Web Terrain Service

Contents

i. Preface

This document comes out of work done on the OpenGIS Consortium (OGC) Military Pilot Project, Phase 1 (MPP-1) to define interoperable standards for requesting three-dimensional terrain scenes from a server capable of their generation. A "3D scene" is defined as a 2D projection of three-dimensional features into a viewing plane located at an arbitrary position and angle in 3D space, in such a way as to preserve the visual appearance of a third spatial dimension. A "terrain scene" is defined as a rendering of geometric representations of geographic entities upon a land surface in such a way as to simulate the view of a human observer. While there are many complications to implementing such functionality, the Web Terrain Service specification is intended to promote interoperability in Open Web Services by defining as simple an interface as possible for requesting such scenes from a diversity of scene generators.

ii. Submitting Organizations

The OGC Web Map Terrain Service RFC group submits this Document to the OGC Technical Committee as a Request for Comments of the OpenGIS® Web Terrain Service Implementation Specification.

iii. Submission Contact Points

All questions regarding this submission should be directed to the Editors:

Joshua E. Lieberman, Ph.D.
Syncline.com, Incorporated
373 Washington Street
Boston, MA 02108 USA
+1 617 603-2209
jlieberman@syncline.com

Jerome Sonnet
IONIC Software sa
Rue de Wallonie, 18
B-4460 Grâce-Hollogne
BELGIUM

+32 4 364 0 364
jerome.sonnet@ionicsoft.com

Additional Contributors

Nuke Goldstein
Syncline, Inc.
nuke@syncline.com

Jeff de La Beaujardière
NASA Goddard Space Flight Center
delabeau@sunrise.gsfc.nasa.gov

Markus Müller
Lat-lon
mueller@lat-lon.de

iv.   Revision History

Date	Release	Editor	Description
2001-05-05	0.1	Raj Sing	MPP1 Internal document
2001-06-27	0.2	Raj Singh	MPP1 Development document
2001-06-11	0.3	Raj Singh	MPP1 Draft
2001-08-24	0.3.2	Raj Singh	OGC Interoperability Program Report
2002-08-19	0.3.3	Joshua Lieberman	Incorporating comments from IPR
2003-09-22	0.5	Joshua Lieberman, Jerome Sonnet	Implementation Specification draft for RFC

v.   Changes to the OpenGIS Abstract Specification

The OpenGIS**®** Abstract Specification requires no change to accommodate this OpenGIS**®** standard:

vi.   Future work

This document describes a variation on Web Map Service operations for the purpose of supporting the return of images representing a 3D terrain viewpoint of feature, grid, and image data. These operations are presented here as an independent specification in the anticipation that more sophisticated 3D operations will continue to be added to future revisions of the specification.

It has been identified that the Web Terrain Service may use several other OpenGIS® Web Services as its sources of data. This general use case may require extensions to either or both SLD and Context specifications to describe the content of the scene as well as the rendering of these data. For example, SLD should be extended to describe the 3D rendering of features and the extrusion of 2D features. The Context should be able to access multiple sources of data such as WCS, WFS…

Foreword

This version of the specification cancels and replaces all previous versions.

Introduction

This document is a companion specification to the OpenGIS Web Map Service Interface Implementation Specification version 1.1.1 [4], hereinafter "WMS 1.1.1."
WMS 1.1.1 specifies how individual map servers describe and provide their map content. The present Web Terrain Service specification describes a new operation, GetView, and extended Capabilities which allow a 3D terrain view image to be requested, given a map composition, a terrain model on which to drape the map, and a 3D viewpoint from which to render the terrain view. A simple attempt is also made to reconcile 2D and 3D viewpoints by allowing the requested 3D area of view to be approximated with a WMS 1.1.1 bounding box.

Web Terrain Service

1   Scope

This specification applies to capabilities and contents of a Web Terrain Service This specification defines an encoding for service requests and responces using Key-Value Pairs, and for service capabilities using eXtensible Markup Language [XML 1.0].
This specification is a companion to the OGC Web Map Service Specification [WMS 1.0.0, WMS 1.1.0, WMS 1.1.1]. Reference is made to normative material from [WMS 1.1.1]. In some cases, reference is made to normative material from the Styled Layer Descriptor specification [SLD]

2   Conformance

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

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

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., <http://www.ietf.org/rfc/rfc2119.txt>.

SLD 1.0.0 (September 2002), *Styled Layer Description (SLD) 1.0.0*, Open GIS Consortium Proposed Implementation Specification, William Lalonde, http://www.opengis.org/techno/specs/02-070.pdf

WMS 1.1.1 (December 2001) OGC 01-068r2, Web Map Service Implementation Specification, v1.1.1, Jeff de la Beaujardiere, http://www.opengis.org/docs/01-068r2.pdf

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]

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

4.1
Capabilities
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.

5.2 Abbreviated Terms

CGI	Common Gateway Interface
EPSG	European Petroleum Survey Group
GIF	Graphics Interchange Format
GIS	Geographic Information System
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
URL	Uniform Resource Locator
WCS	Web Coverage Service
WFS	Web Feature Service
WMS	Web Map Service
XML	Extensible Markup Language

6   Overview of Web Terrain Service

6.1 Historical Background

The specification for receiving a two-dimensional map has been a focus of OGC interoperability work, and has been explored in Web Mapping Testbeds 1 and 2, the Upper Susquehanna-Lackawanna Pilot Project, and the Geospatial Fusion Services Testbed, as well as many others. These efforts have created and refined a simple, easily implemented common language for web mapping services. In the same spirit, this specification presents a simple baseline for web mapping services that incorporate terrain elevation and three-dimensional perspective.

Capabilities for visualization and analysis of multi-dimensional geodata which incorporate terrain elevation are becoming increasingly common in spatial software. The perspective offered by a third spatial dimension adds significant value to many analyses, such as viewshed studies, flight planning, and troop movements. Providing 3D terrain rendering and perspective can add significant complexity to a map specification. A goal of the Web Terrain Service specification is to encode the most important parameters of a three-dimensional "terrain view" into a specification that is easy to implement, remains consistent with 2D interfaces, and returns accurate results.

Reaching the goals of MPP-1 led initially to the definition of a new operation on a Web Mapping Service (WMS), GetView. For MPP1, the GetView operation can be implemented as an extension of the WMS interface, since a normal GetMap operation can be regarded as a base or restricted case of a GetView operation (i.e. orthographic looking straight down). In other words, a service that supports the GetView operation specified here can also fulfill all GetMap operations. This allows the implementers to reuse all the operations of their existing WMS implementation.

Extending the WMS interface for the GetView operation requires a minor extension of the WMS GetCapabilities response. However, fulfilling the full range of potential three-dimensional scenes requires significant extension of the WMS request parameters. Moreover, one could imagine that a Terrain View Service might eventually include other terrain-specific operations such as GetFootprint and GetViewshed. There might even be an analogous operation to GetFeatures for three-dimensional geometry and three-dimensional annotation of terrain views. These add so many additional complexities which are out of the scope of a 2D Web Map Service that a new Web Terrain Service is proposed here.

6.2 <u>Web Terrain Service and Interoperable Web Mapping</u>

A Web Terrain Service (WTS) produces views of georeferenced data. We define a "view" as a visual representation of geodata; a view is not the data itself. These views are generally rendered in a 2D pictorial format such as Portable Network Graphics (PNG), Graphics Interchange Format (GIF) or Joint Photographic Expert Group (JPEG) format. This specification standardizes the way in which clients request views and the way that servers describe their data holdings. Two required operations are defined:

> - **GetCapabilities:** Obtain service-level metadata, which is a machine-readable (and human-
> - readable) description of the WTS's information content and acceptable request parameters.
> - **GetView:** Obtain a 3D scene whose geospatial and dimensional parameters are indirectly defined.

