by the WMS (default=application/vnd.ogc.se_xml).
Vendor-specific parameters	O	Optional experimental parameters.

7.3.3 Request Parameters

7.3.3.1 Prefix

The role of the URL prefix is specified in the Basic Service Elements section. The prefixes for GetCapabilities, GetMap and GetFeatureInfo may be different.

7.3.3.2 VERSION

The **required** VERSION parameter is specified in the Basic Service Elements section.

7.3.3.3 REQUEST

The nature of the **required** REQUEST parameter is specified in the Basic Service Elements section. For GetFeatureInfo, the value "GetFeatureInfo" **shall** be used.

7.3.3.4 map_request_copy

<map request copy> is not a name/value pair like the other parameters. Instead, most of the GetMap request parameters that generated the original map are repeated. Two are omitted because GetFeatureInfo provides its own values: VERSION and REQUEST. The remainder of the GetMap request **shall** be embedded contiguously in the GetFeatureInfo request.

© OGC 2001 – All rights reserved

7.3.3.5 QUERY_LAYERS

The **required** QUERY_LAYERS parameter states the map layer(s) from which feature information is desired to be retrieved. Its value is a comma-separated list of one or more map layers. This parameter **shall** contain at least one layer name, but **may** contain fewer layers than the original GetMap request.

If any layer in this list is not contained in the Capabilities XML of the WMS, the results are undefined and the WMS **shall** produce an exception response.

7.3.3.6 INFO_FORMAT

The **optional** INFO_FORMAT indicates what format to use when returning the feature information. Supported values for a GetFeatureInfo request on a WMS instance are listed as MIME types in one or more <Format> elements inside the <Request><FeatureInfo> element of its Capabilities XML. The entire MIME type string in <Format> is used as the value of the INFO_FORMAT parameter. In an HTTP environment, the MIME type **shall** be set on the returned object using the Content-type entity header.

EXAMPLE: The parameter INFO_FORMAT=application/vnd.ogc.gml requests that the feature information be formatted in Geography Markup Language (GML) [1].

7.3.3.7 FEATURE_COUNT

The **optional** FEATURE_COUNT parameter states the maximum number of features for which feature information should be returned. Its value is a positive integer greater than zero. The default value is 1 if this parameter is omitted.

7.3.3.8 X, Y

The **required** X and Y parameters indicate a point of interest on the map. X and Y identify a single point within the borders of the WIDTH and HEIGHT parameters of the embedded GetMap request. The origin is set to (0,0) centered in the pixel at the upper left corner; X increases to the right and Y increases downward.

7.3.3.9 EXCEPTIONS

The **optional** EXCEPTIONS parameter states the manner in which errors are to be reported to the client. The default value is `application/vnd.ogc.se_xml` if this parameter is absent from the request. At present, not other values are defined for the WMS GetFeatureInfo request.

7.3.4 GetFeatureInfo Response

The WMS **shall** return a response according to the requested INFO_FORMAT if the request is valid, or issue an exception otherwise. The nature of the response is at the discretion of the WMS provider, but it **shall** pertain to the feature(s) nearest to (X,Y).

© OGC 2001 – All rights reserved

7.4 DescribeLayer (SLD WMS only)

The **optional** DescribeLayer operation applies only to a Styled Layer Descriptor WMS. See the SLD specification [3] for this operation.

7.5 GetLegendGraphic (SLD WMS only)

The **optional** GetLegendGraphic operation applies only to a Styled Layer Descriptor WMS. See the SLD specification [3] for this operation.

NOTE: Change from version 1.1.0: This is a new operation.

7.6 GetStyles (SLD WMS only)

The **optional** GetStyles operation applies only to a Styled Layer Descriptor WMS. See the SLD specification [3] for this operation.

NOTE: Change from version 1.1.0: This is a new operation.

7.7 PutStyles (SLD WMS only)

The **optional** PutStyles operation applies only to a Styled Layer Descriptor WMS. See the SLD specification [3] for this operation.

NOTE: Change from version 1.1.0: This is a new operation.

© OGC 2001 – All rights reserved

Annex A
(normative)

XML Document Type Definitions

A.1 WMS Capabilities DTD (Normative)

This annex contains the WMS Capabilities Document Type Definition corresponding to this version of the specification. The DTD may also be found on-line at <http://www.digitalearth.gov/wmt/xml/capabilities_1_1_1.dtd>. Comments in the DTD are informative; in case of conflict with the main body of this specification the main body takes precedence.

```
<!ELEMENT WMT_MS_Capabilities (Service, Capability) >

<!ATTLIST WMT_MS_Capabilities
               version CDATA #FIXED "1.1.1"
               updateSequence CDATA #IMPLIED>

<!-- Elements used in multiple places. -->

<!-- The Name is typically for machine-to-machine communication. -->
<!ELEMENT Name (#PCDATA) >

<!-- The Title is for informative display to a human. -->
<!ELEMENT Title (#PCDATA) >

<!-- The abstract is a longer narrative description of an object. -->
<!ELEMENT Abstract (#PCDATA) >

<!-- An OnlineResource is typically an HTTP URL.  The URL is placed in the
xlink:href attribute.  The xmlns:xlink attribute is a required XML namespace
declaration. -->
<!ELEMENT OnlineResource EMPTY>
<!ATTLIST OnlineResource
          xmlns:xlink CDATA #FIXED "http://www.w3.org/1999/xlink"
          xlink:type CDATA #FIXED "simple"
          xlink:href CDATA #REQUIRED >

<!-- A container for listing an available format's MIME type. -->
<!ELEMENT Format (#PCDATA) >


<!-- General service metadata. -->

<!ELEMENT Service (Name, Title, Abstract?, KeywordList?, OnlineResource,
                   ContactInformation?, Fees?, AccessConstraints?) >

<!-- List of keywords or keyword phrases to help catalog searching. -->
<!ELEMENT KeywordList (Keyword*) >

<!-- A single keyword or phrase. -->
<!ELEMENT Keyword (#PCDATA) >

<!-- Information about a contact person for the service. -->
<!ELEMENT ContactInformation  (ContactPersonPrimary?, ContactPosition?,
                               ContactAddress?, ContactVoiceTelephone?,
                               ContactFacsimileTelephone?,
                               ContactElectronicMailAddress?) >

<!--The primary contact person.-->
```

© OGC 2001 – All rights reserved

```
<!ELEMENT ContactPersonPrimary  (ContactPerson, ContactOrganization) >

<!--The person to contact.-->
<!ELEMENT ContactPerson  (#PCDATA) >

<!--The organization supplying the service.-->
<!ELEMENT ContactOrganization  (#PCDATA) >

<!--The position title for the contact person.-->
<!ELEMENT ContactPosition  (#PCDATA) >

<!--The address for the contact supplying the service.-->
<!ELEMENT ContactAddress  (AddressType,Address,City,StateOrProvince,PostCode,
               Country) >

<!--The type of address.-->
<!ELEMENT AddressType  (#PCDATA) >

<!--The street address.-->
<!ELEMENT Address  (#PCDATA) >

<!--The address city.-->
<!ELEMENT City  (#PCDATA) >

<!--The state or province.-->
<!ELEMENT StateOrProvince  (#PCDATA) >

<!--The zip or postal code.-->
<!ELEMENT PostCode  (#PCDATA) >

<!--The address country.-->
<!ELEMENT Country  (#PCDATA) >

<!--Contact telephone number.-->
<!ELEMENT ContactVoiceTelephone  (#PCDATA) >

<!--The contact fax number.-->
<!ELEMENT ContactFacsimileTelephone  (#PCDATA) >

<!--The e-mail address for the contact.-->
<!ELEMENT ContactElectronicMailAddress  (#PCDATA) >


<!-- Elements indicating what fees or access constraints are imposed. -->
<!ELEMENT Fees (#PCDATA)>
<!ELEMENT AccessConstraints (#PCDATA)>


<!-- A Capability lists available request types, how exceptions
may be reported, and whether any vendor-specific capabilities are defined.  It
also includes an optional list of map layers available from this server. -->
<!ELEMENT Capability
          (Request, Exception, VendorSpecificCapabilities?,
               UserDefinedSymbolization?, Layer?) >

<!-- Available WMS Operations are listed in a Request element. -->
<!ELEMENT Request (GetCapabilities, GetMap, GetFeatureInfo?,
                   DescribeLayer?, GetLegendGraphic?, GetStyles?, PutStyles?) >

<!-- For each operation offered by the server, list the available output
formats and the online resource. -->
<!ELEMENT GetCapabilities (Format+, DCPType+)>
<!ELEMENT GetMap (Format+, DCPType+)>
<!ELEMENT GetFeatureInfo (Format+, DCPType+)>
<!-- The following optional operations only apply to SLD-enabled WMS -->
<!ELEMENT DescribeLayer (Format+, DCPType+)>
<!ELEMENT GetLegendGraphic (Format+, DCPType+)>
<!ELEMENT GetStyles (Format+, DCPType+)>
<!ELEMENT PutStyles (Format+, DCPType+)>

<!-- Available Distributed Computing Platforms (DCPs) are
listed here.  At present, only HTTP is defined. -->
<!ELEMENT DCPType (HTTP) >
```


