Open Geospatial Consortium

Publication Date: 2014-04-14

Approval Date: 2013-11-25

Submission Date: 2012-09-20

External identifier of this OGC® document: http://www.opengis.net/doc/bp/wms-tnz/1.0

Internal reference number of this OGC® document: 12-111r1

Version: 1.0

Category: OGC® Best Practice

Editors: Marie-Françoise Voidrot-Martinez, Chris Little, Jürgen Seib, Roy Ladner, Adrian Custer, Jeff de La Beaujardière
Stephan Siemen

OGC Best Practice for using Web Map Services (WMS) with Time-Dependent or Elevation-Dependent Data

Warning

This document defines an OGC Best Practice on a particular technology or approach related to an OGC standard. This document is not an OGC Standard and may not be referred to as an OGC Standard. It is subject to change without notice. However, this document is an official position of the OGC membership on this particular technology topic.

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type: OGC® Publicly Available Best Practice
Document subtype: Application profile
Document stage: For release
Document language: English

License

This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it. None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable

i. Abstract

This document proposes a set of best practices and guidelines for implementing and using the Open Geospatial Consortium (OGC) Web Map Service (WMS) to serve maps which are time-dependent or elevation-dependent. In particular, clarifications and restrictions on the use of WMS are defined to allow unambiguous and safe interoperability between clients and servers, in the context of expert meteorological and oceanographic usage and non-expert usage in other communities. This Best Practice document applies specifically to WMS version 1.3, but many of the concepts and recommendations will be applicable to other versions of WMS or to other OGC services, such as the Web Coverage Service.

ii. Keywords

The following are keywords to be used by search engines and document catalogues:

meteorology oceanography time elevation 'time-dependent' 'elevation-dependent' wms 'web map service' 1.3 1.3.0 ogc 'best practice' ogcdoc

iii. Preface

This Best Practice document is the result of discussions within the Meteorology and Oceanography Domain Working Group (MetOcean DWG) of the Technical Committee (TC) of the Open Geospatial Consortium (OGC) regarding the use of the OGC Web Map Service (WMS) to provide map visualizations from the various types of data regularly produced, analyzed, and shared by those communities. The discussion considered the differences in the types of data as well as the issues, concerns, and responsibilities of data producers when sharing those data as maps with end users, including analysts within the meteorological and oceanographic communities, users with specific needs and the general public. The limited scope of the requirements and recommendations in this document reflects the consensus reached by groups with vastly different types of data, limitations in the current design of the WMS specification, compromises to ensure these services remain applicable to a mass market audience, and a wish to release a first version of document without undue delay. Future work includes extending this Best Practice once the community gains more experience with WMS and with implementing the provisions of this document. This document does not require any changes to other OGC standards but it is hoped that the WMS specification will evolve to address issues encountered in this work such as providing a mechanism to define exclusive dimensions and to define sparse combinations of dimensions.

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

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

iv. Submitting organizations

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

DWD (Deutscher Wetterdienst)
ECMWF (European Centre for Medium-range Weather Forecasts)
Météo-France
Technische Universität Dresden
UK Met Office
US COMNAVMETOCCOM (Naval Meteorology and Oceanography Command)
US NOAA (National Oceanic and Atmospheric Administration)

v. Submitters

All questions regarding this submission should be directed to the editor or the submitters:

Name	**Affiliation**
Marie-Françoise Voidrot-Martinez	Météo-France
Chris Little	UK Met Office
Jürgen Seib	DWD
Roy Ladner	US COMNAVMETOCCOM
Adrian Custer	unaffiliated
Jeff de La Beaujardière	US NOAA
Stephan Siemen	ECMWF

1. Introduction

The meteorological and oceanographic communities have been exchanging information internationally for at least 150 years and well understand the importance of geospatial standards for interoperability. These standards have typically defined data formats, interfaces, processes, shared conceptual models, and sustainable maintenance processes.

Because of the demanding nature of meteorological and oceanographic data processing, the communities have evolved domain specific solutions. However, as computers have become more powerful, it has become feasible to use general geospatial software for day-to-day operational purposes, and interoperability problems have arisen. There has also been an increasing need to combine meteorological and oceanographic data with other forms of geospatial data from other domains, in ways convenient for those domains.

Meteorological and oceanographic data are inherently multidimensional in time and space. Annex A provides a formalized description of some of the issues for these temporal and spatial dependencies for the context of Web Map Services (WMS). Serving these data through WMS can be done with a variety of implementation choices or interpretations.

This document describes and justifies a set of best practices for offering and requesting meteorological and oceanographic data with temporal or elevation dependencies through WMS. This set of best practices is intended to meet the interoperability requirements of the meteorological and oceanographic communities and enable them and their customers to gain the economic benefits of using commercial off the shelf (COTS) software implementations of WMS servers and clients.

1.1 Time

Complex geospatial data such as that in common use by the meteorological and oceanographic communities may be temporally independent or may have one or more temporal dependencies that may express the relation of the data to different time references such as applicable time, collection time or other significant times. The WMS specification permits such data to be offered through a WMS in many different ways including many different WMS layers with or without a temporal dimension, or as a single WMS layer with many temporal dimensions. This document makes recommendations to promote interoperability by focusing on the most common temporal dependencies in the domain, by developing a consistent terminology for such data, and by defining an approach to structuring such data for distribution as layers in a WMS.

The WMS specification provides for definition of a time value, a single timestamp, a time interval, or a list of timestamps and time intervals. A **timestamp** t is a sequence $<c_1,\ldots,c_n>$, $1 \leq n \leq 7$, of time components, where the components are defined as follows:

- c_1 stands for a year. A value is between 1 and 9999. Negative values or more than 4 characters are also allowed by WMS1.3, Annex D and ISO8601.
- c_2 stands for a month. A value is between 1 and 12.
- c_3 stands for a day. A value is between 1 and 31.
- c_4 stands for an hour. A value is between 0 and 23.

- c_5 stands for a minute. A value is between 0 and 59.
- c_6 stands for a second. A value is usually between 0 and 59 and rarely 60 for leap seconds.
- c_7 stands for a millisecond. A value is between 0 and 999.

The precision of a timestamp t is determined by the last time component of t. Timestamps may be associated with a time zone. If no time zone is specified with a timestamp t, then t is assumed to be in local time. If a timestamp t should be considered as a time in UTC, then the character Z has to be added to the string representation of t. The string representation of a timestamp is built from the time components and specific separators. A full string representation has the following format:

"YYYY-MM-DDThh:mm:ss.SSSZ"

where:

YYYY indicates a 4-digit year
MM indicates a month
DD indicates a day of a month
T is the separator between the date part and the time part
hh indicates an hour
mm indicates a minute
ss indicates a second
SSS indicates a millisecond
Z is the time zone designator for the zero UTC offset

Timestamps are comparable. A timestamp t_1 is before a time stamp t_2 if and only if the string representation of t_1 is lexicographically less than the string representation of t_2. A timestamp t_1 is after a time stamp t_2 if and only if the string representation of t_1 is lexicographically greater than the string representation of t_2. Two timestamps are equal if and only if their string representations are equal.

This definition of comparison ensures that two timestamps of different precision are different. Consider for example the timestamps t_1 = 2013-01-22 and t_2 = 2013-01-22T00:00:00Z. t_1 has a day precision and t_2 is exact to a second. t_1 is before t_2. They are not equal.