Normative definitions of the GetCapabilities operation, as well as the GetMap operation from which GetView is derived, may be found in the Web Map Service Implementation Specification

v. 1.1.1. The changes to the Capabilities response required to support WTS are described later in this document.

The GetView operation defines parameters for an HTTP GET request, and an XML DTD for an HTTP POST request. It is intended that GetView may support a limited form of the Styled Layer Descriptor specification to support named styles and user layers. This will allow Web Terrain Services to extract data from Web Feature Services, drape them on the terrain, and style them appropriately.

The GetView operation of a WTS is normally run after a GetCapabilities response has indicated what queries are allowed and what data are available. The GetView operation returns an image in the specified format. Its syntax and semantics are similar to and generalized from the WMS GetMap operation.

6.3 Chaining Terrain Servers

Similar to having the Cascading Map Server component that behaves like a client of other WMSs there can also be a Cascading Terrain Server. This can provide a convenient mechanism for aggregating the capabilities of individual WTSs into one logical "place." In order for terrain views to be cascaded, the following need to be specified identically for each view: view point parameters, image size, SRS, DEM (elevation model), and (if specified) illumination model.

This set of parameters enables in principle the mapping of a point on the ground to the same pixel in the terrain view for each cascaded WTS. In practice and given the complex psychology of 3D vision, there may be additional properties of terrain views which need to match in order for cascaded terrain views to be convincingly composited.

A Cascading Terrain Server may perform additional services. For example, a Cascading Terrain Server that can convert among many different graphics formats may accommodate Viewer Clients which can only display GIF to utilize the output of WMSs that produce only JPEG or PNG.

6.4 Chaining Terrain Servers

A WTS may potentially get the data required to build the view from other OpenGIS® Web Services.
- Web Coverage Servers may provide the elevation data to model the terrain and some coverage information to be rendered in the scene as 3D shapes.

v. 1.1.1. The changes to the Capabilities response required to support WTS are described later in this document.

The GetView operation defines parameters for an HTTP GET request, and an XML DTD for an HTTP POST request. It is intended that GetView may support a limited form of the Styled Layer Descriptor specification to support named styles and user layers. This will allow Web Terrain Services to extract data from Web Feature Services, drape them on the terrain, and style them appropriately.

The GetView operation of a WTS is normally run after a GetCapabilities response has indicated what queries are allowed and what data are available. The GetView operation returns an image in the specified format. Its syntax and semantics are similar to and generalized from the WMS GetMap operation.

6.3 Cascading Terrain Servers

Similar to having the Cascading Map Server component that behaves like a client of other WMSs there can also be a Cascading Terrain Server. This can provide a convenient mechanism for aggregating the capabilities of individual WTSs into one logical "place." In order for terrain views to be cascaded, the following need to be specified identically for each view: view point parameters, image size, SRS, DEM (elevation model), and (if specified) illumination model.

This set of parameters enables in principle the mapping of a point on the ground to the same pixel in the terrain view for each cascaded WTS. In practice and given the complex psychology of 3D vision, there may be additional properties of terrain views which need to match in order for cascaded terrain views to be convincingly composited.

A Cascading Terrain Server may perform additional services. For example, a Cascading Terrain Server that can convert among many different graphics formats may accommodate Viewer Clients which can only display GIF to utilize the output of WMSs that produce only JPEG or PNG.

6.4 Chaining Terrain Servers

A WTS may potentially get the data required to build the view from other OpenGIS® Web Services.
- Web Coverage Servers may provide the elevation data to model the terrain and some coverage information to be rendered in the scene as 3D shapes.

- Web Map Servers may provide the image(s) to be draped onto the terrain, such as ortho-photos or rendered feature maps.
- Web Feature Servers may provide features with 2D geometry to be extruded as 3D object, for example annotations or building footprints, or features with 3D geometry to be directly rendered into the scene.

6.5 Styled Layer Descriptors

This specification applies to a Web Terrain Server which publishes its ability to produce views rather than its ability to access specific data holdings. A basic WTS classifies its georeferenced information holdings into "Layers" and offers one or more predefined "Styles" (including "default") in which to display those layers.

The behavior of a Web Terrain Server may be extended to allow user-defined symbolization of user-selected external feature data as an alternative to the pre-defined set of named Layers and Styles. The Styled Layer Descriptor (SLD) specification [19] describes this extension in the case of a WMS. In brief, an SLD-enabled WTS may retrieve features from a Web Feature Server [18], apply explicit styling information provided in the SLD request, then combine these features with a terrain or elevation model in order to render a terrain view.

An SLD WTS adds the following additional operation:

- DescribeLayer: Ask for an XML description of a map layer. The result is the URL of the WFS containing the data and the feature type names included in the layer.

6.6 Terrain Mapping Examples

The first example illustrates the special case of GetView. When the pitch is 90 degrees and the yaw is 0 degrees, the GetView response resembles that of GetMap, although the bounding box here is undefined and the view extent is determined with Point of Interest (POI) approach.

http://alpha.skylinesoft.com/services/ogc/WMS/WMS3D.asp?
server=WTS&request=GetView&version=0.3&srs=EPSG:4326&poi=84.405049,33.745012,0&distance=500&pitch=90&yaw=0&aov=53&width=512&height=300&format=jpeg&quality=medium

Example 2 is an informative response. Here the pitch is 30 degrees everything else being the same as the first example. Pitch is one of the new parameters (the others being shown in the table located under the 'Request Overview' section) that distinguish GetView from GetMap.

http://alpha.skylinesoft.com/services/ogc/WMS/WMS3D.asp?server=WTS&request=GetView&version=0.3&srs=EPSG:4326&poi=84.405049,33.745012,0&distance=500&pitch=30&yaw=0&aov=53&width=512&height=300&format=jpeg&quality=medium

7   Web Terrain Service Operations (Normative)

The two operations required for a Web Terrain Server are GetView and GetCapabilities. This section specifies the implementation and use of these WTS operations in the Hypertext Transfer Protocol (HTTP)[8] Distributed Computing Platform (DCP). Future versions may apply to other DCPs.

As previously stated, Web Terrain Service GetView operations will optionally support a limited form of the SLD specification to support named styles and user layers.

7.1 GetCapabilities (required)

7.1.1 General

The GetCapabilities request supported by a WTS returns enough information for a client to construct further, valid requests. For example the capabilities of a WTS include what NamedLayers are known to it. A GetView request is not valid if it references a NamedLayer not known to the WTS.

A WTS should advertise the GetView operation in its capabilities document.

<!ELEMENT Request (GetCapabilities , GetView, GetMap, DescribeLayer?) >

<!-- GetView interface: Presence of the View element means this server can generate a terrain view of a specified area as a picture -->
<!ELEMENT GetView (Format+, DCPType+)>

7.1.2 GetCapabilities Request Overview

The general form of a GetCapabilities request is defined in the Basic Services Model. When making this request of a WTS, which may offer other OGC Web Services as well, it is necessary to indicate that the client seeks information about the WTS in particular. Thus, the SERVICE parameter of the request must have the value "WTS" as shown in the table below.

7.1.3 Request Parameters

PREFIX=http://server_address/path/script?
The role of the URL prefix is specified in the Web Map Service 1.1.0 specification [2]. The prefixes for GetCapabilities, GetView and GetMap may be different.

VERSION=version

This parameter, and its use in version negotiation, is specified in the Basic Service Model.

SERVICE=service_name
This 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 WTS that implements this version of the specification or a later one, the service_name value "WTS" must be used.

REQUEST=GetCapabilities
This nature of this parameter is specified in the Basic Service Model.