© OGC 2001 – All rights reserved

```
<!-- Available HTTP request methods.  One or both may be supported. -->
<!ELEMENT HTTP (Get | Post)+ >

<!-- URL prefix for each HTTP request method. -->
<!ELEMENT Get (OnlineResource) >
<!ELEMENT Post (OnlineResource) >

<!-- An Exception element indicates which error-reporting formats are supported. -->
<!ELEMENT Exception (Format+)>

<!-- Optional user-defined symbolization (used only by SLD-enabled WMSes). -->
<!ELEMENT UserDefinedSymbolization EMPTY >
<!ATTLIST UserDefinedSymbolization
          SupportSLD (0 | 1) "0"
          UserLayer (0 | 1) "0"
          UserStyle (0 | 1) "0"
          RemoteWFS (0 | 1) "0" >

<!-- Nested list of zero or more map Layers offered by this server. -->
<!ELEMENT Layer ( Name?, Title, Abstract?, KeywordList?, SRS*,
                  LatLonBoundingBox?, BoundingBox*, Dimension*, Extent*,
                  Attribution?, AuthorityURL*, Identifier*, MetadataURL*, DataURL*,
                  FeatureListURL*, Style*, ScaleHint?, Layer* ) >

<!-- Optional attributes-->
<!ATTLIST Layer queryable (0 | 1) "0"
                cascaded CDATA #IMPLIED
                opaque (0 | 1) "0"
                noSubsets (0 | 1) "0"
                fixedWidth CDATA #IMPLIED
                fixedHeight CDATA #IMPLIED >

<!-- Identifier for a single Spatial Reference Systems (SRS). -->
<!ELEMENT SRS (#PCDATA) >

<!-- The LatLonBoundingBox attributes indicate the edges of the enclosing
rectangle in latitude/longitude decimal degrees (as in SRS EPSG:4326 [WGS1984
lat/lon]). -->
<!ELEMENT LatLonBoundingBox EMPTY>
<!ATTLIST LatLonBoundingBox
          minx CDATA #REQUIRED
          miny CDATA #REQUIRED
          maxx CDATA #REQUIRED
          maxy CDATA #REQUIRED>

<!-- The BoundingBox attributes indicate the edges of the bounding box
in units of the specified spatial reference system. -->
<!ELEMENT BoundingBox EMPTY>
<!ATTLIST BoundingBox
          SRS CDATA #REQUIRED
          minx CDATA #REQUIRED
          miny CDATA #REQUIRED
          maxx CDATA #REQUIRED
          maxy CDATA #REQUIRED
          resx CDATA #IMPLIED
          resy CDATA #IMPLIED>

<!-- The Dimension element declares the _existence_ of a dimension. -->
<!ELEMENT Dimension EMPTY >
<!ATTLIST Dimension
          name CDATA #REQUIRED
          units CDATA #REQUIRED
          unitSymbol CDATA #IMPLIED>

<!-- The Extent element indicates what _values_ along a dimension are valid. -->
<!ELEMENT Extent (#PCDATA) >
<!ATTLIST Extent
          name CDATA #REQUIRED
          default CDATA #IMPLIED
          nearestValue (0 | 1) "0">

<!-- Attribution indicates the provider of a Layer or collection of Layers.
The provider's URL, descriptive title string, and/or logo image URL may be
supplied.  Client applications may choose to display one or more of these
```

© OGC 2001 – All rights reserved

```
items.  A format element indicates the MIME type of the logo image located at
LogoURL.  The logo image's width and height assist client applications in
laying out space to display the logo. -->
<!ELEMENT Attribution ( Title?, OnlineResource?, LogoURL? )>
<!ELEMENT LogoURL (Format, OnlineResource) >
<!ATTLIST LogoURL
          width NMTOKEN #REQUIRED
          height NMTOKEN #REQUIRED>

<!-- A Map Server may use zero or more MetadataURL elements to offer detailed,
standardized metadata about the data underneath a particular layer. The type
attribute indicates the standard to which the metadata complies.  Two types
are defined at present: 'TC211' = ISO TC211 19115; 'FGDC' = FGDC CSDGM.  The
format element indicates how the metadata is structured. -->
<!ELEMENT MetadataURL (Format, OnlineResource) >
<!ATTLIST MetadataURL
          type ( TC211 | FGDC ) #REQUIRED>

<!-- A Map Server may use zero or more Identifier elements to list ID numbers
or labels defined by a particular Authority.  For example, the Global Change
Master Directory (gcmd.gsfc.nasa.gov) defines a DIF_ID label for every
dataset.  The authority name and explanatory URL are defined in a separate
AuthorityURL element, which may be defined once and inherited by subsidiary
layers.  Identifiers themselves are not inherited. -->

<!ELEMENT AuthorityURL (OnlineResource) >
<!ATTLIST AuthorityURL
          name NMTOKEN #REQUIRED >
<!ELEMENT Identifier (#PCDATA) >
<!ATTLIST Identifier
          authority CDATA #REQUIRED >

<!-- A Map Server may use DataURL to offer more information about the data
underneath a particular layer. While the semantics are not well-defined, as
long as the results of an HTTP GET request against the DataURL are properly
MIME-typed, Viewer Clients and Cascading Map Servers can make use of this. -->
<!ELEMENT DataURL (Format, OnlineResource) >

<!-- A Map Server may use FeatureListURL to point to a list of the features
represented in a Layer. -->
<!ELEMENT FeatureListURL (Format, OnlineResource) >

<!-- A Style element lists the name by which a style is requested and a
human-readable title for pick lists, optionally (and ideally) provides a
human-readable description, and optionally gives a style URL. -->
<!ELEMENT Style ( Name, Title, Abstract?,
                  LegendURL*, StyleSheetURL?, StyleURL? ) >

<!-- A Map Server may use zero or more LegendURL elements to provide an
image(s) of a legend relevant to each Style of a Layer.  The Format element
indicates the MIME type of the legend. Width and height attributes are
provided to assist client applications in laying out space to display the
legend. -->
<!ELEMENT LegendURL (Format, OnlineResource) >
<!ATTLIST LegendURL
          width NMTOKEN #REQUIRED
          height NMTOKEN #REQUIRED>

<!-- StyleSheeetURL provides symbology information foreach Style of a Layer. -->
<!ELEMENT StyleSheetURL (Format, OnlineResource) >

<!-- A Map Server may use StyleURL to offer more information about the data or
symbology underlying a particular Style. While the semantics are not
well-defined, as long as the results of an HTTP GET request against the
StyleURL are properly MIME-typed, Viewer Clients and Cascading Map Servers can
make use of this. A possible use could be to allow a Map Server to provide
legend information. -->
<!ELEMENT StyleURL (Format, OnlineResource) >

<!-- Minimum and maximum scale hints for which it is appropriate to
display this layer. -->
<!ELEMENT ScaleHint EMPTY>
<!ATTLIST ScaleHint
          min CDATA #REQUIRED
```