A **time interval** is a triple $t_{min}/t_{max}/r$ where t_{min} and t_{max} are timestamps that define the lower and upper bounds of the interval and r is the resolution. The interval contains all timestamps $t_{min} + i * r$, $i >= 0$, that are lower or equal than t_{max}.

A resolution r is represented by the format P [n_1Y] [n_2M] [n_3D] [T [n_4H] [n_5M] [n_6S]] where

P is a starting character.
Y is the year designator that follows the value n_1 for the number of years.
M is the month designator that follows the value n_2 for the number of months.
D is the day designator that follows the value n_3 for the number of days.
T is the time designator that precedes the time components of the representation.
H is the hour designator that follows the value n_4 for the number of hours.
M is the minute designator that follows the value n_5 for the number of minutes.
S is the second designator that follows the value n_6 for the number of seconds.

A time interval with a resolution value of zero means that the interval has an infinitely-fine resolution. It contains all possible timestamps between the lower and the upper bound of the interval. In this case the resolution part can be omitted and the interval should be written as a pair t_{min}/t_{max}.

Data that have no temporal dependency can be offered as single WMS layers without any associated temporal dimension. For example, topographic data or data for the climatological normal[1] may be offered as WMS layers without any WMS dimensions, implicitly making the assumption that any temporal dependency is irrelevant over the time scale under consideration.

Data that have a single temporal association also can be treated simply being offered as single WMS layers with a single temporal dimension. For example, satellite imagery taken from the same platform for the same spatial extent at different moments in time can be offered as a single WMS layer with a single temporal dimension. Here, the temporal dimension represents a time used to identify individual satellite products (conventionally it is usually the beginning or the end of the scan).

Data that have multiple temporal associations, such as numerical weather forecasts, series of alerts, or data associated with trajectories of moving objects, do not have a single obvious representation as WMS layers. The WMS standard only defines a single temporal dimension called time. Each user community has the flexibility of defining the use of that dimension, the possibility of defining other temporal dimensions for structuring this multi-temporal data, and the work of structuring common metadata using those temporal axes. This may result in splitting such data into many layers, each representing different temporal associations, or using several temporal dimensions to reduce the number of WMS layers.

Numerical weather forecast calculations present an example of data with multiple temporal associations. Numerical weather forecasts, for example, may have three separate calculations for air temperature, each containing 48 forecasts at successive 1 hour intervals. In such a case, the available combinations of validity time and calculation nominal start time are as shown in the table below. Columns sharing an "*" across rows indicate forecasts from different calculations (i.e., nominal start times) that share the same validity time.

[1] WMO defines a normal as "period average computed for a uniform and relatively long period comprising at least three consecutive 10 year periods.

	Validity Time of Forecasts					
	2009-11-26		2009-11-27		2009-11-28	
Nominal Start Time	00-12	12-00	00-12	12-00	00-12	12-00
2009-11-26T00:00	*	*	*	*		
2009-11-26T12:00		*	*	*	*	
2009-11-27T00:00			*	*	*	*

Table 1: Example of two dimensional time axis for numerical model forecast data

The forecasts associated with these three calculations might be represented in a WMS as 144 separate layers (with or without an associated time dimension) or as a single layer with two time dimensions. The former approach will result in a larger, more complex capabilities document; the latter approach will permit the valid combinations of nominal start time and validity time to be expressed more compactly as a single layer.

Warnings and alerts are common categories of meteorological data that often have multiple temporal associations. A meteorological warning is a message about a meteorological event that will probably occur in the future at a geographic location. Typically, a warning becomes effective when it has been sent out. But, it will be also possible to specify an effective time explicitly. The validity period of a warning is the period between the expected beginning of the event in the warning and its expiry.

For example, the Common Alerting Protocol (CAP) Version 1.2, which is an OASIS standard format for exchanging all-hazard emergency alerts and public warnings, defines four time attributes for warnings and alerts. These are detailed in the following table.

Element name	**Definition**	**Optionality**	**Notes**
sent	The time and date of the origination of the warning	required	
effective	The effective time of the information of the warning	optional	If the effective time is not explicitly defined, then this time shall be assumed to be the same as the sent time.
onset	The expected time of the beginning of the subject event of the warning	optional	
expires	The expiry time of the information of the warning	optional	If this value is not provided, each recipient is free to set its own policy as to when the warning is no longer in effect.

Table 2: Time related attributes in CAP Version 1.2

According to these definitions, the validity period of a warning is the period between the onset time and the expiry time, if the onset time is given. Otherwise, it is the period between the effective time and the expiry time.

A warning is valid at a particular time if it has not expired at that moment. A warning situation is a set of valid warnings at a particular time. A warning situation changes if a new warning is sent out or if an existing valid warning is withdrawn or modified. Typical modifications are the adjustment of the validity period or the change of the severity level. Figure 1 shows a warning situation on 12th August 2012 at 13:00 UTC drawn as a blue line. The diagram contains five warnings. But only three of them are part of the warning situation at the time under consideration because the first two are or will be expired.

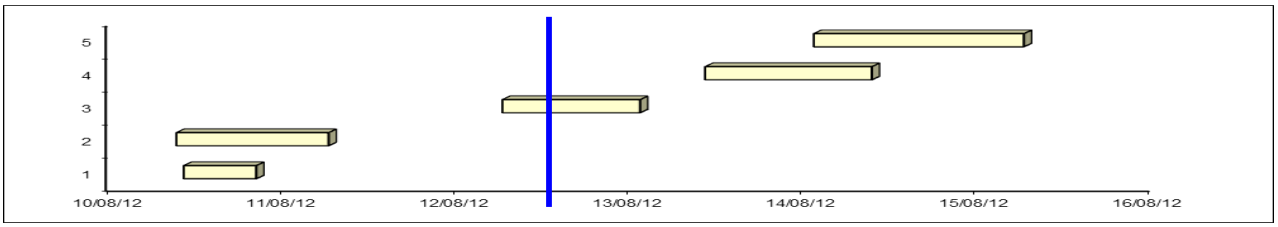

Figure 1: Example of a warning situation

The WMS specification provides the flexibility of representing the warning shown in Figure 1 in a WMS as five separate layers (with or without a time dimension) or as a single layer with two time dimensions (e.g., effective time and validity period).

Trajectory data present a third example of meteorological data that often have multiple temporal associations. A trajectory is the path that a moving object follows through space as a function of time.

Trajectories can run forward or backward. Good examples of forward trajectories are trajectories for volcanic ash. They are usually calculated using the data of a numerical weather forecast calculation. Such a forward trajectory predicts the movement of air masses from a given geographical position, the location of the volcano. In this case, the trajectory has the same temporal associations as the numerical weather forecast because it is based on these data. For example, if the data of the forecast calculation contain 48 forecasts at successive 1 hour intervals, then the trajectory can have up to 48 validity times.