7.1.4 **GetCapabilities Response**

The Basic Services Model specifies general rules about the GetCapabilities response. However, in the case of Web Terrain Server, GetCapabilities returns a service-level metadata of the WTS's information content and acceptable request parameters. GetView request parameter will indicate what layers are available and what service can be performed (i.e., SLD, cascading). In the particular case of complying with this version of the draft standard, the Extensible Markup Language (XML)[9] response must be valid according to the XML Document Type Definition (DTD) in Appendix A of this document. The DTD specifies the required and optional content of the response and how the content is formatted. Comments in the DTD include additional normative rigor not enforced by the XML DTD but required by this specification.
 A WTS may comply with other published or experimental versions, in which case it must support Version Negotiation as described in the Basic Services Model. Other DTDs are archived at http://www.digitalearth.gov/wmt/xml/.

7.2 **GetView (required)**

7.2.1 **General**

A View is a picture of a place which represents the normal viewpoint of a person, or at least of a camera or other imaging sensor. It is a three-dimensional, perspective (i.e. not orthographic[1]) view of the world. The way in which this perspective view can be specified is well defined in the photography and 3D modeling domains.

---

[1] In this formalism, an (optional) orthographic view is defined as the perspective view where the distance from the viewer to the terrain is arbitrarily large so that the lines of sight are essentially parallel.

VERTICAL (PITCH = 90)

OBSERVER

NORTH (YAW = 0)

DISTANCE (MAP UNITS)

YAW (0 - 360 deg)

PITCH (0-90 deg)

POI (POINT OF INTEREST)

Specification of 3D Perspective Terrain View
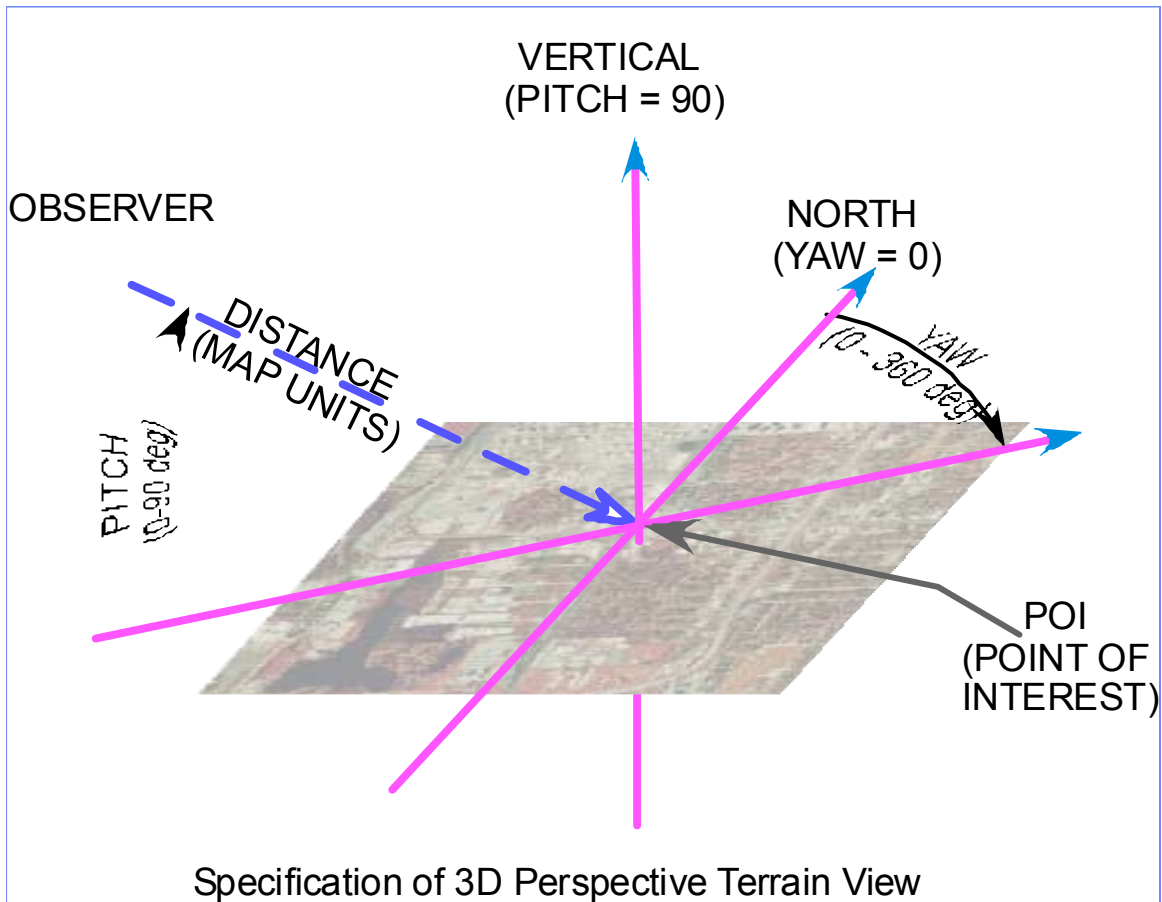
A set of parameters sufficient (but not unique) to define a view of a 3D scene is:

1. Point of Interest (POI): the exact location in x,y,z space of the viewer's focus.
2. Distance: the distance between the viewer and the POI in meters.
3. Pitch: the angle or inclination (in degrees) between the viewer and the POI (0º means the viewer is looking horizontally and -90º means the viewer is looking straight down on the POI).
4. Yaw: azimuth, the angle representing the "head swivel" (0º faces due north, 90º faces due east, etc.).
5. Angle of view (AOV): The angle representing the breadth of landscape in the viewer's scene. Also commonly described as the field of view or field of vision. Corresponds to the width of the scene intersecting the POI.

A traditional Web Map Server image is a special case of a Terrain View where the pitch is -90º (straight down), the yaw is 0º (directly north), and the distance is sufficiently large to produce essentially parallel (orthographic) sightlines.

Another special case involves setting the distance to zero, such that the POI becomes essentially the location of the observer. This might be useful where one wanted to see what terrain would be visible from a given fixed observer location given various angles and directions of view. In contrast to WMS requests, which are largely independent of intended usage, the most appropriate of several ways to request a particular terrain view will be dependent on the particular physical observer paradigm that is being visualized.

It should be noted that the perspective of a WTS GetView is independent of the spatial reference system in which the terrain data themselves are being presented, although the POI (or BBOX) will of course be specified in the same SRS as the terrain.

One difficulty with the GetView request as defined above is the lack of a bounding box. The surface region visible in a defined terrain from a defined perspective can be well defined, but may not be explicitly known. This visible region will in general take the form of a complex cone-shaped polygon with holes where topographic features obscure terrain behind them, as well as a farfield boundary possibly defined by limits on visibility. A future GetViewshed or GetFootprint operation will be able to return such a polygon for a given GetView request, but a GetView request will in general not be able to return a view which includes exactly a requested viewshed, in the way that a WMS or WFS can interact with a given BBOX.

One resolution of this practical problem is to define an alternative set of parameters for the GetView request, substituting a BBOX for the POI and DISTANCE parameters. Upon receiving these alternative parameters, a WTS will return an image of the most restricted terrain view that still includes the requested BBOX. A WTS may (or have no choice but to) only display data from within the BBOX. This approach will allow a reasonable degree of geographic coordination between a WTS and the various OGC services that rely on BBOX parameters in their requests. This algorithm could also be formulated as "the least restricted terrain view which lies completely within the requested BBOX", however this would require visibility conventions and calculations that are beyond the scope of the present specification.

A formulas and their derivation for converting between (POI & Distance) and BBOX are presented in Appendix C.

Upon receiving a GetView request, a Web Terrain Server must either satisfy the request or throw an exception in the format requested.

7.2.2 **GetView Request Overview**

The parameters of a GetView request are listed below. In addition to these, all parameters in a GetMap request are applicable, except BBOX, which becomes optional. The request is typically encoded as a URL, which is invoked on the WTS using the HTTP GET operation. (Reference: *Web Map Service Implementation Specification*, *Version 1.1.1*, OpenGIS Project Document 01-068r3, Jeff de La Beaujardière (NASA) Editor, June 2001, http://www.opengis.org/techno/specs/01-068r3.pdf) .