© OGC 2001 – All rights reserved

```
          max CDATA #REQUIRED>
```

A.2 Sample WMS Capabilities XML (Informative)

As an aid to understanding and a guide for implementation, this Annex contains **example** XML which is valid according to the DTD in Annex A.1. Implementers should consult the main body of the specification document and the DTD to ensure compliance rather than editing this XML without full understanding, as the examples herein apply only to a limited range of cases. This is a copy of <http://www.digitalearth.gov/wmt/xml/capabilities_1_1_1.xml>.

```
<?xml version='1.0' encoding="UTF-8" standalone="no" ?>
<!-- The DTD (Document Type Definition) given here must correspond to the version number
declared in the WMT_MS_Capabilities element below. -->
<!DOCTYPE WMT_MS_Capabilities SYSTEM
 "http://www.digitalearth.gov/wmt/xml/capabilities_1_1_1.dtd"
 [
 <!-- Vendor-specific elements are defined here if needed. -->
 <!-- If not needed, you can omit these [] and everything inside. -->
 <!ELEMENT VendorSpecificCapabilities EMPTY>
 ]>  <!-- end of DOCTYPE declaration -->

<!-- Note: this XML is just an EXAMPLE that attempts to show all
required and optional elements for illustration.  Consult the Web Map
Service 1.1.0 specification and the DTD for guidance on what to actually
include and what to leave out. -->

<!-- The version number listed in the WMT_MS_Capabilities element here must
correspond to the DTD declared above.  See the WMT specification document for
how to respond when a client requests a version number not implemented by the
server. -->
<WMT_MS_Capabilities version="1.1.1" updateSequence="0">
<!-- Service Metadata -->
<Service>
  <!-- The WMT-defined name for this type of service -->
  <Name>OGC:WMS</Name>
  <!-- Human-readable title for pick lists -->
  <Title>Acme Corp. Map Server</Title>
  <!-- Narrative description providing additional information -->
  <Abstract>WMT Map Server maintained by Acme Corporation.  Contact:
webmaster@wmt.acme.com.  High-quality maps showing roadrunner nests and possible ambush
locations.</Abstract>
  <KeywordList>
    <Keyword>bird</Keyword>
    <Keyword>roadrunner</Keyword>
    <Keyword>ambush</Keyword>
  </KeywordList>
  <!-- Top-level web address of service or service provider.  See also OnlineResource
  elements under <DCPType>. -->
  <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple"
   xlink:href="http://hostname/" />
  <!-- Contact information -->
  <ContactInformation>
    <ContactPersonPrimary>
      <ContactPerson>Jeff deLaBeaujardiere</ContactPerson>
      <ContactOrganization>NASA</ContactOrganization>
    </ContactPersonPrimary>
    <ContactPosition>Computer Scientist</ContactPosition>
    <ContactAddress>
      <AddressType>postal</AddressType>
      <Address>NASA Goddard Space Flight Center, Code 933</Address>
      <City>Greenbelt</City>
      <StateOrProvince>MD</StateOrProvince>
      <PostCode>20771</PostCode>
      <Country>USA</Country>
    </ContactAddress>
    <ContactVoiceTelephone>+1 301 286-1569</ContactVoiceTelephone>
```

© OGC 2001 – All rights reserved

```
    <ContactFacsimileTelephone>+1 301 286-1777</ContactFacsimileTelephone>

<ContactElectronicMailAddress>delabeau@iniki.gsfc.nasa.gov</ContactElectronicMailAddress>
  </ContactInformation>
  <!-- Fees or access constraints imposed. -->
  <Fees>none</Fees>
  <AccessConstraints>none</AccessConstraints>
</Service>
<Capability>
  <Request>
    <GetCapabilities>
      <Format>application/vnd.ogc.wms_xml</Format>
      <DCPType>
        <HTTP>
          <Get>
            <!-- The URL here for invoking GetCapabilities using HTTP GET
            is only a prefix to which a query string is appended. -->
            <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
             xlink:type="simple"
             xlink:href="http://hostname:port/path" />
          </Get>
          <Post>
            <!-- The URL here for invoking GetCapabilities using HTTP POST
            includes the complete address to which a query would be sent in
            XML format.  This is here for future expansion; no POST encoding
                 has yet been defined. -->
            <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
             xlink:type="simple"
             xlink:href="http://hostname:port/path" />
          </Post>
        </HTTP>
      </DCPType>
    </GetCapabilities>
    <GetMap>
      <Format>image/gif</Format>
      <Format>image/png</Format>
      <Format>image/jpeg</Format>
      <DCPType>
        <HTTP>
          <Get>
            <!-- The URL here for invoking GetCapabilities using HTTP GET
            is only a prefix to which a query string is appended. -->
            <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
             xlink:type="simple"
             xlink:href="http://hostname:port/path" />
          </Get>
        </HTTP>
      </DCPType>
    </GetMap>
    <GetFeatureInfo>
      <Format>application/vnd.ogc.gml</Format>
      <Format>text/plain</Format>
      <Format>text/html</Format>
      <DCPType>
        <HTTP>
          <Get>
            <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
             xlink:type="simple"
             xlink:href="http://hostname:port/path" />
          </Get>
        </HTTP>
      </DCPType>
    </GetFeatureInfo>
    <DescribeLayer><!--optional; used only by SLD-enabled WMS-->
      <Format>application/vnd.ogc.gml</Format>
      <DCPType>
        <HTTP>
          <Get>
            <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
             xlink:type="simple"
             xlink:href="http://hostname:port/path" />
          </Get>
        </HTTP>
      </DCPType>
```