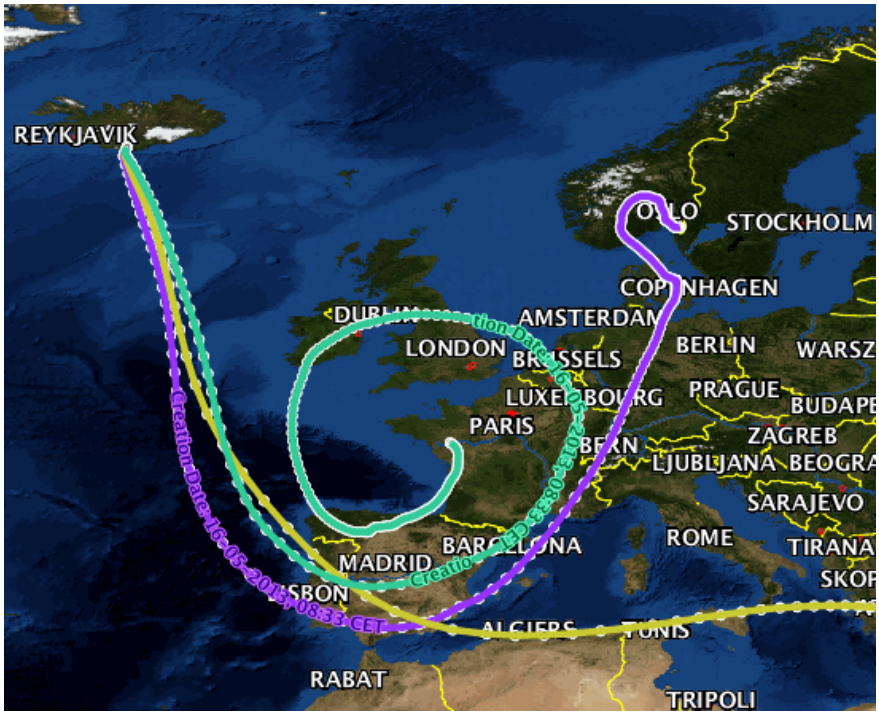

Figure 2: Forward trajectories from the volcano Katla

Consider Figure 2. It shows three forward trajectories for the forecast movement of volcanic ash released into the upper troposphere at the geographical position of the volcano Katla in Iceland. The trajectories are calculated by Deutscher Wetterdienst's global numerical weather prediction model. The model output used in this example dates from 2013-05-16T00:00:00Z and includes forecasts for seven days. Each trajectory belongs to a different height regime (pink = lower, yellow = middle, sea green = higher). The temporal resolution is about 6 hours.

Figure 3 shows another example of trajectory data with multiple temporal associations, but for a hypothetical scenario. The trajectories are calculated for a simulated hydrocarbon leak in the English Channel at the black dot locating the source of the pollution beginning January 15th at 12:00 UTC. The marine pollution model of Météo-France is used for the calculation. The Figure represents the forecast pollution of the model's calculations on April 23rd forecast for April 26th at 12:00 UTC, combining the result of simulations made every 30 minutes over a temporal period of 2460 hours from January 15th.

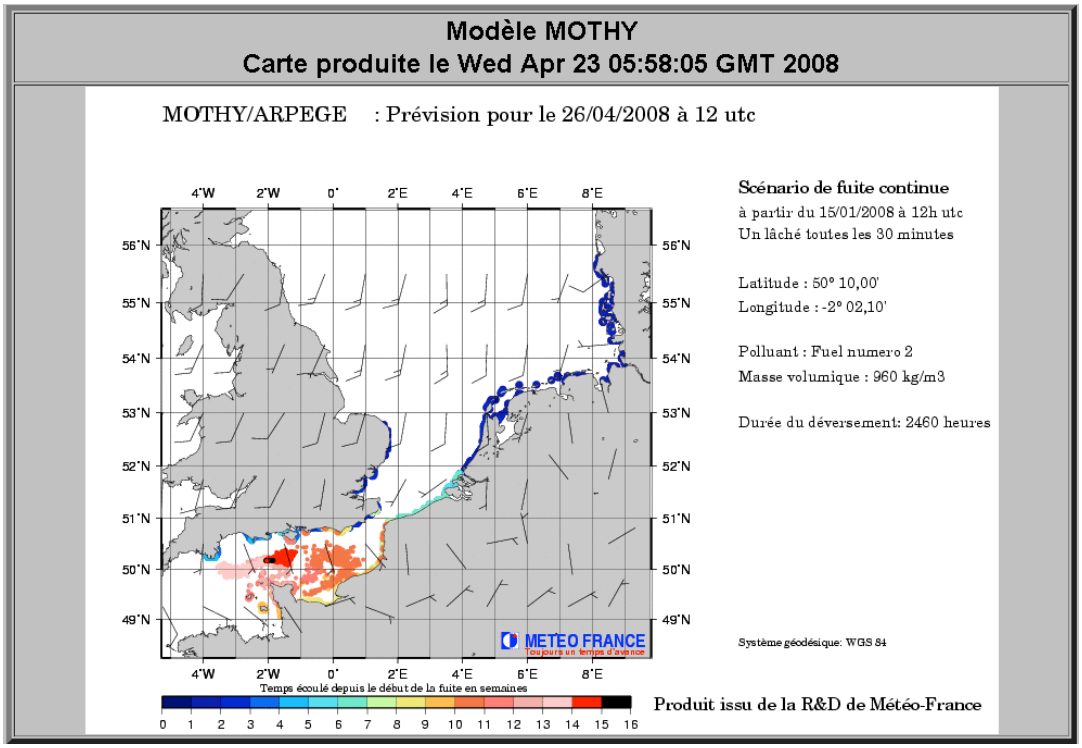

Figure 3: Simulation of the resulting hydrocarbons pollution due to a continuous leak over 2460 hours

1.2 Elevation

Complex geospatial data such as that in common use by the meteorological and oceanographic communities may be independent of elevation or may have one or more vertical dependencies.

The WMS specification permits such data to be offered through a WMS in different ways: as many layers with or without a vertical dimension, or as a single layer with many vertical dimensions. Elevations can also be specified using multiple measure types including nominal values such as 'earth surface' 'tropopause,' or 'cloud top', or numeric values specifying the distance above or below some vertical datum. Furthermore, data may have multiple vertical dependencies that are mutually exclusive; unfortunately the WMS standard does not specify any mechanism to declare two dimensions on a given layer but make them mutually exclusive to prevent clients asking for values in both dimensions at once. This document makes recommendations to promote interoperability by focusing on the most common elevation dependencies in the domain, by developing a consistent terminology for such data, and by defining an approach to structuring such data for distribution as layers in a WMS.

The following examples taken from the meteorological and oceanographic community illustrate different types of elevation-dependent data:

- A data set recording total cloud cover or bathymetry values may have no elevation dependency
- A data set recording air temperature at a given numeric height has an elevation dependency that can be specified by using a numeric value for the elevation and associating that value to a vertical coordinate reference system (CRS) specifying the vertical datum anchoring the height value

- A data set recording an environmental parameter on a computed surface, such as the tropopause, an isotherm, the base (or top) of contrail, a surface of constant potential vorticity, or a jet level, has an elevation dependency that can be specified using a nominal value identifying the surface. In general, it is best if these nominal values are defined by some authority possibly through a controlled vocabulary
- A data set recording ocean salinity at multiple, fixed depths has an elevation dependency that can be specified by using multiple numeric values for the depths and associating those values to a vertical CRS specifying the vertical datum anchoring the depth value
- A data set recording the wind speed of one or more atmospheric 'layers,' or 'non-zero vertical thickness surfaces,' between two isobaric levels (e.g. 1000-900 hPa) has an elevation dependency that can be specified, for each layer, through a pair of numeric elevation values specifying the bottom and the top of the layer along with a vertical CRS. The recorded values are related to the full thickness.