URL Component	Required/ Optional	Description
http://server_address/path/script?	R	URL prefix of server
VERSION=version	R	Request version
REQUEST=GetView	R	Request name
SRS=namespace:identifier	R	Spatial Reference Systems
POI=point_of_interest	R	x, y, z point in SRS units
PITCH=pitch	R	Angle of inclination
YAW=yaw	R	Azimuth
DISTANCE=dist_to_poi	R	Distance between the viewer and the POI in meters
AOV=angle_of_view intersecting the POI	R	The angle representing the breadth of landscape in the viewer's scene. Specifying AOV=0 indicates an orthographic (parallel sightline) projection.
BBOX=xmin,ymin,xmax,ymax	O	Minimum geographic extent in SRS units of the view (alternative to POI + DISTANCE)
Layers=layer_list	O	Comma-separated list of one or more map layers. Optional if SLD parameter is present.
Syles=style_list	O	Comma-separated list of one rendering style per requested layer. Optional if SLD parameter is present.
FORMAT	O	Mime type of requested response format
DEM=digital elevation model	O	Name expressing an available combination of elevation data and terrain model used to render a terrain view
TERRAIN=terrain base layer	O	Name expressing an available base image layer to underly any other layers in the terrain view
TRANSPARENT	O	Yes / No
BGCOLOR	O	Color to render any response pixel without terrain content
*SLD=styled_layer_descriptor document URL	O	Web-accessible URL of the SLD document
*SLD_Body=SLD text	O	text of StyledLayerDescriptor (subject to HTTP Get size limitations)
EXCEPTION=application/vnd.ogc.se_xml	O	The format in which exceptions are to be reported by the Map Server
QUALITY [0...100]	O	An integer between 0 and 100 that specifies the quality (e.g. data resolution, rendering accuracy) of the view to be returned.
Vendor-specific parameters	O	Discussed in the Web Map Service specification.

SLD_Body parameter is used only with Web Terrain Services that support the Styled Layer Descriptor specification[10].

7.2.3 <u>Request Parameters</u>

7.2.3.1 <u>Prefix</u>

The role of the URL prefix is specified in the WMS 1.1.1 specification [2]. The URL prefixes for GetCapabilities, GetView, and GetMap info may be different.

7.2.3.2 <u>Version</u>

This parameter is specified in the WMS 1.1.1 specification [2].

7.2.3.3 <u>Request</u>

This nature of this parameter is specified in the WMS 1.1.1 specification [2].
For GetView, the value "GetView" must be used.

7.2.3.4 <u>Layers</u>

The LAYERS parameter is a comma-separated list of one or more georeferenced layers. The first layer represents the terrain surface, while any subsequent layers are to be drawn on top of the terrain. [OR: zero or more layers, with terrain either specific to that WMS instance or requested by the TERRAIN parameter.] A WTS must render the requested layers by drawing the leftmost in the list bottommost, the next one over that, and so on.Valid layer names for a particular WTS are the character data content of any <Layer><Name> element in its Capabilities XML. If a Layer is given a Name, the WTS must accept that as a valid value for the LAYERS request parameter. If a Layer is not given a Name, it serves merely as a category for any other Layers nested within, and a Client must not attempt to request a view of an unnamed Layer. A server must throw an exception if an unadvertised Layer is requested.

7.2.3.5 <u>Styles</u>

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 view in the list of LAYERS is drawn using the corresponding style in the same position in the list of STYLES. Members of this list must be taken from the character data content of a <Style><Name> element in the Capabilities XML. A server must throw an exception if an unadvertised Style is requested. This style must be one which was defined in a <Style> element that is either directly contained within, or inherited by,

19

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 client requests 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.

7.2.3.6   SRS

The SRS (Spatial Reference System) parameter is specified in WMS 1.1.1 [2].

In the WTS Capabilities XML, there is a top-level Layer element which encapsulates all the other Layers. This parent element shall include a list of SRS values that every layer available from the Terrain Server supports (in other words, the set of values in the parent SRS list is the intersection of the set of values in the individual Layer SRS lists). This list may be empty if no SRS is common to all layers. Any SRS in the parent list is inherited by all child Layers. Subsidiary Layer elements may contain additional SRS supported by that Layer or by its children. Lower-level layers may repeat an SRS mentioned from a parent; such duplication must be ignored by clients.

7.2.3.7   BBOX

The BBOX (Bounding Box) parameter is specified in WMS 1.1.1 [2]. Except in certain special cases, the region visualized by a GetView request will not correspond exactly to any rectangular BBOX, so that the response to a BBOX request will be a matter of a defined protocol as discussed above.

7.2.3.8   POI

The POI (Point of Interest) parameter defines the x, y, z point in SRS units. For the case of geographic XY coordinates (degrees), the Z coordinate shall be given in meters.

7.2.3.9   Pitch

The PITCH parameter defines the angle of inclination. PITCH=0 is defined as a horizontal view (corresponding to the ecliptic of astronomical convention), with negative pitches inclining downwards, and positive pitches inclining upwards.

7.2.3.10   <u>Yaw</u>

The YAW parameter defines the Azimuth. YAW=0 corresponds to a top-to-the-north view.

7.2.3.11   <u>Distance</u>

The DISTANCE parameter defines the distance between the viewer and the POI in meters. This implies that setting DISTANCE = zero will make the POI coincident with the viewer.

7.2.3.12   <u>AOV</u>

The AOV (Angle of View) parameter defines the angle representing the breadth of landscape inthe viewer's scene. The terrain view image is a scaling to WIDTHxHEIGHT pixels of a rectangle which lies in a plane that is perpendicular to the line of sight and centered about the Point of Interest. The AOV is measured in a plane parallel to the WIDTH axis.

7.2.3.13   <u>Format</u>

The nature of the FORMAT parameter is specified in WMS 1.1.1 [2]. Allowed values for Web Terrain Server GetView requests are typically pictorial image formats, although a WTS implementation may also support 3D formats such as VRML. This specification does not require that a WTS support any specific format, but it does mandate the way in which formats are declared in WTS Capabilities XML and requested in a GetView operation. Formats are named using Multipurpose Internet Mail Extensions (MIME) [7] types such as image/png, image/gif, image/jpeg, and others.

The output format(s) supported by a particular WTS are listed in its Capabilities XML, in one or more <Format> elements. A WTS is not required to offer all known formats, but the server must advertise those formats it does support and must accept requests for any format it advertises. If a request contains a format not offered by a particular server, the server must throw a Service Exception (with code "InvalidFormat").

7.2.3.14   <u>Transparent</u>

The TRANSPARENT parameter can take on two values, TRUE or FALSE. The ability to return pictures drawn with transparent pixels allows results of different View requests to be layered, producing a composite view. As mentioned before, in order for two or more views to be cascaded, the same DEM (elevation model) and view parameters need to specified.

When TRANSPARENT is set to TRUE and the FORMAT parameter contains a Picture format, then a WTS must 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. When the Layer has been declared as "opaque," then significant portions, or the entirety, of the view may not be able to made transparent. For example, a view that contains a continuous orthoimage or that represents topography and bathymetry as 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 views. 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 WTS. The semantic of TRANSPARENT is that the "background" of the rendered terrain view should be transparent, but that features within the view which would normally be hidden by opaque terrain should still be hidden in the rendered view.

7.2.3.15 BgColor

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 and either upper or lower case characters are allowed for RR, GG, and BB values. The 0x prefix shall have a lower case x . When FORMAT is a Picture format, a WTS shall render its output on a background whose pixels were initially uniformly of the color encoded in BGCOLOR (or the default value 0xFFFFFF ["white"] if the BGCOLOR parameter is not present in the request). When FORMAT is a Graphic Element format (which does not have an explicit background), a WTS 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," then significant portions, or the entirety, of the view may not show any background at all.