© OGC 2001 – All rights reserved

```
    </DescribeLayer>
  </Request>
  <Exception>
    <Format>application/vnd.ogc.se_xml</Format>
    <Format>application/vnd.ogc.se_inimage</Format>
    <Format>application/vnd.ogc.se_blank</Format>
  </Exception>
  <!-- Any text or markup is allowed here, as required to describe
       vendor-specific capabilities.  Please define elements and attributes
       in the DOCTYPE declaration at the start of the document. -->
  <!-- This example is empty because no VSPs were defined in preamble -->
  <VendorSpecificCapabilities />
  <!-- Place-holder for Styled Layer Descriptor (SLD)-enabled WMSes.
       This element is absent for a basic WMS. -->
  <UserDefinedSymbolization SupportSLD="1" UserLayer="1" UserStyle="1"
             RemoteWFS="1" />
  <Layer>
    <Title>Acme Corp. Map Server</Title>
    <SRS>EPSG:4326</SRS> <!-- all layers are available in at least this SRS -->
    <AuthorityURL name="DIF_ID">
      <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple"
       xlink:href="http://gcmd.gsfc.nasa.gov/difguide/whatisadif.html" />
    </AuthorityURL>
    <Layer>
      <!-- This parent layer has a Name and can therefore be requested from a Map Server,
yielding a map of all subsidiary layers. -->
      <Name>ROADS_RIVERS</Name>
      <Title>Roads and Rivers</Title>
      <!-- See the spec to learn how some characteristics are inherited by
           subsidiary layers. -->
      <SRS>EPSG:26986</SRS> <!-- An additional SRS for this layer -->
      <LatLonBoundingBox minx="-71.63" miny="41.75" maxx="-70.78" maxy="42.90"/>
      <!-- The optional resx and resy attributes below indicate the X and Y spatial
           resolution in the units of that SRS. -->
      <!-- The EPSG:4326 BoundingBox duplicates some of the info in LatLonBoundingBox
           and is therefore optional, but using it here allows the additional
           resolution attributes. -->
      <BoundingBox SRS="EPSG:4326"
       minx="-71.63" miny="41.75" maxx="-70.78" maxy="42.90" resx="0.01" resy="0.01"/>
      <BoundingBox SRS="EPSG:26986"
       minx="189000" miny="834000" maxx="285000" maxy="962000" resx="1" resy="1" />
      <!-- Optional Title, URL and logo image of data provider. -->
      <Attribution>
        <Title>State College University</Title>
        <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple"
         xlink:href="http://www.university.edu/" />
        <LogoURL width="100" height="100">
          <Format>image/gif</Format>
          <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
           xlink:type="simple"
           xlink:href="http://www.university.edu/icons/logo.gif" />
        </LogoURL>
      </Attribution>
      <!-- Identifier whose meaning is defined in an AuthorityURL element -->
      <Identifier authority="DIF_ID">123456</Identifier>
      <FeatureListURL>
        <Format>application/vnd.ogc.se_xml"</Format>
        <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple"
         xlink:href="http://www.university.edu/data/roads_rivers.gml" />
      </FeatureListURL>
      <Style>
        <Name>USGS</Name>
        <Title>USGS Topo Map Style</Title>
        <Abstract>Features are shown in a style like that used in USGS topographic
maps.</Abstract>
        <!-- A picture of a legend for a Layer in this Style -->
        <LegendURL width="72" height="72">
          <Format>image/gif</Format>
          <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
           xlink:type="simple"
           xlink:href="http://www.university.edu/legends/usgs.gif" />
        </LegendURL>
        <!-- An XSL stylesheet describing how GML feature data will rendered to create
             a map of this layer. -->
```

© OGC 2001 – All rights reserved

```
        <StyleSheetURL>
          <Format>text/xsl</Format>
          <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
           xlink:type="simple"
           xlink:href="http://www.university.edu/stylesheets/usgs.xsl" />
        </StyleSheetURL>
      </Style>
      <ScaleHint min="4000" max="35000"></ScaleHint>
      <Layer queryable="1">
             <Name>ROADS_1M</Name>
             <Title>Roads at 1:1M scale</Title>
             <Abstract>Roads at a scale of 1 to 1 million.</Abstract>
             <KeywordList>
          <Keyword>road</Keyword>
          <Keyword>transportation</Keyword>
          <Keyword>atlas</Keyword>
             </KeywordList>
             <Identifier authority="DIF_ID">123456</Identifier>
        <!-- Metadata specific to this particular layer.  The same FGDC metadata is
offered in two formats. -->
             <MetadataURL type="FGDC">
          <Format>text/plain</Format>
          <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
           xlink:type="simple"
           xlink:href="http://www.university.edu/metadata/roads.txt" />
        </MetadataURL>
             <MetadataURL type="FGDC">
           <Format>text/xml</Format>
           <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
            xlink:type="simple"
            xlink:href="http://www.university.edu/metadata/roads.xml" />
        </MetadataURL>
        <!-- In addition to the Style specified in the parent Layer, this Layer is
available in this style. -->
             <Style>
               <Name>ATLAS</Name>
               <Title>Road atlas style</Title>
               <Abstract>Roads are shown in a style like that used in a commercial road
atlas.</Abstract>
        <LegendURL width="72" height="72">
          <Format>image/gif</Format>
          <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink"
           xlink:type="simple"
           xlink:href="http://www.university.edu/legends/atlas.gif" />
        </LegendURL>
             </Style>
      </Layer>
      <Layer queryable="1">
             <Name>RIVERS_1M</Name>
             <Title>Rivers at 1:1M scale</Title>
             <Abstract>Rivers at a scale of 1 to 1 million.</Abstract>
             <KeywordList>
          <Keyword>river</Keyword>
          <Keyword>canal</Keyword>
          <Keyword>waterway</Keyword>
             </KeywordList>
      </Layer>
    </Layer>
    <Layer queryable="1">
      <Title>Weather Forecast Data</Title>
      <SRS>EPSG:4326</SRS> <!-- harmless repetition of common SRS -->
      <LatLonBoundingBox minx="-180" miny="-90" maxx="180" maxy="90" />
      <!-- These weather data are available daily from 1999-01-01 through
           2000-08-22. -->
      <Dimension name="time" units="ISO8601" />
      <Extent name="time" default="2000-08-22">1999-01-01/2000-08-22/P1D</Extent>
      <Layer>
             <Name>Clouds</Name>
             <Title>Forecast cloud cover</Title>
      </Layer>
      <Layer>
             <Name>Temperature</Name>
             <Title>Forecast temperature</Title>
      </Layer>
```


© OGC 2001 – All rights reserved

```
      <Layer>
             <Name>Pressure</Name>
             <Title>Forecast barometric pressure</Title>
        <!-- Pressure is available at several elevations.
          EPSG:5030 is WGS 84 ellipsoid, units in metres.
          Pressure is also available at several times.
          NOTE: first list all Dimension elements, then all Extent elements. -->
          <Dimension name="time" units="ISO8601" />
          <Dimension name="elevation" units="EPSG:5030" />
          <Extent name="time" default="2000-08-22">1999-01-01/2000-08-22/P1D</Extent>
          <Extent name="elevation" default="0"
nearestValue="1">0,1000,3000,5000,10000</Extent>
      </Layer>
    </Layer>
    <!-- Example of a layer which is a static map of fixed
         size which the server cannot subset or make transparent -->
    <Layer opaque="1" noSubsets="1" fixedWidth="512" fixedHeight="256">
      <Name>ozone_image</Name>
      <Title>Global ozone distribution (1992)</Title>
      <LatLonBoundingBox minx="-180" miny="-90" maxx="180" maxy="90" />
      <Extent name="time" default="1992">1992</Extent>
    </Layer>
    <!-- Example of a layer which originated from another WMS and has been
         "cascaded" by this WMS. -->
    <Layer cascaded="1">
      <Name>population</Name>
      <Title>World population, annual</Title>
      <LatLonBoundingBox minx="-180" miny="-90" maxx="180" maxy="90" />
      <Extent name="time" default="2000">1990/2000/P1Y</Extent>
    </Layer>
  </Layer>
</Capability>
</WMT_MS_Capabilities>
```

A.3 Service Exception DTD (Normative)

This annex contains the Service Exception Document Type Definition corresponding to this version of the specification. The DTD may also be found on-line at <http://www.digitalearth.gov/wmt/xml/exception_1_1_1.dtd>. This section also summarizes the defined exception codes and their meanings.

```
<!ELEMENT ServiceExceptionReport (ServiceException*)>
<!ATTLIST ServiceExceptionReport version CDATA #FIXED "1.1.1">

<!ELEMENT ServiceException (#PCDATA)>
<!ATTLIST ServiceException code CDATA #IMPLIED>
```

Table A.1 — Exception codes defined by this specification

Exception Code	Meaning
InvalidFormat	Request contains a Format not offered by the service instance.
InvalidSRS	Request contains an SRS not offered by the service instance for one or more of the Layers in the request.
LayerNotDefined	Request is for a Layer not offered by the service instance.
StyleNotDefined	Request is for a Layer in a Style not offered by the service instance.
LayerNotQueryable	GetFeatureInfo request is applied to a Layer which is not declared queryable.
CurrentUpdateSequence	Value of (optional) UpdateSequence parameter in GetCapabilities request is equal to current value of Capabilities XML update sequence number.
InvalidUpdateSequence	Value of (optional) UpdateSequence parameter in GetCapabilities request is greater than current value of Capabilities XML update sequence number.