Fully defined vertical CRSs are critical for the proper expression of elevation using numeric values. Sources of predefined vertical CRS definitions include:

- World Meteorological Organization (WMO) *Code Table 4.5* http://www.wmo.int/pages/prog/www/WMOCodes/WMO306_vI2/LatestVERSION/LatestVERSION.html
- EPSG Geodetic Parameter Registry http://www.epsg-registry.org/EPSG.

New, custom vertical CRS definitions can be created using the data structures of the standard:

- *ISO 19111: 2007 Geographic information -- Spatial referencing by coordinates*

and expressing those in data structures in any of several binary formats, notably in XML.

2. Scope

This version of this Best Practice document intentionally addresses a limited number of issues related to the use of WMS for time-dependent and elevation-dependent data in order to produce an initial document as a basis for future expansion. The document considers the issues with some of the most common time-dependent and elevation-dependent data.

The document describes how to offer WMS layers for:

- data with no temporal relation to 'validity time' (i.e. the Gregorian moment to which the map applies),
- data with a temporal relation to 'validity time' and
- data with a temporal relation to 'validity time' and also to some other relevant reference time.

The document also describes how to offer WMS layers for:

- data with no vertical dimension,
- data with a single vertical dimension expressed as numeric values, as nominal values, as a zero thickness surface, or as a non-zero thickness volume, and

- data with several vertical dimensions based on separate vertical coordinate reference systems.

The document specifies the use of the dimensions `time` and `elevation` defined in the WMS standard and also the use of a newly defined dimension `reference_time`.

The document also specifies constraints on the behavior of WMS clients that have been created specifically to use WMS implementations that follow the requirements of this document. This document specifies a constrained, consistent interpretation of the WMS 1.3 standard that is applicable to government, academic, or commercial providers or users of time-dependent or elevation-dependent data offered as a WMS product.

This version of this Best Practice document has left many issues out of its scope. Design issues with WMS such as issues related to offering a large number of layers or to offering data that are updated frequently were not directly tackled. Issues in the workflow of users relying of a distributed spatial data infrastructure such as the discovery of services, or directly of layers, were not examined. Rules for the use of data using climatological periods, climatological ranges and non-Gregorian calendars proved too complex for this version. No work was done to address issues related to expressing the semantic content of particular layers. Developing a mechanism to obtain visualizations of non-horizontal data such as vertical slices was considered but rejected as this would require modification of the design of WMS itself. Related work on the internationalization of human visible text and on styling has been undertaken as separate efforts.

3. References

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

CAP Common Alerting Protocol Version 1.2, OASIS Standard, 2010-07-01. http://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.html .

ISO 8601:2004, Data elements and interchange formats — Information interchange — Representation of dates and times.

ISO 19111:2007 Geographic information -- Spatial referencing by coordinates.

ISO 19156:2011, Geographic information -- Observations and measurements.

OpenGIS® Web Map Server Implementation Specification Version 1.3.0. 2006-03-15. http://portal.opengeospatial.org/files/?artifact_id=14416 .

WMO No. 306, Manual on Codes, World Meteorological Organization operational data formats. http://www.wmo.int/pages/prog/www/WMOCodes/WMO306_vI2/LatestVERSION/LatestVERSION.html .

4. Terms and Definitions

This document uses the terms defined in Sub-clause 5.3 of [OGC 06-121r8], which is based on the ISO/IEC Directives, Part 2: Rules for the structure and drafting of International Standards.

In particular:

1. **SHALL** – verb form used to indicate a requirement to be strictly followed to conform to this document, from which no deviation is permitted.

2. **SHOULD** – verb form used to indicate desirable ability or use, without mentioning or excluding other possibilities.

3. **MAY** – verb form used to indicate an action permissible within the limits of this document.

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

4.1 Client

Software component that can invoke an **operation** from a **server**

4.2 Elevation

Vertical distance of a point or level, above or below some datum, expressed in a predefined Coordinate Reference System

4.3 Feature

Abstraction of real world phenomena [ISO 19101:2002, definition 4.11]

4.4 Feature collection

A collection of features together with a set of properties that apply to these features. GML defines an XML schema for feature collections. When a GetFeatureInfo request is made for GML output, data is always returned as a GML feature collection.

4.5 Geographic information

Information concerning phenomena implicitly or explicitly associated with a location on Earth [ISO 19101]

4.6 Interface

Named set of **operations** that characterize the behaviour of an entity [ISO 19119]

4.7 Layer

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

4.8 Map

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

4.9 Named surface

Surface of specific interest for meteorology, oceanography or users that meets predefined physical criteria such as TROPOPAUSE, LEVEL_OF_MAXIMUM_WIND

4.10 Operation

Specification of a transformation or query that an object may be called to execute [ISO 19119]

4.11 Portrayal

Presentation of information to humans [ISO 19117]

4.12 Reference time

Temporal parameter used to represent a time axis that can be mapped to some relevant referent time other than validity time. The semantic meaning can differ for different types of data

4.13 Request

Invocation of an **operation** by a **client**

4.14 Response

Result of an **operation** returned from a **server** to a **client**

4.15 Server

A particular instance of a **service**

4.16 Service

Distinct part of the functionality that is provided by an entity through interfaces [ISO 14252]

4.17 Timestamp

Terminology used to unambiguously indicate a moment or instant in time using ISO 8601 notation, as distinct from a time interval, duration or period.

4.18 Validity period

Period of time when it is valid to use data, a map or layer. Often used for warnings or forecasts. In [ISO 19156], the validity period has the semantics of validTime

4.19 Validity time

An attribute value specified by an instant in, or duration of, universal chronological time that identifies when information is valid or applicable. In [ISO 19156], the validity time has the semantics of phenomenonTime. Deciding if the data have a ‘validity time’ is an important step.

5. Conventions

5.1 Abbreviated terms

CAP	Common Alerting Protocol
CRS	Coordinate Reference System
GIS	Geographic Information System
GRIB	Gridded Binary data format
IEC	International Electro-technical Commission
ISO	International Standards Organisation
OGC	Open Geospatial Consortium
Met Ocean	Meteorology and Oceanography
URL	Uniform Resource Locator
WMO	World Meteorological Organization
WMS	Web Map Service
XML	eXtensible Mark-up Language

5.2 Notational conventions

This sub-clause provides details and examples for any conventions used in the document. Examples of conventions are symbols, abbreviations, use of XML schema, or special notes regarding how to read the document.

Any keywords from the WMS Standard are depicted in `Courier font`, lower case if the words ‘time’ or ‘elevation’ refer to the dimension, UPPER case if it refers to the parameter of the GetMap request.

Any example values for keywords from the WMS Standard or this Best Practice document are depicted in *lower case italic.*

6. Requirements

This document defines injunctions for the use of OGC Web Map Service implementations for the distribution of map visualizations based on data with temporal or elevation dependencies. The injunctions constrain how such data offerings should be structured into WMS layers, the way such layers should be described in the WMS Capabilities document, the way requests for such layers should be handled, and the way clients should issue requests for such layers. These injunctions do not necessitate modifications to the WMS standard but merely define rules for the use of that standard.