7.2.3.16 Exceptions

The EXCEPTIONS parameter in a request indicates the format in which the Client wishes to be notified of Service Exceptions. The three possible values of this parameter for a WTS are defined below. See Section 6.7 of WMS 1.1.0 [2] for additional information about service exceptions.

Note that a GetView request may return nothing (due to circumstances beyond its control; this might result from the HTTP server's behavior caused by a malformed request, by an invalid HTTP request, by access violations, or any other causes). Terrain View clients should be prepared for this eventuality. A client should also be prepared for other returned values and types

because there is a possibility that a WTS is poorly behaved or that a request was directed at a non-WTS.

application/vnd.ogc.se_xml (default):
This represents an XML document containing the text of the error message(s) formatted according to the DTD in Appendix A.3 of WMS 1.1.0 [2].

application/vnd.ogc.se_inimage:
If the EXCEPTIONS parameter is set to application/vnd.ogc.se_inimage, an error message to the user shall be graphically returned as part of the content. This would usually take the form of text containing the message being painted into the returned view.

application/vnd.ogc.se_blank:
In the case of a Picture format, if the EXCEPTIONS parameter is set to application/vnd.ogc.se_blank, the WTS 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 BGCOLOR 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.

7.2.3.17   Vendor-specific Parameters

The use of Vendor-Specific Parameters (VSPs) is discussed in WMS 1.1.1 [2]. We repeat here only that clients may ignore any VSP they encounter in Capabilities XML, and that servers must not require the presence of VSPs in a request.

7.2.3.18   Quality

The QUALITY parameter allow the client to request a specified relative quality of view from 0 to 100. This allows a client to trade off response speed interactivity for view quality as needed.

7.2.3.19   DEM

In the case that a WTS allows layers to be draped over more than one Digital Elevation Model for scene rendering, that choice is made in the DEM parameter by specifying a DEM name from among those offered by that service.

7.2.3.20   <u>Terrain</u>

A Web Terrain Server may offer a choice of terrain base layers (or NONE) (e.g. orthoimagery) separately from the list of layers which may be draped on top (e.g. feature layers). The TERRAIN parameter specifies the name of such a base layer from among those offered by a particular service.

7.3 <u>Approximate Response to GetView</u>

The response to a valid GetView request must be a view of the georeferenced information layer requested, in the desired style, and having the specified SRS, POI, distance, pitch, yaw, AOV, size, format and transparency. An invalid GetView request must yield an error output in the requested Exceptions format (or a network protocol error response in extreme cases). In an HTTP environment, the Multipurpose Internet Mail Extensions (MIME) [7] type of the returned value's Content-type entity header must match the format of the return value.

A WTS may in some cases return only an approximation to the client's request. In the case that the client request specifies a BBOX, the returned view will be a best fit to the requested BBOX, as discussed in Sec. 3.2.1. As another example, it may not be able to provide the requested resolution or extent along various spatial or sample dimensions or the specified view Distance, AOV, etc. However, any Terrain View encoding will exactly specify its actual resolution or extent, and where it allows resampling, resizing, or subsetting. Moreover, the Capabilities XML metadata may also include three optional attributes on the Layer element that indicate a terrain server that is less functional than a normal WTS because it is not able to extract a subset of a larger dataset and/or because it only serves views of a fixed size and cannot resize them. Clients, therefore, should not make a priori assumptions about the resolution or extent of the returned Terrain View. Instead, the clients should consult the setting for a layer before requesting Terrain Views from it, and inspect each Terrain View reply before using or rendering it. The server can still respond to GetView requests for complete views in the original size.

noSubsets (boolean: 1 if true, 0 or absent if false).
When present and nonzero, indicates that the Server is not able to crop the data or map to a geographic area smaller than its enclosing bounding box.

FixedHeight, fixedWidth (integer: nonzero if image size is fixed, 0 or absent if image is resizable).
When present and nonzero, these indicate that the Server is not able to resize, resample or interpolate
the data to a different resolution than its native resolution.

All of these attributes are optional and default to 0.

Note: static image collections may not have a well-defined coordinate system, in which case the server must declare SRS=NONE as described in the Basic Services model.