© OGC 2001 – All rights reserved

MissingDimensionValue	Request does not include a sample dimension value, and the service instance did not declare a default value for that dimension.
InvalidDimensionValue	Request contains an invalid sample dimension value.

A.4 Sample Service Exception XML (Informative)

As an aid to understanding and a guide for implementation, this Annex contains **example** XML which is valid according to the DTD in Annex A.3. Implementers should consult the main body of the specification document and the DTD to ensure compliance rather than editing this XML without full understanding. This is a copy of <http://www.digitalearth.gov/wmt/xml/exception_1_1_0.xml>.

```
<?xml version='1.0' encoding="UTF-8" standalone="no" ?>
<!DOCTYPE ServiceExceptionReport SYSTEM
"http://www.digitalearth.gov/wmt/xml/exception_1_1_0.dtd">
<ServiceExceptionReport version="1.1.0">
<ServiceException>
Plain text message about an error.
</ServiceException>
<ServiceException code="InvalidUpdateSequence">
Another error message, this one with an exception code supplied.
</ServiceException>
<ServiceException>
<![CDATA[
Error in module <foo.c>, line 42

A message that includes angle brackets in text
must be enclosed in a Character Data Section
as in this example.  All XML-like markup is
ignored except for this sequence of three
closing characters:
]]>
</ServiceException>
<ServiceException>
<![CDATA[
<Module>foo.c</Module>
<Error>An error occurred</Error>
<Explanation>Similarly, actual XML
can be enclosed in a CDATA section.
A generic parser will ignore that XML,
but application-specific software may choose
to process it.</Explanation>
]]>
</ServiceException>
</ServiceExceptionReport>
```


© OGC 2001 – All rights reserved

Annex B
(normative)

Formatting Dates and Times

B.1 Overview

This annex specifies the encoding of moments and periods in time. These allow OGC Web Services to support temporal data descriptions and requests. This specification is based on [ISO 8601:1988(E)]; it **extends** ISO 8601 in the following ways:

- It defines a syntax for expressing the start, end and periodicity of a data collection.
- It defines terms to represent the 7 days of the week.
- It allows years before 0001 AD.
- It allows times in the distant geologic past (thousands, millions or billions of years before present).

This Annex specifies the formats for a moment in time, including years in the current era ("AD") and years before the current era or on geologic timescales, the format for a period of time, and the format for incomplete data fragments.

NOTE: A revision of the ISO date standard, ISO 8601:2000, allows years more than 4 digits long and a leading minus sign to denote negative years, and allows recurring intervals. The next revision of the WMS specification will adopt ISO 8601:2000 and thereby allow some of the extensions defined in this Annex to be removed.

B.2 Time Format Details

B.2.1 Basic Syntax

The basic time format uses ISO 8601:1988(E) "extended" format: up to 14 digits specifying century, year, month, day, hour, minute, seconds, and seconds, and optionally a decimal point followed by zero or more digits for fractional seconds, with non-numeric characters to separate each piece:

ccyy-mm-ddThh:mm:ss.sssZ

The precision may be reduced by omitting least-significant digits, as in the examples below. ISO 8601:1988(E) prefers a decimal comma before fractional seconds but allows a decimal period as in this document.

© OGC 2001 – All rights reserved

All times **should** be expressed in Coordinated Universal Time (UTC) as indicated by the suffix Z (for "zulu"); this suffix is **required** when UTC applies if the hours field appears in the time string. When a local time applies, a numeric time zone suffix as defined by clause 5.3.3.1 of [ISO 8601:1988(E)] **may** be used. The absence of any suffix at all means local time in an undefined zone, which **must not** be used in the global network of map servers enabled by the WMS specification.

NOTE: Change from version 1.1.0: the use of the suffix Z was previously incorrectly stated to be optional for UTC.

EXAMPLE 1:	ccyy	Year only
EXAMPLE 2:	ccyy-mm	Year and month
EXAMPLE 3:	ccyy-mm-dd	Year, month and day
EXAMPLE 4:	ccyy-mm-ddThhZ	Year, month, day and hour in UTC

B.2.2 Extension for years B.C.E.

An **extension** to [ISO 8601:1988(E)] is used to handle years B.C.E. (before year 0001 AD). Because a leading hyphen has meaning in ISO 8601:1988(E) (it can indicate missing parts of the time), preceding times with a minus sign is not advisable. Instead, the prefix 'B' is used to indicate B.C.E. years.

EXAMPLE: B1350 (1350 "BC", approximate time of Tutankhamen).

B.2.3 Extension for geologic datasets

An **extension** to [ISO 8601:1988(E)] is used to handle geologic datasets referring to the distant past. Geologists refer to years some number of thousand, million or billion (10^9) years ago as ka, Ma or Ga. Therefore, the prefixes K, M and G, followed by an integer or floating-point number of years, are used.

EXAMPLE 1: M150 (Jurassic period, 150 Ma)

EXAMPLE 2: K150000 (Jurassic period, 150 Ma)

EXAMPLE 3: K18 (last Ice Age, 18 ka)

EXAMPLE 4: M0.018 (last Ice Age, 18 ka)

EXAMPLE 5: G5 (Earth's crust solidifies, 5 Ga)

B.3 Period Format

An **extension** to [ISO 8601:1988(E)] is used to indicate the time resolution of the available data. The [ISO 8601:1988(E)] format for representing a period of time is used to represent the resolution: Designator P (for Period), number of years Y, months M, days D, time designator T, number of hours H, minutes M, seconds S. Unneeded elements may be omitted.

© OGC 2001 – All rights reserved

EXAMPLE 1: P1Y -- 1 year

EXAMPLE 2: P1M10D -- 1 month plus 10 days

EXAMPLE 3: PT2H -- 2 hours

EXAMPLE 4: PT1.5S -- 1.5 seconds

B.4 Time Lists and Ranges

As shown in Table C.1, a list of several times is expressed by separating valid time values with a comma (","). A temporal range, on the other hand, is expressed using an **extension** to [ISO 8601:1988(E)]. The syntax start/end/period indicates the start time of the data, the ending time, and the time resolution or refresh rate. Data which are refreshed at regular intervals but not archived are represented using time/time/period, where both time values are the same (that of the latest data) and the period indicates the refresh rate.

B.5 Date Fragments

In some contexts, date information may be incomplete. For example, a GeoParser service that scans text documents for references to locations or times may encounter partial references such as "Tuesday" or "August 30." To allow this time specification to handle such fragments, we include the "truncated representation" allowed by ISO 8601:1988(E), and we add 7 terms to represent days of the week.

B.5.1 Truncated representation

The "truncated representation" allowed by ISO 8601:1988(E) uses hyphens to denote pieces missing from the most significant end of the time string. Add one hyphen for each missing two-digit century, year, month, day, hour or minute. If the time zone is unknown, then the time zone suffix **may** be omitted from the date fragment. See examples in B.6.2, below.

B.5.2 Days of the week

We add 7 terms to represent days of the week: 'MON', 'TUE', 'WED', 'THU', 'FRI', 'SAT', 'SUN'. These optional terms appear at the start of the date/time string, and a comma separates the weekday (without whitespace) from the ISO 8601:1988(E) string. If the date fragment contains only a day of week, then the comma is not necessary. See examples in B.6.2, below.