The injunctions of this document target server and client implementations. “Conformant WMS servers” are WMS server implementations that follow the injunctions of this document to offer time-dependent or elevation-dependent data as WMS layers. This document places restrictions on how these layers will be structured so that it will be useful to general purpose client software applications and to advanced client software applications specifically built for the needs of the community. “Conformant WMS clients” are WMS clients that are expected to provide a user interface to select dimensional values; this document places certain restrictions on these clients to reduce the chance of confusion or error.

The WMS 1.3.0 standard defines a system for declaring and requesting map layers with more dimensions than the two spatial dimensions represented in the map visualization. The standard defines a mechanism to assign dimensions to a map layer in the Capabilities document and then defines two specific dimensions, `time` and `elevation`. A WMS Layer is declared to be available at one or more values in a dimension by declaring an XML `<Dimension>` element either in the `<Layer>` element itself or in a parent layer, in which case the dimension will be inherited. For example, a Capabilities document might contain:

```
<Layer>
   ...
   <Dimension name="time" units="ISO8601" default="1970-01-01"
              nearestValue="0" current="0"
              >1900-01-01/2010-12-31/P1D</Dimension>
   <Layer>
       <Name>Surface_Irradiance</Name>
       ...
   </Layer>
   <Layer>
       <Name>Temperature</Name>
       ...
       <Dimension name="elevation" units="computed_surface" unitSymbol=""
                  default="surface" multipleValues="0" nearestValue="0"
                  >surface,tropopause</Dimension>

   </Layer>
</Layer>
```

where the outer layer defines a temporal dimension which is inherited by both inner layers and the inner layer `Temperature` also declares that it is available at two heights. The time dimension is declared with a `<Dimension name="time" ...>` element and the corresponding GetMap request may include the condition `TIME=`*t* with the parameter name TIME and an appropriate value *t*. The

`elevation` dimension is declared with a `<Dimension name="elevation" ...>` element and the corresponding GetMap request may include the condition `ELEVATION=`*z* with the parameter name ELEVATION and an appropriate value *z*. Further dimensions can be declared with a `<Dimension name="somename" ...>` element and the corresponding GetMap request may include the condition `DIM_SOMENAME=`*v* with the parameter name DIM_SOMENAME and an appropriate value v.

While the mechanism defined by WMS 1.3.0 is powerful, it unfortunately also has limitations. The mechanism does not provide any way to declare that dimensions are exclusive of one another. As a result, elevation-dependent data accessible using different vertical units, say meters and millibars, must be split into separate layers (otherwise GetMap requests with both values would cause problematic errors). The mechanism does not provide any alternate way to declare what combinations of values in the different dimensions are available, making the discovery of available combinations a guessing game for the client.

Another limitation is that the WMS 1.3.0 protocol does not provide a general mechanism to return supplemental information for a request. However, the need for such a mechanism is widespread, especially for the handling of multi-dimensional layers. Consider for example a GetMap request with no value for a dimension of the requested layer. The WMS server shall use the default value in such a case, but the issuer of the request will not know the value that was actually used in preparing the response. Chapter C.4.2 of the WMS 1.3.0 specification addresses this issue and proposes the use of the HTTP response header for a transmission of the default value.

A second scenario is the applicability of dimension values in GetMap requests for multiple layers. Chapter C.3.5 of the WMS 1.3.0 specification considers this case and requires conformant WMS servers to apply a dimension value that is given in a GetMap request only to those layers that have that dimension. The value is ignored for all other layers. Here, a map is returned and the issuer of the request will not know the layers to which the dimension value has been applied. A solution, similar to the one described in chapter C.4.2, could be the use of the HTTP response header.

The solution with the HTTP response header has several drawbacks, however. It is only available to clients aware of its presence and able to get the header, and the solution only works when the WMS messaging is directly embedded in the HTTP header rather than, say, embedded in XML (i.e. SOAP). The solution is also unreliable as a source of error prevention because the absence of the header does not indicate anything since the request may have been satisfactory.

The need for supplemental information for GetMap requests is one of the motivations for the development of WMS 2.0. This Best Practice document will therefore neither make requirements nor recommendations for how this problem should be handled. It seems better to wait for the solution developed in WMS 2.0.

This clause specifies requirements and recommendations for the use of the `time` dimension and the `elevation` dimension defined in the WMS standard, and for the use of a new dimension named `reference_time`. This specification considers only the OGC standard:

OpenGIS® Web Map Server Implementation Specification, version 1.3.0 (OGC 06-042)

which is the only version of the standard that has not been deprecated at the time of writing of this Best Practice document.

6.1 Time Independent Data

Data providers may wish to offer WMS layers that have no time dependency that can be mapped to ’validity time‘. In this case the following requirements apply:

Req. 1 Conformant WMS servers SHALL not offer data as a WMS layer for which a dimension named `time` is declared or inherited in the Capabilities document if these data have no temporal dependency that can be mapped to ‘validity time.’

6.2 Time-Dependent Data

Complex data sets can have temporal dependencies of many kinds. This Best Practice document focuses on two in particular.

The most pervasive temporal dependency comes from data values having different chronological times: observations may have been recorded at different moments, models may be predicting conditions for different days, alerts may refer to conditions expected at certain moments in the future, and warnings may apply to the entire duration of a time interval or part of it. This document adopts the phrase 'validity time' in reference to this type of temporal dependency despite the potentially misleading natural language association of the root 'validity.' This concept is essentially identical to the concept of 'phenomenonTime' from the standard *ISO 19156:2011, Geographic information -- Observations and measurements*. This document reserves the dimension `time` for this type of temporal dependency, which refers to the applicability of the data to the chronological Gregorian calendar.

Frequently, data are additionally temporally dependent relative to some reference time instant: observations may have an accession time into a data repository, numerical weather forecasts may have a nominal time where observations have been assimilated to initialize the calculation, and alerts may have a time when they are issued or published. The diversity of such references precludes defining a dimension type with explicit semantics though the need for a mechanism to distinguish data based on some temporal referent is widely shared. This document therefore defines a generic dimension called `reference_time`, which may be used for any of these situations. If it is used, however, the requirements specify that the semantics must be elucidated in the `<Abstract>` element of the Layer in which the dimension is defined.

The following sub-clauses distinguish all WMS offerings based on their temporal dependency. For data with a temporal dependency that can be mapped to validity time, sub-clause 6.2.1 applies. For data with a temporal dependency that can be mapped to validity time and another temporal dependency that can be mapped to a referential instant in time, sub-clause 6.2.2 applies.

6.2.1 TIME axis

Data providers may wish to offer data that have a temporal dependency that can be mapped to 'validity time,' in which case the dimension TIME SHALL be used in accordance with the following guidelines:

Req. 2 Conformant WMS servers SHALL only offer data that have a temporal dependency that can be mapped to 'validity time' as a WMS layer for which a dimension named `time` is declared or inherited in the Capabilities document.

Req. 3 Conformant WMS clients SHALL use the request parameter TIME only to refer to 'validity time' which refers to the applicability of the data to the chronological Gregorian calendar.

Req. 4 Conformant WMS servers SHALL aggregate multiple data sets differing only in 'validity time' values into a single WMS Layer with a `time` dimension.