Annex A

A.1   WTS Capabilities (DTD Normative)

```
<!-- NOTE: comments in this Document Type Definition impose additional
constraints beyond those codified in the DTD syntax.  A conformant Web Terrain
Server must provide Capabilities XML that (1) validates against the DTD and
(2) does not violate the constraints stated in comments herein. -->
<!-- The parent element of the Capabilities document includes as children a
Service element with general information about the server and a Capability
element with specific information about the kinds of functionality offered by
the server. -->
<!ELEMENT OGC_TS_Capabilities (Service, Capability)>
<!-- The version attribute specifies the specification revision to which this
DTD applies.  Its format is one, two or three integers separated by periods:
"x", or "x.y", or "x.y.z", with the most significant number appearing first.
Future revisions are guaranteed to be numbered in monotonically increasing
fashion, though gaps may appear in the sequence.  All known versions may
be found at http://ogc.syncline.com/xml/ogc/ -->
<!-- The updateSequence attribute is a sequence number for managing
propagation of the contents of this document.  For example, if a Terrain Server
adds some data layers it can increment the update sequence to inform catalog
servers that their previously cached versions are now stale.  The format is a
positive integer. -->
<!ATTLIST WMT_TS_Capabilities
  version CDATA #FIXED "0.3"
  updateSequence CDATA "0"
>
<!-- The Service element provides metadata for the service as a whole. -->
<!ELEMENT Service (Name, Title, Abstract?, KeywordList?, OnlineResource, ContactInformation, Fees?,
AccessConstraints?)>
<!-- A service name defined within the Web Mapping Specification namespace. -->
<!ELEMENT Name (#PCDATA)>
<!-- A human-readable title to briefly identify this server in menus. -->
<!ELEMENT Title (#PCDATA)>
<!-- A descriptive narrative for more information about this server. -->
<!ELEMENT Abstract (#PCDATA)>
<!-- List of keywords or keyword phrases to help catalog searching.
Currently, no controlled vocabulary has been defined. -->
<!ELEMENT KeywordList (Keyword*)>
<!-- A single keyword or phrase. -->
<!ELEMENT Keyword (#PCDATA)>
<!-- An HTTP URL of a service.  This may appear in several places:
- The Service element describing the service as a whole (in which case the
  OnlineResource may be, for example, the top-level "home page" of the service
  provider).
- The URLs for HTTP GET and POST requests of the specific interfaces offered
  by the service (e.g., the URL to use for a GetMap request).
- The Attribution information for Layer(s).
The use of an xlink:href attribute implies the HTTP Distributed Computing Platform;
currently, no non-HTTP platforms have been specified. -->
<!ELEMENT OnlineResource EMPTY>
<!ATTLIST OnlineResource
```

```
  xlink:href CDATA #REQUIRED
>
<!ELEMENT ContactInformation (ContactPersonPrimary?, ContactPosition?, ContactAddress?, ContactVoiceTelephone?,
ContactFacsimileTelephone?, ContactElectronicMailAddress?)>
<!--The primary contact person.-->
<!ELEMENT ContactPersonPrimary (ContactPerson, ContactOrganization)>
<!--The person to contact.-->
<!ELEMENT ContactPerson (#PCDATA)>
<!--The organization supplying the service.-->
<!ELEMENT ContactOrganization (#PCDATA)>
<!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.
The reserved word "none" indicates no constraint exists.  -->
<!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 WMT-defined request types are listed here. At least
one of the values is required, but more than one may be given. -->
<!ELEMENT Request (GetCapabilities, GetView, GetMap, DescribeLayer?)>
<!-- For each request method offered by the server, list the
available output formats and the supported distributed
computing platforms (DCPs). Example:
   <GetView>
    <Format>image/jpg</Format>
    <Format>image/tiff</Format>
    <DCPType><HTTP><Get>
      <OnlineResource xlink:type="simple" xlink:href="http://my.site.com/getview?" />
    </Get></HTTP></DCPType>
   </GetView>
 -->
<!-- GetMap interface: Presence of the Map element means this server can
generate a map of a specified area, either as a picture or a feature
collection -->
<!ELEMENT GetMap (Format+, DCPType+)>
<!-- GetView interface: Presence of the View element means this server
can generate a terrain view of a specified area as a picture. -->
<!ELEMENT GetView (Format+, DCPType+)>
<!-- GetCapabilities interface: Presence of the Capabilities element means
this server can generate a description of its abilities and holdings formatted
in XML that complies with this DTD.  -->
```

```
<!ELEMENT GetCapabilities (Format+, DCPType+)>
<!ELEMENT DescribeLayer (Format+, DCPType+)>
<!-- Available Distributed Computing Platforms (DCPs) are
listed here.  At present, only HTTP is defined.  -->
<!ELEMENT DCPType (HTTP)>
<!-- Available HTTP request methods. -->
<!ELEMENT HTTP (Get | Post)+>
<!-- HTTP request methods.  The OnlineResource attribute indicates the URL
prefix for HTTP GET requests; for HTTP POST requests, OnlineResource is the
complete URL.-->
<!ELEMENT Get (OnlineResource)>
<!ELEMENT Post (OnlineResource)>
<!-- Available formats.  Not all formats are relevant to all  requests.
Individual servers MAY add new formats as shown in the sample XML accompanying
this DTD. -->
<!ELEMENT Format (#PCDATA)>
<!-- An Exception element indicates which output formats are supported
for reporting problems encountered when executing a request.  Available
Exception formats MUST include one or more of WMS_XML, INIMAGE, or BLANK.
Example: <Exception><Format><INIMAGE /><WMS_XML /></Format></Exception>.  -->
<!ELEMENT Exception (Format)>
<!-- see WMS DTD 1.0.8 for comments on VendorSpecificCapabilities -->
<!-- DEFINE THIS ELEMENT AS NEEDED IN YOUR XML
 <!ELEMENT VendorSpecificCapabilities (your stuff here) >
-->
<!-- Optional user-defined symbolization -->
<!ELEMENT UserDefinedSymbolization EMPTY>
<!ATTLIST UserDefinedSymbolization
  SupportSLD (0 | 1) "1"
  UserLayer (0 | 1) "1"
  UserStyle (0 | 1) "1"
  RemoteWFS (0 | 1) "1"
>
<!-- Nested list of zero or more map Layers offered by this server. -->
<!-- A Layer element has two functions: it either refers to a map layer which
can be requested by Name in the LAYERS parameter of a GetMap or GetView request, or
it is a category Title for all the layers nested within.  In the latter case, the
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 "County Roads" and allow the user to request either child
individually or both together.  -->
<!-- A Server which advertises a Layer containing a Name element MUST be
able to accept that Name as the value of LAYERS argument in a GetMap request
and return the corresponding map.  A Viewer Client MUST NOT attempt to request a
Layer that has a Title but no Name.  -->
<!-- A Server MUST include at least one <Layer> element for each map layer
offered. If desired, data layers MAY be repeated in different categories when
relevant. A Layer element MAY state the Name by which a map of the layer is
requested, MUST give a Title to be used in human -readable menus, and MAY
include: a human-readable Abstract containing further description, available
Spatial Reference Systems (SRS), bounding boxes in Lat/Lon and SRS -specific
coordinates indicating the available geographic coverage, styles in which the
layer is available, a URL for more information about the data, and a hint
concerning appropriate map scales for displaying this layer.  Use of the
nesting hierarchy is optional. -->
<!-- The following table specifies the number and source of the various
elements (and one attribute) describing a Layer that has a Name.  Without a
Name, the Layer is merely a category title and all other elements are
optional; if present, some of those elements may be inherited by children as
described in the table.-->
<!ELEMENT Layer (Name?, Title, Abstract?, KeywordList?, SRS?, LatLonBoundingBox?, BoundingBox*, Dimension*, Extent*,
Attribution?,DEM?,TERRAIN?, MetadataURL*, DataURL?, FeatureListURL?, Style*, ScaleHint?, Layer*)>
<!-- see WMS DTD 1.0.8 for comments -->
```

```
<!ATTLIST Layer
  opaque (0 | 1) "0"
  noSubsets (0 | 1) "0"
  fixedWidth CDATA #IMPLIED
  fixedHeight CDATA #IMPLIED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT SRS (#PCDATA)>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT LatLonBoundingBox EMPTY>
<!ATTLIST LatLonBoundingBox
  minx CDATA #REQUIRED
  miny CDATA #REQUIRED
  maxx CDATA #REQUIRED
  maxy CDATA #REQUIRED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT BoundingBox EMPTY>
<!ATTLIST BoundingBox
  SRS CDATA #REQUIRED
  minx CDATA #REQUIRED
  miny CDATA #REQUIRED
  maxx CDATA #REQUIRED
  maxy CDATA #REQUIRED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT Dimension EMPTY>
<!ATTLIST Dimension
  name CDATA #REQUIRED
  units CDATA #REQUIRED
  unitSymbol CDATA #IMPLIED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT Extent (#PCDATA)>
<!ATTLIST Extent
  name CDATA #REQUIRED
  default CDATA #IMPLIED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT Attribution (Title?, OnlineResource?, LogoURL?)>
<!ELEMENT LogoURL (#PCDATA)>
<!ATTLIST LogoURL
  width NMTOKEN #REQUIRED
  height NMTOKEN #REQUIRED
>
<!ELEMENT DEM EMPTY>
<!ATTLIST DEM
  name CDATA #REQUIRED
>
<!ELEMENT TERRAIN EMPTY>
<!ATTLIST TERRAIN
  name CDATA #REQUIRED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT MetadataURL (#PCDATA)>
<!ATTLIST MetadataURL
  type (TC211 | FGDC) #REQUIRED
  format (XML | SGML | TXT) #REQUIRED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT DataURL (#PCDATA)>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT FeatureListURL (#PCDATA)>
```

```
<!ATTLIST FeatureListURL
  format (%KnownFormats;) #REQUIRED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT Style (Name, Title, Abstract?, LegendURL*, StyleSheetURL?, StyleURL?)>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT LegendURL (#PCDATA)>
<!ATTLIST LegendURL
  format (%KnownFormats;) #REQUIRED
  width NMTOKEN #IMPLIED
  height NMTOKEN #IMPLIED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT StyleSheetURL (#PCDATA)>
<!ATTLIST StyleSheetURL
  format (XSL) #REQUIRED
>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT StyleURL (#PCDATA)>
<!-- see WMS DTD 1.0.8 for comments -->
<!ELEMENT ScaleHint EMPTY>
<!ATTLIST ScaleHint
  min CDATA #REQUIRED
  max CDATA #REQUIRED
>
```

A.2   Service Exception (DTD Normative)

```
<!ELEMENT ServiceExceptionReport (ServiceException*)>
<!ATTLIST ServiceExceptionReport version CDATA #REQUIRED>
<!ELEMENT ServiceException (#PCDATA)>
<!ATTLIST ServiceException code CDATA #IMPLIED>
```