© OGC 2001 – All rights reserved

B.6 Examples

B.6.1 Complete dates and times

A single moment (scalar value):

2000-06-23T20:07:48.11Z

Quarterly data (comma-separated list):

1999-01-01,1999-04-01,1999-07-01,1999-10-01

Daily data taken at noon since April 15 1995 (periodic interval):

1995-04-22T12:00Z/2000-06-21T12:00Z/P1D

Current data refreshed every 30 min (periodic interval):

2000-06-18T14:30Z/2000-06-18T14:30Z/PT30M

B.6.2 Date and time fragments

--11-21 (Nov. 21, truncated century and year, omitted hh:mm:ss)

TUE,--11-21 (Tuesday, Nov. 21)

TUE (Tuesday)

2000 (year 2000)

-99 (year 99, century not given)

T14 (2pm local time)

MON,---21T14 (Monday the 21st at 2pm local time)

T14Z (2pm UTC)

T14-04:00 (2pm EDT, 4hrs earlier than UTC)

T-10 (ten minutes past an unspecified hour)

© OGC 2001 – All rights reserved

Annex C
(normative)

Handling Multi-Dimensional Data

C.1 Overview

This annex describes support for multi-dimensional data in the service metadata and operation request parameters of OGC Web Services. "Dimensions" in this Annex refers to Time, Elevation, and sample dimensions.

EXAMPLE: Satellite imagery may be available at several different times and several different wavelength bands. The valid times and band numbers can each be separately specified.

C.2 Declaring Dimensions

The optional element <Dimension> is used in Capabilities XML to declare that one or more dimensional parameters are relevant to the information holdings of that server. The Dimension element does not provide valid values for a Dimension; that is the role of the Extent element described below. A Dimension element includes a **required** name, a **required** measurement units specifier, and an **optional** unitSymbol.

The following is a DTD fragment expression for the Dimension element (from the full DTD in Annex A.1):

```
<!Element Dimension EMPTY>
<!ATTLIST Dimension
name          ID  #REQUIRED
units         CDATA   #REQUIRED
unitSymbol    CDATA   #IMPLIED>
```

EXAMPLES: The following are some examples of server-defined dimensions:

```
<Dimension name="wavelength" units="Angstrom" unitSymbol="Ao" />
<Dimension name="temperature" units="Kelvin" unitSymbol="K" />
<Dimension name="pressure_scale_height" units="millibar" unitSymbol="mbar" />
<Dimension name="altitude" units="kilometer" unitSymbol="km" />
```

Dimension elements **shall** comply with the following rules:

– Dimension name values **shall** be case-insensitively unique within a service instance and **should** contain no whitespace.

– Unit names and abbreviations are from the Unified Code for Units of Measure [UCUM]. The **required** units attribute is from the UCUM "name" column. The

© OGC 2001 – All rights reserved

optional (but **recommended**) unitSymbol is from the UCUM "c/s" column (case-sensitive abbreviation). Prefix names and symbols **may** be present as well.

– If the dimensional quantity has no units (e.g., band number in a multi-wavelength sensor), use the null string: units="".

All Dimensions in a Capabilities XML response are server-defined, with two exceptions. The dimensions named time and elevation are privileged special cases, predefined as follows:

```
<Dimension name="time" units="ISO8601" />
<Dimension name="elevation" units="EPSG:vertical_datum" />
```

The units "ISO8601" refers to the Time representation specified in Annex B. The units "EPSG:vertical_datum" refers to elevation units and a reference used by one of the European Petroleum Survey Group [ref. 13] vertical datums (use the actual datum number in place of the string "vertical_datum").

EXAMPLE: Elevation units given as "EPSG:5030" means "meters above the WGS84 ellipsoid."

Although predefined, a server **shall** include Time and/or Elevation declarations if it provides extents for those dimensions.

If a server opts to specify elevation or time values in reference systems other than those listed above, it **shall** use other dimension names instead of "elevation" and "time" (e.g., "altitude" and "date").

Dimension declarations are inherited by enclosed Layers, as specified in Section 7.1.4.6. A dimension **shall not** be re-declared using the same name (case-insensitively) with conflicting information.

C.3 Specifying Dimensional Extents

Having declared the existence of a Dimension, the Capabilities XML response uses corresponding Extent elements to specify the bounds of a geodata object along zero or more independent single dimensions.

Format of Extent element:

```
<Extent name="dimension_name" default="default_value"
  multipleValues="0|1" nearestValue="0|1">
extent_value</Extent>
```


© OGC 2001 – All rights reserved

Extent values:

The extent_value string declares what value(s) along the Dimension axis are appropriate for this specific geospatial data object. The extent_value has the syntax shown in Table C.1.

Table C.1 — Syntax for Listing One or More Dimensional Values

Syntax	Meaning
value	A single value
value1,value2,value3,... [a]	A list of multiple values
min/max/resolution	An interval defined by its lower and upper bounds and its resolution.
min1/max1/res1,min2/max2/res2,... [a]	A list of multiple intervals
[a] Whitespace is allowed following commas in a list in an <Extent> element.	

Extent attributes:

Several attributes are defined for the Extent element. Only name is **required.**

- name (required): The value dimension_name is the same as the value of a name attribute of a <Dimension> element previously defined as described in Section C.2. The Dimension name and Extent name **shall** match (case-insensitively).

- default (optional): The default_value declares what value would be used along that Dimension if a Web request omits a value for that Dimension. See Section C.5

- multipleValues (optional): This attribute, if absent or zero, indicates that requests **shall** include only a single value for this dimension. If present and nonzero, then requests **may** include multiple values. See Section C.4.3.

- nearestValues (optional): This attribute, if absent or zero, indicates that the server will not round off inexact dimension values to the nearest valid value. If present and nonzero, then the server will perform rounding. See Section C.5.

- current (optional; only valid for Time extents): This attribute, if present and nonzero, indicates (1) that temporal data are normally kept current and (2) that the request parameter TIME **may** include the keyword 'current' instead of an ending value (see Section C.4.1).

Extent declarations are inherited by enclosed Layers, as specified in Section 7.1.4.6. A extent that is re-specified in an enclosed layer using the same name (case-insensitively) replaces the extent that had been inherited.

© OGC 2001 – All rights reserved

A resolution value of zero (as in min/max/0) means that the data are effectively at infinitely-fine resolution for the purposes of making requests on the service instance. For instance, an instrument which continuously monitors randomly-occurring data may have no explicitly defined temporal resolution.

EXAMPLE:

Here is a <Layer> element with Extents specified in WMS Capabilities XML ('...' indicates XML omitted for clarity):

```
<Layer>
  <!-- Declare dimensions in use.  Declarations are inherited by enclosed Layers. -->
  <Dimension name="time" units="ISO8601" />
  <Dimension name="temperature" units="Kelvin" unitSymbol="K" />
  <Dimension name="elevation" units="EPSG:5030" />
      …
  <Layer>  …
    <!-- Specify extent of Layer.  Extents are inherited by enclosed Layers. -->
    <Extent name="time" default="2000-10-17">1996-01-01/2000-10-17/P1D</Extent>
    <Extent name="elevation" default="0">0/10000/100</Extent>
    <Extent name="temperature" default="300">230,300,400</Extent>
  </Layer>
  …
</Layer>
```

C.4 Including Dimensional Values in a Request

Requests against multi-dimensional data objects described with <Dimension> and <Extent> require additional parameters to formulate a complete query. This following section specifies how Clients should include dimensional parameters in those operations which support such parameters.

C.4.1 Elevation and Time Values in Requests

If a data object has an Elevation extent defined, then operation requests to retrieve that object may include

`ELEVATION=`*`value`*

If a Layer has a Time extent defined, then requests may include

`TIME=`*`value`*

In either case, *value* uses the format described in Table C.1 to provide a single value, a comma-separated list, or an interval of the form start/end without a resolution. *value* **shall not** contain whitespace. An interval in a request value is a request for all the data from the start value up to and including the end value.