Req. 5 Conformant WMS servers SHALL set the `time` dimension `units` value to ISO8601 to indicate time values conformant with the standard ISO8601:2004.

Req. 6 Conformant WMS servers and clients SHALL specify all TIME values in UTC in accordance with the rules of ISO 8601:2004.

Req. 7 Conformant WMS servers SHALL declare in each `time` dimension of the Capabilities document the attribute `default` and assign it the time value that will be used for maps returned in response to requests that do not specify a TIME value as specified in Annex C.2 of the WMS 1.3.0 standard.

The use of a default value for TIME is intended for mass-market clients.

Rec. a Conformant WMS servers SHOULD declare, for any layer that declares a `time` dimension in the Capabilities document, an Abstract property for the same layer, as defined in clause 7.2.4.6 of the WMS 1.3.0 standard. The Abstract property should explain the policy used to choose the default value for the `time` dimension.

Rec. b The default `time` dimension value SHOULD be the value that is closest to the time at which the request was received.

Req. 8 Conformant WMS servers SHALL express the values of the `time` dimension either as one time interval with non-zero resolution when the temporal dependency of the data is time instants or as a comma separated list of timestamps.

The WMS 1.3 specification allows the possibility to define multiple times as several triplets: start/end/interval. The triplets define sets of time instants that may overlap. It is difficult for a human reader to determine the allowed times of the `time` dimension. Hence, this Best Practice document does not recommend the specification of the `time` dimension as a list of triplets, except if the time instants are regularly spaced with a unique interval of time.

Req. 9 Conformant WMS servers SHALL express the values of the `time` dimension as one time interval with zero resolution when the temporal dependency of the data is related to validity periods. Each period has to be part of this time interval.

Example: Consider the warnings shown in Figure 1. If all warnings are represented as a single layer with a `time` dimension, then this dimension should be defined as follows:

```
<Dimension name="time" units="ISO8601" default="2012-08-16"
nearestValue="0" current="0"
>2012-08-10/2012-08-16</Dimension>
```

Req. 10 Conformant WMS servers SHALL include, for all <Dimension ... /> elements declared in the Capabilities document with the name time, a `nearestValue` field with the value set to *0* (ASCII zero), indicating false.

Req. 11 Conformant WMS servers SHALL respond with an `InvalidDimensionValue` exception to a GetMap request that has a TIME request element set to a valid value but that value is not in the domain set of the `time` dimension for any of the layers specified in the request.

Example: Consider two layers, one for temperature observations and one for precipitation observations. The temperature values are measured every 5 minutes while the precipitation values are measured every 15 minutes. The measurement times of the observations are specified in a `time` dimension. The `time` dimension of the temperature layer may be:

```
<Dimension name="time" units="ISO8601"
default="2013-09-24T23:55:00Z" nearestValue="0" current="0"
>2013-09-22T12:00:00Z/2013-09-24T12:00:00Z/PT5M</Dimension>
```

The `time` dimension of the precipitation layer may be:

```
<Dimension name="time" units="ISO8601"
default="2013-09-24" nearestValue="0" current="0"
>2013-09-22T12:00:00Z/2013-09-24T12:00:00Z/PT15M</Dimension>
```

If a user specifies a GetMap request for both layers with the TIME parameter values 2013-09-23T12:00:00Z and 2013-09-23T12:15:00Z, then these timestamps are in the domain of both `time` dimensions. The WMS server shall return a valid map. But if the user specifies the timestamp 2013-09-23T:12:25:00Z as a third TIME parameter value, then this value is not in the domain of the precipitation layer. The WMS server shall respond with an `InvalidDimensionValue` exception.

Rec. c For each invalid dimension value, the exception text SHOULD include the list of layers that do not have this value in their domain sets of the `time` dimension.

Req. 12 Conformant WMS servers SHALL respond with a *`NoMatch`* exception to a GetMap request that includes a TIME value for which a match could not be found for at least one layer. This requirement addresses the case where the time value specified in the request is found in the domain sets of some but not all of the layers in the request.

Example: Consider the warnings shown in Figure 1. If all warnings are represented in a

single layer with a `time` dimension, then this dimension may be defined as follows:

```
<Dimension name="time" units="ISO8601"
default="2012-08-16" nearestValue="0" current="0"
>2012-08-10/2012-08-16</Dimension>
```

If a user specifies a GetMap request for that layer with the TIME parameter values 2012-08-12T00:00:00Z and 2012-08-12T13:00:00Z, then these timestamps are in the domain of the `time` dimension. However, the timestamp 2012-08-12T00:00:00Z will match no warning. The GetMap request shall respond with a `NoMatch` exception.

Rec. d The exception text SHOULD include the list of time values that do not match all requested layers.

Req. 13 Conformant WMS servers SHALL declare in each `time` dimension of the Capabilities document the attribute `current` and assign it the appropriate Boolean value, either `0` (ASCII zero) or `1` (ASCII one) meaning *false* and *true* respectively, as specified in Annex C.2 of the WMS 1.3.0 standard. Conformant WMS servers SHALL set the `current` attribute value to `1` (ASCII one) meaning *true* for any layer produced from data which are continually updated.

Rec. e Conformant WMS clients SHOULD specify a value for the TIME parameter in any GetMap request including a WMS Layer for which a `time` dimension has been defined in the Capabilities document.

This recommendation to include the TIME parameter in every GetMap request rather than relying on the default value is intended to increase precision and is primarily intended for specialized WMS client software. The use of a default value for TIME is intended for mass-market WMS client applications.

Conformant WMS servers can handle GetMap Requests according to the following decision tree.