A.3   Sample WTS Capabilities (XML Informative)

```
<?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_0_7.dtd" [

<!-- Output formats known to this server are defined here
if necessary (only if a format not already listed in the WMT
Capabilities DTD is needed).  To define a new format, place
an entity definition for KnownFormats like the one below in
the DOCTYPE declaration of your Capabilities XML, listing at
minimum all of the formats you support and separating each
by logical-OR (|) characters.  Then, define a new element
for any formats not predefined by WMT.  For example, in the
following list "SGI" is a server -specific format, while all
the others are known WMT-wide.  Thus, SGI is included in the
    KnownFormats list and a new empty element <SGI/> is  defined. -->

<!ENTITY % KnownFormats " SGI | GIF | JPEG | PNG | WebCGM | SVG | GML.1
 | WMS_XML | MIME | INIMAGE | PPM | BLANK " >
<!ELEMENT SGI EMPTY> <!-- Silicon Graphics RGB Format  -->
 <!-- other vendor-specific elements defined here -->
 <!ELEMENT VendorSpecificCapabilities (YMD)>
 <!ELEMENT YMD (Title, Abstract)>
 <!ATTLIST YMD required (0 | 1) "0">
  ]>  <!-- end of DOCTYPE declaration -->

<!-- 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.  -->
```

```xml
<WMT_MS_Capabilities version="1.0.7" updateSequence ="0">
<!-- Service Metadata -->
<Service>
  <!-- The WMT-defined name for this type of service  -->
  <!-- Human-readable title for pick lists  -->
  <Name>GetView</Name>
  <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 xlink:href="http://hostname/" />
  <!-- Contact information -->
  <ContactInformation>
    <ContactPersonPrimary>
      <ContactPerson>person_name</ContactPerson>
      <ContactOrganization>organization_name</ContactOrganization>
    </ContactPersonPrimary>
    <ContactPosition>Computer Scientist</ContactPosition >
    <ContactAddress>
      <AddressType>postal</AddressType>
      <Address>address_here</Address>
      <City>city_here</City>
      <StateOrProvince>state_here</StateOrProvince>
      <PostCode>postcode_here</PostCode>
      <Country>USA</Country>
    </ContactAddress>
    <ContactVoiceTelephone>phone_here</ContactVoiceTelephone>
    <ContactFacsimileTelephone>fax_here</ContactFacsimileTelephone>
<ContactElectronicMailAddress>email_here</ContactElectronicM ailAddress>
  </ContactInformation>
  <!-- Fees or access constraints imposed.  -->
  <Fees>none</Fees>
  <AccessConstraints>none</AccessConstraints>
</Service>
<Capability>
  <Request>
    <GetView>
      <Format>
        <SGI />
        <GIF />
        <JPEG />
        <PNG />
        <GML.1 />
      </Format>
      <DCPType>
        <HTTP>
          <Get>
            <!-- The URL here for invoking GetView using HTTP GET includes
            only the prefix before the '?' and the query string.  Similar to WMS clients, WTS clients are
            expected to append '?WMTVER=nnn&request=view&etc' as described in  the WMS
            specification.  -->
            <OnlineResource xlink:href="http://hostname:port/path" />
          </Get>
        </HTTP>
      </DCPType>
```

```
    </GetView>
    <DescribeLayer>
     <Format>
       <!-- include a valid format specified in the WTS specification  -->
       <GML.1 />
     </Format>
     <DCPType>
      <HTTP>
        <Get>
          <OnlineResource xlink:href="http://hostname:port/path" />
        </Get>
      </HTTP>
     </DCPType>
    </DescribeLayer>
    <Capabilities>
     <Format>
       <WMS_XML />
     </Format>
     <DCPType>
      <HTTP>
        <Get>
          <!-- The URL here for invoking GetCapabilities using HTTP GET
          includes only the prefix  before the '?' and the query string.
          Clients are expected to append '?WMTVER=nnn&request=capabilities'
          as described in the WMS specification.  -->
          <OnlineResource 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.  Not all Terrain Servers support POST.  -->
          <OnlineResource xlink:href="http://hostname:port/path" />
        </Post>
      </HTTP>
     </DCPType>
    </Capabilities>
   </Request>
   <Exception>
    <Format>
      <BLANK />
      <WMS_XML />
    </Format>
   </Exception>
   <!-- Any text or markup is allowed here, as required to describe
       server-specific options.  Please define elements and attributes
       in the DOCTYPE declaration at the start of the document.  -->
   <VendorSpecificCapabilities>
    <YMD required="0">
      <Title>Date in YYYYMMDD format</Title>
      <Abstract>8-digit date in YYYYMMDD format.  If absent,
        the latest available date (usually today, but not for
        non-daily measurements) is sent.</Abstract>
    </YMD>
   </VendorSpecificCapabilities>
   <UserDefinedSymbolization SupportSLD="1" UserLayer="1" UserStyle="1"
       RemoteWFS="1" />
   <Layer>
    <Title>title_name</Title>
    <SRS>EPSG:xxxx</SRS> <! -- all layers are available in at least this SRS  -->
    <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</Name>
```

```
    <Title>Roads</Title>
    <!-- The following characteristics are inherited by subsidiary layers . -->
    <SRS>EPSG:xxxxx</SRS> <! -- An additional SRS for this layer  -->
    <LatLonBoundingBox minx=" -71.634696" miny="41.754149" maxx=" -70.789798"
maxy="42.908459"/>
    <BoundingBox SRS="EPSG:26986" minx="189000" miny="834000" maxx="285000"
maxy="962000"/>
    <Attribution>
     <Title>title_name</Title>
     <OnlineResource xlink:href="http://www.hostname.org/" />
     <LogoURL width="100"
height="100">http://www.hostname.org/icons/logo.gif</LogoURL>
    </Attribution>
    <FeatureListURL
format="GML.1">http://www.hostname.org/data/roads.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 format="GIF" width="72"
height="72">http://www.hostname.org/legends/usgs.gif</LegendURL>
     <!-- An XML (GML) encoding of the legen d for GML-enabled clients -->
     <LegendURL format="GML.1">http://www.hostname.org/legends/usgs.gml</LegendURL>
     <!-- An XSL stylesheet for rendering this layer in this style when requested in GML format -->
     <StyleSheetURL format="XSL">http://www.hostname.org/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>
     <!-- Metadata specific to this particular layer.  The same FGDC metadata is offered
in two formats. -->
     <MetadataURL type="FGDC"
format="TXT">http://www.hostname.org/fgdc/clearinghouse/metadata/roads.txt</MetadataURL>
     <MetadataURL type="FGDC"
format="XML">http://www.university.edu/ fgdc/clearinghouse/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 format="GIF" width="72"
height="72">http://www.university.edu/legends/atlas.gif</LegendURL>
     </Style>
    </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>
```

```
  </Layer>
</Capability>
</WMT_MS_Capabilities>
```

Annex B: WTS GetView DTD

The GetView DTD specifies the format of GetView requests which are transmitted via HTTP POST to the WTS.