The absence of Time or Elevation parameters is equivalent to a request for the Layer's default value (if defined) along that dimension; see Section C.5. All parameter names are

© OGC 2001 – All rights reserved

case-insensitive as stated in Section 6.4.1, so, for example, 'TIME', 'Time' and 'time' are all acceptable.

For the TIME parameter, the special keyword 'current' **may** be used if advertised by the server as in Section C.3. The expression "TIME=current" means "send the most current data available." The expression "TIME=start_time/current" means "send data from start_time up to the most current data available."

C.4.2 Sample Dimension Values in Requests

Sample dimension parameters allow the Client to request a particular data object along one or more additional dimensional axes.

A request parameter name is constructed by concatenating the prefix 'dim_' with the sample dimension Name (the value of the name attribute of the corresponding <Dimension> and <Extent> elements in the Capabilities XML). The resulting "dim_name" is case-insensitive. The use of the 'dim_' prefix is to avoid clashes between server-defined dimension names and current or future OGC Web Service specifications. (Time and Elevation, being predefined, do not use the prefix.)

To include a sample dimension value in a request URL, the dim_name parameter is followed an equals sign '=' and a valid value, a comma-separated list, or an interval, as described in Table C.1.

Requests **may** omit a dimension parameter, or use null values, to request the default value (if supplied) along that dimension; see Section C.5.

EXAMPLE: A WMS Layer is described as having an Extent along a Dimension named "wavelength" with values 3000,4000,5000,6000 (Angstroms). WMS GetMap requests for that Layer may include the parameter "DIM_WAVELENGTH=4000".

C.4.3 Single- and Multiple-Valued Requests

Whether a request for a georeferenced object may include only a single value for each dimension or multiple values depends on the context. Multiple values are only acceptable when both the output format and the nature of the geospatial information permits it. An server **may** include a nonzero multipleValue attribute (Section C.2) in an Extent element to indicate that multi-valued requests are supported. A server **shall** issue a Service Exception if it cannot comply with the dimensional parameters as specified.

EXAMPLE 1: A Web Map Service offers maps in PNG image format of vehicle traffic density updated hourly. A valid GetMap request would include only a single TIME to select the desired map.

EXAMPLE 2: A WMS offers daily ozone maps in QuickTime movie format. The movie can contain multiple frames, so a valid GetMap request could include multiple times.

EXAMPLE 3: A WMS offers maps showing crime locations. Each crime has an associated date. A GetMap image request for all crimes occurring within a defined period would include a TIME range.

EXAMPLE 4: A Web Coverage Service offers atmospheric pollutant data at multiple elevations. A GetCoverage request could include a single elevation to obtain only a slice through the data, or multiple elevations to obtain a three-dimensional slab of data.

© OGC 2001 – All rights reserved

C.4.4 Applicability to Multiple Data Objects

Some OGC Web Service requests allow multiple georeferenced objects to be requested at once. For example, the WMS GetMap request includes a comma-separated list of one or more Layers. In such a request, the TIME, ELEVATION and DIM_ parameters apply to all of objects individually as follows:

– If the georeferenced object has the corresponding dimension as a property, then the dimension value applies as if that object had been requested alone.

– If the georeferenced object does not have that dimension, then the dimension value is ignored for the purposes of responding to the request for that object.

EXAMPLE: A Client's WMS GetMap request includes "LAYERS=temperature,coastlines&TIME=20010509" where 'temperature' is a daily temperature map (for which time is relevant) and 'coastlines' is a coastline map (which is static). The WMS responds with the static coastlines drawn atop the temperature for the specified date, without complaining that TIME does not apply to coastlines. If the date is invalid for the temperature layer, then the server may issue a Service Exception to that effect.

C.4.5 Example requests

The following are examples of multi-dimensional GetMap and GetCoverage requests.

EXAMPLE 1: WMS GetMap (single ozone Map at specified time and height):

```
VERSION=x.y.z  WIDTH=600
REQUEST=map    HEIGHT=300
LAYERS=ozone   TIME=2000-08-03
SRS=EPSG:4326  ELEVATION=1000
BBOX=-180,-90,180,90   FORMAT=image/gif
```

EXAMPLE 2: WMS GetMap (movie loop at specified frame times):

```
VERSION=x.y.z  WIDTH=600
REQUEST=map    HEIGHT=300
LAYERS=ozone   TIME=2000-07-01/2000-07-31/P1D
SRS=EPSG:4326  ELEVATION=1000
BBOX=-180,-90,180,90   FORMAT=video/mpeg
```

EXAMPLE 3: WCS GetCoverage (4D block of data with dimension x,y,z,t):

```
VERSION=x.y.z  SKIPX=10
REQUEST=coverage   SKIPY=10
LAYERS=layer   TIME=2000-07-01/2000-07-31/P1D
SRS=srs_identifier     ELEVATION=0/10000/1000
BBOX=minx,miny,maxx,maxy  FORMAT=application/x-hdf
```


© OGC 2001 – All rights reserved

C.5 Server Responses

C.5.1 Incorrect Values

If the dimension value is invalid or missing from the client request, and no default or nearest-value behavior was enabled as discussed in the following subsections, then the server **shall** respond with a Service Exception.

C.5.1 Default Values

An OGC Web Service **may** declare a default for any dimension. A 'default' is **optional** but **recommended**. If a request does not include a value along this dimension, and a default has been declared, then the server **shall** send the default value. If there is no declared default, then the server **shall** throw an Exception (code="MissingDimensionValue") to indicate that a value was required.

In order that the Client may determine what value was actually sent, the server **shall** label the response object with the actual value used by default. In the HTTP environment, the labeling is performed using an HTTP response header. For each dimension to which a default value was applied, a header line of the following form **shall** be sent:

```
Warning: 99 Default value used: DIM_NAME=value units
```

where "99" is a defined by HTTP [IETF RFC 2616] for use by miscellaneous warnings, DIM_NAME is the corresponding request parameter name, value is the value actually used, and units is the units attribute (see C.2) for that dimension.

C.5.2 Nearest Values

An OGC Web Service **may** declare that it will choose the closest available value for a dimension if an exact value is not specified. This allows, for example, hourly data whose actual recording time is precise to the millisecond to be requested simply by stating the desired date and hour. The nearestValue attribute of the <Extent> element, if present and nonzero, indicates that this behavior is enabled.

If a request does includes an imprecise dimensional value, and nearest value behavior has been declared, then the server **shall** compute and send the nearest available value. The value **shall** be rounded, not merely truncated. If rounding is not supported, then the server **shall** throw an Exception (code="InvalidDimensionValue") to indicate that a value was required.

In order that the Client may determine what value was actually sent, the server **shall** label the response object with the actual value found by rounding. In the HTTP environment, the labeling is performed using an HTTP response header. For each dimension to which a rounded-off value was applied, a header line of the following form **shall** be sent:

```
Warning: 99 Nearest value used: DIM_NAME=value units
```

© OGC 2001 – All rights reserved

where "99" is a defined by HTTP [IETF RFC 2616] for use by miscellaneous warnings, DIM_NAME is the corresponding request parameter name, value is the value actually used, and units is the units attribute (see C.2) for that dimension.

© OGC 2001 – All rights reserved

Annex D
(normative)

Conformance Tests

NOTE: A complete Conformance Testing Guideline document for WMS is presently under development as part of the OGC Conformance Testing Program. When complete, the Guideline will include a description and scope of each test suite, test data used in the tests, and documentation of the conformance items that consitute requirements for conformance. When a complete conformance test is available, its description will be added to this specification.

Minimal conformance with this specification requires the following:

1. The GetCapabilities and GetMap operations **shall** be supported.

2. The Extensible Markup Language (XML) document returned in response to a GetCapabilities request **shall** be valid against the Document Type Definition in Annex A.1. Such validation may be performed using commonly available XML validating tools.