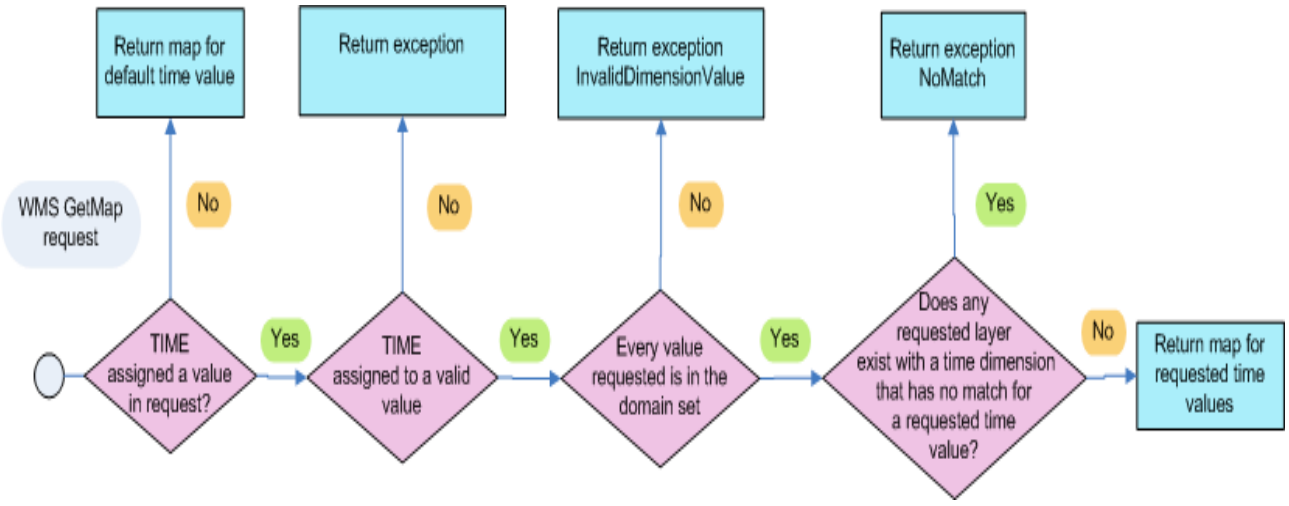

Figure 4: Decision tree for TIME in a GetMap request

6.2.2 REFERENCE_TIME axis

Data providers may wish to offer data that have a second temporal dependency other than the 'validity time'. Examples are the start times of numerical weather calculations or the sending times of warnings. The semantic meaning of a second temporal dependency may differ for different types of data, but the usage pattern has broad application. This Best Practice document recommends the use of a dimension called REFERENCE_TIME if the domain values of this dimension can be restricted to timestamps. Hence, in contrast to the `time` dimension, a value of the REFERENCE_TIME dimension can not be a time interval. Values for this dimension can be specified in client requests using the DIM_REFERENCE_TIME element. As a client request sets a unique value for the DIM_REFERENCE_TIME, a single request cannot be used to request multiple layers that implement different semantics.

The dimension REFERENCE_TIME shall be used in accordance with the following guidelines:

Req. 14 Conformant WMS servers SHALL use the dimension REFERENCE_TIME only for data that have a temporal dependency to 'validity time' and also a temporal dependency to some relevant reference timestamp, such as the beginning of an analysis, a time of data collection, or a time of issuance. Conformant WMS servers SHALL use a REFERENCE_TIME dimension only if a TIME dimension has also been declared in the same or a parent layer.

Req. 15 Conformant WMS clients SHALL use the request parameter REFERENCE_TIME only to refer to a temporal axis other than validity time and in accordance with all other requirements in this sub-clause.

Rec. f Conformant WMS clients SHOULD specify the value of the DIM_REFERENCE_TIME in the GetMap request if the Capabilities document declares a REFERENCE_TIME dimension for the layer in the request.

Req. 16 Conformant WMS servers SHALL use the dimension REFERENCE_TIME only with units declared using the units identifier ”ISO8601” to indicate time values conformant with the standard ISO8601:2004.

Req. 17 Conformant WMS servers and clients SHALL specify all REFERENCE_TIME values in UTC and according to the rules of ISO 8601:2004.

Req. 18 Conformant WMS servers SHALL declare in each REFERENCE_TIME dimension of the Capabilities document the attribute `default` and assign it the value of REFERENCE_TIME that will be used for requests that do not specify a REFERENCE_TIME value.

The use of a default value for REFERENCE_TIME is intended for mass-market clients.

Rec. g Conformant WMS servers SHOULD declare, for any Layer that declares a REFERENCE_TIME dimension in the Capabilities document, an Abstract property for the same Layer, as defined in clause 7.2.4.6 of the WMS 1.3.0 standard. The Abstract property should explain the semantics of the REFERENCE_TIME dimension and the policy used to choose the default value for the REFERENCE_TIME dimension.

Req. 19 Conformant WMS servers SHALL express the values of the REFERENCE_TIME dimension either as a comma separated list of timestamps or as one time interval with non-zero resolution.

Req. 20 Conformant WMS clients SHALL express the value of a REFERENCE_TIME parameter in a GetMap request either as a comma separated list of timestamps or as a time interval with a resolution value of zero.

Req. 21 Conformant WMS servers SHALL include, for all <Dimension ... /> elements declared in the Capabilities document with the name REFERENCE_TIME, a *`nearestValue`* field with the value set to *0* (ASCII zero), indicating false.

Req. 22 Conformant WMS servers SHALL respond with an `InvalidDimensionValue` exception to a GetMap request which has a DIM_REFERENCE_TIME request element set to a valid value but that value is not declared in the Capabilities document as available for any of the layers specified in the request.

Rec. h The exception text SHOULD include the list of layers which do not have this value declared in their domain sets of the REFERENCE_TIME dimension.

Req. 23 Conformant WMS servers SHALL respond with a *`NoMatch`* exception to a GetMap request that includes a DIM_REFERENCE_TIME value for which a match could not be found for at least one of the layers specified in the request. This requirement addresses the case where the DIM_REFERENCE_TIME value specified in the request is found in the domain sets of some but not all of the layers in the request.

Example: Consider a sequence of one month of successive forecasts, some of which are shown in Table 1. All the forecasts may be represented in a single layer with both `time` and `reference-time` dimensions. The `reference-time` dimension may be defined to represent the nominal start times of the forecasts, as follows:

```
<Dimension name="reference_time" units="ISO8601"
 default="2009-11-30T12:00" nearestValue="0" current="0"
>20009-11-01T00:00/2009-11-30T12:00/PT12H</Dimension>
```

If a user specifies a GetMap request for that layer with the REFERENCE_TIME parameter values 2009-11-26T00:00Z, 2009-11-26T12:00Z and 2009-11-27T00:00Z, then these timestamps are in the domain of the time dimension. If however, the forecast for 2009-11-27T00:00Z, with its four validity times, is missing because of operational production problems, the timestamp 2009-11-27T00:00Z will match no available

forecast. The GetMap request shall respond with a `NoMatch` exception.

Req. 24 Conformant WMS servers SHALL declare in each REFERENCE_TIME dimension of the Capabilities document the attribute `current` and assign it the appropriate boolean value, either 0 (ASCII zero) or 1 (ASCII one) meaning false and true respectively, as specified in Annex C.2 of the WMS 1.3.0 standard. Conformant WMS servers SHALL set the `current` attribute value to *1* (ASCII one) meaning true for any layer produced from data that are continually updated.

Conformant WMS servers can handle GetMap Requests according to the decision tree shown in Figure 5. Note that in Figure 5 the REFERENCE_TIME value is considered set in the request only if it is assigned a value in the request.

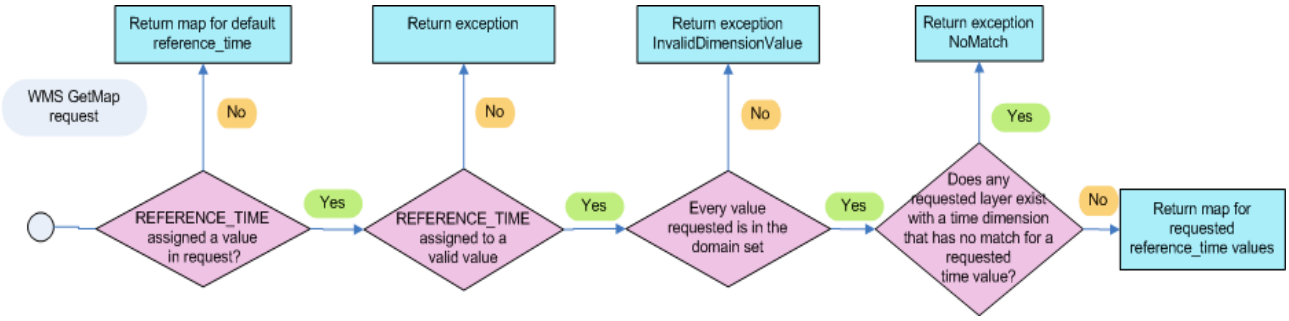

Figure 5: Decision tree for REFERENCE_TIME in a GetMap request

6.3 No vertical dependency

Data providers may wish to offer data that have no vertical dependency, in which case the following requirements apply:

Req. 25 Conformant WMS servers SHALL not offer data as a WMS layer for which a dimension named `elevation` is declared or inherited in the Capabilities document if these data have no vertical dependency.