B.1   WTS GetView (DTD Normative)

```
<!--  G E T V I E W    -->

<!--  The GETVIEW DTD includes the SLD DTD as an external entity reference.  -->
<!ENTITY % SLDDTD SYSTEM "sld.dtd">
%SLDDTD;

<!--  This is a WMT-wide list of possible output formats that can be
redefined by individual servers.    -->
<!ENTITY % KnownFormats " image/gif | image/jpeg | image/png | image/ppm | image/tiff | image/geotiff | image/svg
| image/hdf - eosi | image/webcgm | image/svg | image/wms_xml | image/gml.1 | image/gml.2 | image/gml.3 |
image/wbmp " >

<!-- GetView interface: Presence of the View element means this server can generate a terrain view of a specified area as
a picture. The output format is specified by the outputFormat attribute.  A GetView request is defined by a description of
how the view should appear (the StyledLayerDescriptor) and information about this particular request, for example the
extent of the view being requested and the display SRS expressed as a DISTANCE and POI, PITCH, YAW, and AOV)   -->

<!ELEMENT GetView (POI, Pitch, Yaw, Distance, AOV), Format, BoundingBox, StyledLayerDescriptor, Width, Height,
ExceptionFormat?, Transparent?, BackgroundColor?, VendorSpecificParameters?) >

<!-- A request version number should be encoded.  Perhaps like this: -->
<!ATTLIST GetView
  version CDATA #FIXED "0.3.2" >

<!--  The POI of the view measured in SRS.    -->
<!ELEMENT POI (#PCDATA) >
<!ATTLIST POI
    x CDATA #REQUIRED
    y CDATA #REQUIRED
    z CDATA #REQUIRED >

<!--  The Pitch of the view measured in degrees.    -->
<!ELEMENT Pitch (#PCDATA) >

<!--  The Yaw of the view measured in degrees.    -->
<!ELEMENT Yaw (#PCDATA) >

<!--  The Distance of the view measured in pixels.    -->
<!ELEMENT Distance (#PCDATA) >

<!--  The AOV of the view measured in degrees.    -->
<!ELEMENT AOV (#PCDATA) >

<!--  Format in which to return map.    -->
<!ELEMENT Format (#PCDATA) >

<!--  Optional choice of DEM or TERRAIN.    -->
```

```
<!ELEMENT DEM (#PCDATA) >
<!ELEMENT TERRAIN (#PCDATA) >
<!-- The BoundingBox attributes indicate the edges of the bounding
box in units of the specified spatial reference system. This is a
repeat of the BoundingBox element defined in the WMS Capabilities
document.     -->
<!ELEMENT BoundingBox EMPTY >
<!ATTLIST BoundingBox
        SRS CDATA #REQUIRED
        minx CDATA #REQUIRED
        miny CDATA #REQUIRED
        maxx CDATA #REQUIRED
        maxy CDATA #REQUIRED >

<!-- The Width of the map measured in pixels.     -->
<!ELEMENT Width (#PCDATA) >

<!-- The Height of the map measured in pixels.     -->
<!ELEMENT Height (#PCDATA) >

<!-- Indicates how exceptions should be reported back to the client.
The default is INIMAGE.     -->
<!ELEMENT ExceptionFormat (SE_XML|INIMAGE|BLANK) >
<!ELEMENT SE_XML EMPTY>
<!ELEMENT INIMAGE EMPTY>
<!ELEMENT BLANK EMPTY>

<!-- Indicates how the background pixels on a map where the format
permits transparency should be handled. The default is not
transparent.     -->
<!ELEMENT Transparent (TRUE|FALSE) >
<!ELEMENT TRUE EMPTY>
<!ELEMENT FALSE EMPTY>

<!-- Specifies the color that should be assigned to background
pixels. This is a hexadecimal encoding of an RGB value (#RRGGBB)
with a default value of 'white' (#FFFFFF).     -->
<!ELEMENT BackgroundColor (#PCDATA) >
```

Annex C WTS BBOX to POI+Distance Calculation

This appendix describes a protocol for deriving best-fit POI and DISTANCE parameters for a terrain view from a requested BBOX.

C.1 Conversion Protocol

As discussed above, there is no 1-to-1 correlation between a geographic bounding box on the ground (BBOX) and a 3-D terrain view, simply because the true geographic extent of a terrain view comprises a complex polygon with interior and exterior boundaries defined not only by the view parameters, but also by topographic complexities and visibility / horizon constraints. The complete polygonal region is usually termed a "viewshed", while its outer boundary is termed a "footprint".

Still, it is useful to define a protocol for translating a bounding box into an "equivalent" terrain view for compatibility with other operations. For example, one might want to use for GetMap for data exploration and then use the same BBOX to obtain a corresponding terrain view. The first level of this correspondence would be to match the centers of view, e.g.:

1) [(xmax + xmin)/2, (ymax + ymin)/2, groundlevel] = POI (Point of Interest)

This does not completely address the issue of equivalent extent, since a terrain view may appear rather differently than a map view, so that figuring out the right scale (distance + angle-of-view) may be difficult.

Calculation of Bounding 3D Perspective Terrain View from BBOX
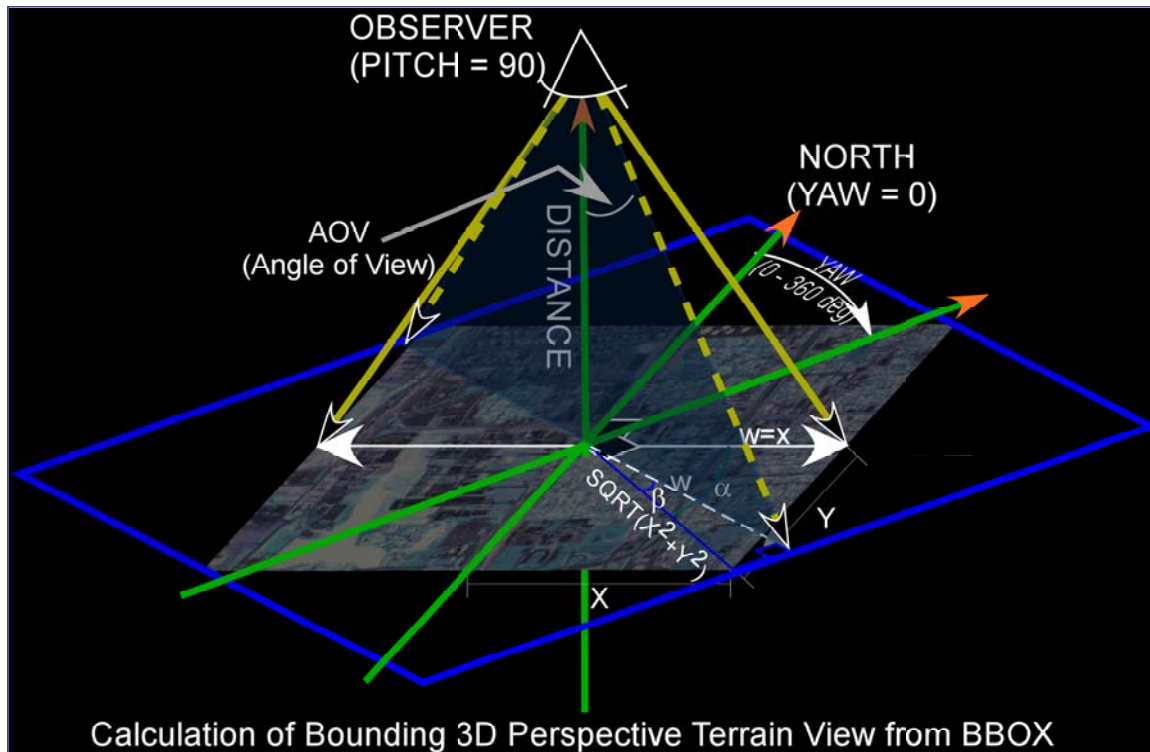
A second level of correspondence is proposed, in which perspective parameters are chosen to approximate the minimum footprint which contains the requested BBOX. This method depends on the fact that whatever the pitch of the terrain view, the width ($2W$) of the resulting view does not change to a great degree. The simple case is one where the yaw is set to 0 (north) or 180 degrees. In this case, the necessary width of the view equals the width of the BBOX. Simple trigonometry then returns the needed distance for a given AOV:

2) $DISTANCE = 2*(W / \arctan(AOV))$

When the yaw is set to 90 or 270 degrees, the width ($2W$) would be equal to the height of the BBOX. The general situation for any yaw between 0 and 360 degrees is a little more complicated, but expressed by the following formula:

If x and y are half the width and height of the BBOX, respectively, and $\alpha = \arctan(y/x)$ then

3) $\beta = ABS(ABS(180-ABS(90-YAW))-90) - \alpha$

and

4) $W = \text{SQRT}(x^2 + y^2) \cos (\beta)$

Eq. 2) would then apply to calculate **DISTANCE** or **AOV**.

Annex D

NOTE: A complete Conformance Testing Guideline document for Web Terrain Services may be taken up under the OGC Conformance Integration and Test Environment initiative. 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 constitute 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. Web Terrain Service Instance Capabilities Documents shall be valid against the included DTD. This validity test shall use standard DTD validation techniques.
2. All clauses in the normative sections of this specification that use the keywords "required", "shall", and "shall not" have been satisfied.