3. The maps returned in response to a valid GetMap request **shall** be accurately registered according to the requested projection and bounding box.

4. All clauses in the normative sections of this specification that use the keywords "required", "shall", and "shall not" have been satisfied.

© OGC 2001 – All rights reserved

Annex E
(normative)

Automatic Projections

This Annex lists the identification codes defined thus far for automatic projections (see Section 6.5.5.2). In additional to the official codes defined below, for experimental purposes an informal list of additional values is maintained on-line in reference [12]. Experimental values may be rendered official in future versions of this specification. In all cases the AUTO projection codes are in the range 42000-42499, which is beyond the range reserved by EPSG (Section 6.5.5.1) for static projections.

The definitions use OGC Well-Known Text (WKT) format. In the notation below, "`${var}`" is a reference to the value of variable "`var`". The variables "lon0" and "lat0" are the central point of the projection appearing in the SRS parameter of the map request (see Section 6.5.5.2).

E.1 Auto Universal Transverse Mercator (AUTO:42001)

```
PROJCS["WGS 84 / Auto UTM",
    GEOGCS["WGS 84",
        DATUM["WGS_1984",
        SPHEROID["WGS_1984", 6378137, 298.257223563]],
        PRIMEM["Greenwich", 0],
        UNIT["Decimal_Degree", 0.0174532925199433]],
    PROJECTION["Transverse_Mercator"],
    PARAMETER["Central_Meridian", ${central_meridian}],
    PARAMETER["Latitude_of_Origin", 0],
    PARAMETER["False_Easting", 500000],
    PARAMETER["False_Northing", ${false_northing}],
    PARAMETER["Scale_Factor", 0.9996],
    UNIT["Meter", 1]]
```

where

```
central_meridian = -183.0 + ${zone} * 6.0
zone = min( floor( (${lon0} + 180.0) / 6.0 ) + 1, 60 )
false_northing = (${lat0} >= 0.0) ? 0.0 : 10000000.0
```


© OGC 2001 – All rights reserved

E.2 Auto Transverse Mercator (AUTO:42002)

```
PROJCS["WGS 84 / Auto Tr. Mercator",
    GEOGCS["WGS 84",
        DATUM["WGS_1984",
        SPHEROID["WGS_1984", 6378137, 298.257223563]],
        PRIMEM["Greenwich", 0],
        UNIT["Decimal_Degree", 0.0174532925199433]],
    PROJECTION["Transverse_Mercator"],
    PARAMETER["Central_Meridian", ${central_meridian}],
    PARAMETER["Latitude_of_Origin", 0],
    PARAMETER["False_Easting", 500000],
    PARAMETER["False_Northing", ${false_northing}],
    PARAMETER["Scale_Factor", 0.9996],
    UNIT["Meter", 1]]
```

where

```
central_meridian = ${lon0}
false_northing = (${lat0} >= 0.0) ? 0.0 : 10000000.0
```

E.3 Auto Orthographic (AUTO:42003)

```
PROJCS["WGS 84 / Auto Orthographic",
    GEOGCS["WGS 84",
        DATUM["WGS_1984",
        SPHEROID["WGS_1984", 6378137, 298.257223563]],
        PRIMEM["Greenwich", 0],
        UNIT["Decimal_Degree", 0.0174532925199433]],
    PROJECTION["Orthographic"],
    PARAMETER["Central_Meridian", ${central_meridian}],
    PARAMETER["Latitude_of_Origin", ${latitude_of_origin}],
    UNIT["Meter", 1]]
```

where

```
central_meridian = ${lon0}
latitude_of_origin = ${lat0}
```

© OGC 2001 – All rights reserved

E.4 Auto Equirectangular (AUTO:42004)

```
PROJCS["WGS 84 / Auto Equirectangular",
    GEOGCS["WGS 84",
        DATUM["WGS_1984",
        SPHEROID["WGS_1984", 6378137, 298.257223563]],
        PRIMEM["Greenwich", 0],
        UNIT["Decimal_Degree", 0.0174532925199433]],
    PROJECTION["Equirectangular"],
    PARAMETER["Central_Meridian", ${central_meridian}],
    PARAMETER["Latitude_of_Origin", 0],
    PARAMETER["Standard_Parallel_1", ${standard_parallel}],
    UNIT["Meter", 1]]
```

where

```
central_meridian = ${lon0}
standard_parallel = ${lat0}
```


© OGC 2001 – All rights reserved

Annex F
(informative)

Future Work

This Annex indicates future work expected on this specification.

F.1 UML Model

A Unified Modeling Language (UML) representation of web mapping that is independent of the HTTP Distributed Computing Platform treated in this implementation specification will be produced. The UML will be published either within this document or a separate Part 0. The needed material is expected to emerge in part from UML work undertaken by the OGC Documentation Subcommittee. As part of that work, and following on technical advances by the OGC Web Services testbed, Figure 1 (OGC Web Services Architecture Diagram) of the present document will be revised.

F.2 Support for HTTP Post

The specification will be expanded with a Part 2 to support operation requests using HTTP POST.

F.3 Use of XML Schema

The Capabilities XML Document Type Definition (DTD) will be replaced by an XML Schema document that more precisely dictates the format of the Capabilities XML.

F.4 Layer Identification Mechanism

The Capabilities XML will be revised, replacing some ad hoc constructs with a unified treatment of layer identification that includes metadata about the nature of the map information, its source, appropriate scales for display, and any cascading, caching, transformation or value-adding processes that may have occurred.

F.5 Adoption of Revised ISO 8601 Standard

Annex B of the present document introduces several extensions to ISO 8601:1988(E) to allow years before 0001 and to express the periodicity of data over a certain interval. The revised standard ISO 8601:2000 eliminates the need for those extensions. Annex B will be revised to take account of these changes in ISO 8601.

© OGC 2001 – All rights reserved

Bibliography

[1] Cox, S., Cuthbert, A., Lake, R., and Martell, R. (eds.), *Geography Markup Language (GML) Implementation Specification 2.0*, OGC Document #01-029, February 2001, <http://www.opengis.org/techno/specs/>.

[2] Cuthbert, A., *User Interaction with Geospatial Data*, OGC Document #98-060, October 1998.

[3] Cuthbert, A. (ed.), *Styled Layer Descriptor Draft Candidate Implementation Specification 0.7.0*, OGC Document #01-028, February 2001, <http://www.opengis.org/info/discussion.htm>.

[4] de La Beaujardiere, J. (ed.), *Web Map Service Implementation Specification 1.1.0*, OGC Document #01-047r2, June 2001, <http://www.opengis.org/techno/specs/>.

[5] Doyle, A.,*WWW Mapping Framework*, OGC Document #97-009, April 1997.

[6] Doyle, A. (ed.), *OpenGIS Web Map Service Implementation Specification 1.0.0*, OGC Document #00-028, April 2000, <http://www.opengis.org/techno/specs/>.

[7] Gardels, K., *A Web Mapping Scenario*, OGC Document #98-068, January 1998.

[8] Vretanos, P. (ed.), *Web Feature Service Draft Candidate Implementation Specification 0.0.12*, OGC Document #01-023, January 2001, <http://www.opengis.org/info/discussion.htm>

[9] Internet Assigned Numbers Authority, *Media Types*, <http://www.isi.edu/in-notes/iana/assignments/media-types/media-types>.

[10] OpenGIS Consortium, *Request for Technology In Support of a Web Mapping Technology Testbed*, October 1998.

[11] OpenGIS Consortium, *Request For Quotation and Call For Participation in the OGC Web Mapping Testbed Initial Operating Capability and Demonstration*, March 1999.

[12] OpenGIS Consortium, *Automatic Projections for Web Mapping*, <http://www.digitalearth.gov/wmt/auto.html>.

© OGC 2001 – All rights reserved