6.4 Elevation-dependent data

Complex data sets can have elevation dependencies of many kinds. Data might be referenced to the distance above mean sea level, distance below the surface of the sea, depth below the earth surface, distance from the calculated center of the earth, or height in the atmosphere; possibly metric, barometric, or a named surface.

Data sets being offered as elevation dependent WMS layers must define that dependency according to the type of elevation values present. Data without any elevation dependency must not declare or inherit any `elevation` dimension. This section applies to all data sets with elevation-dependent data. Data sets, which are available at one or more elevations specified by numeric values, must follow the injunctions of sub-clause 6.4.1. Data sets that are available at elevations specified for a named surface must follow the injunctions of sub-clause 6.4.2.

In all cases, the dimension ELEVATION must be used in accordance with the following:

Req. 26 Conformant WMS servers SHALL offer any data that have a vertical dependency that can be referenced to numeric or named surface as a WMS Layer with an `elevation` dimension.

Req. 27 Conformant WMS servers and clients SHALL use the dimension ELEVATION only to refer to a vertical axis having numeric values or named surfaces.

Req. 28 Conformant WMS servers SHALL NOT declare in the Capabilities document a WMS Layer with an `elevation` dimension that includes both numeric and named surfaces.

Req. 29 Conformant WMS servers SHALL aggregate multiple data sets differing only in elevation values into a single WMS layer with an `elevation` dimension. This will avoid multiple independent layers related by a common coordinate reference system (CRS).

Req. 30 Conformant WMS servers SHALL declare in each `elevation` dimension of the Capabilities document the attribute *`multipleValues`* and SHALL assign it the boolean value *0* (ASCII zero) indicating *false* as specified in Annex C.2 of the WMS 1.3.0 standard.

Req. 31 Conformant WMS servers SHALL declare in each `elevation` dimension of the Capabilities document the attribute *`nearestValue`* and assign it the boolean value *0* (ASCII zero) indicating *false* as specified in Annex C.2 of the WMS 1.3.0 standard.

The following sub-clauses distinguish all WMS offerings based on their vertical dependency. For data with a vertical dependency referenced to numeric values, sub-clause 6.4.1 applies. For data with a vertical dependency referenced to named surfaces, sub-clause 6.4.2 applies.

6.4.1 ELEVATION axis using numeric values

Data providers may wish to offer data that have a vertical dependency that can be referenced to numeric values, in which case the following requirements apply:

Req. 32 Conformant WMS servers SHALL declare in each `elevation` dimension of the Capabilities document the attribute UNITS and assign it a CRS identifier specifying the vertical CRS. Elevation metric values SHALL be expressed in units of the CRS.

Req. 33 Conformant WMS servers SHALL declare in each `elevation` dimension of the Capabilities document the attribute UNITSYMBOL and SHALL assign it the string representation of the symbol for the units.

Req. 34 Conformant WMS servers SHALL declare in each `elevation` dimension of the Capabilities document the attribute *`default`* and SHALL assign it the ELEVATION

value that will be used for requests that do not specify an elevation value.

Rec. i Conformant WMS clients SHOULD specify a value for the ELEVATION parameter in any GetMap request including a WMS Layer for which an `elevation` dimension has been defined in the Capabilities document.

Rec. j Conformant WMS clients SHOULD specify in GetMap requests only metric values declared in the `elevation` dimension. Where the metric value declared in the Capabilities document represents a layer of non-zero vertical thickness, the GetMap requests SHOULD specify ELEVATION using the pairs of values exactly as stated in the list of available elevations.

Req. 35 Conformant WMS servers SHALL declare in each `elevation` dimension of the Capabilities document the attribute *current* and assign it the Boolean value *0* (ASCII zero) indicating false as specified in Annex C.2 of the WMS 1.3.0 standard. This means that the vertical axis SHALL be static, i.e. the domain set values of the axis are not continually updated.

Rec. k Conformant WMS servers offering any data that have a vertical dependency that can be referenced to metric values SHOULD sort lists of elevation metric values in order from nearest the surface of the Earth (or ocean) to farthest from the surface. In other words, lower elevations should come before higher elevations, and shallower depths should come before greater depths.

Req. 36 Conformant WMS servers SHALL express the values of the `elevation` dimension either as a comma separated list of numeric values or as one interval with non-zero resolution if the vertical dependency of the data is referenced by discrete values.

Req. 37 Conformant WMS servers SHALL express the values of the `elevation` dimension as an interval with zero resolution, if the vertical dependency contains ranges, i.e. at least one layer element is a range. All ranges have to be part of this interval.

Example: Consider a data set recording wind speed of several atmospheric surfaces. Each surface or 'thick slice' is defined by the space between two isobaric levels (e.g. 1000-900 hPa). All surfaces are between 1000 hPa and 100 hPa. The data set should be represented as a single layer with an `elevation` dimension. This dimension may be defined as follows:

```
<Dimension name="elevation" units="someCRS:xxxx" unitSymbol="hPa"
default="1000" multipleValues="0" nearestValue="0"
current="0">100/1000</Dimension>
```

Req. 38 Conformant WMS servers SHALL respond with an *`InvalidDimensionValue`* exception to a GetMap request that has an ELEVATION request element set to a valid value but that value is not in the domain set of the `elevation` dimension for any of the layers

specified in the request

Rec. l For each invalid dimension value, the exception text SHOULD include the list of layers that do not have this value in their domain sets of the `elevation` dimension.

Req. 39 Conformant WMS servers SHALL respond with a *`NoMatch`* exception to a GetMap request that includes an ELEVATION value for which a match could not be found for at least one layer. This requirement addresses the case where the elevation value specified in the request is found in the domain sets of some but not all of the layers in the request.

Example: Consider a data set recording wind speed of several atmospheric surfaces. Each surface or 'thick slice' is defined by the space between two isobaric levels (e.g. 1000-900 hPa). All surfaces are between 1000 hPa and 100 hPa. The data set should be represented as a single layer with an `elevation` dimension. This dimension may be defined as follows:

```
<Dimension name="elevation" units="someCRS:xxxx" unitSymbol="hPa"
default="1000" multipleValues="0" nearestValue="0"
current="0">100/1000</Dimension>
```

Assume that the surface or 'thick slice' defined by the isobaric levels 700-600 hPa is missing. If a user specifies a GetMap request for the layer with the elevation parameter values 550 and 650, then these values are in the domain of the `elevation` dimension. However, the value 650 will match no surface. The GetMap request shall respond with a *`NoMatch`* exception.

6.4.2 ELEVATION axis using named surfaces

Data providers may wish to offer data that have a vertical dependency that can be referenced to named surfaces, in which case the following requirements apply:

Req. 40 Conformant WMS servers SHALL declare in the `elevation` dimension the attribute UNITS and SHALL assign it the value *`computed_surface`*, `UNITS="computed_surface"`.

Req. 41 Conformant WMS servers SHALL declare in the `elevation` dimension the attribute UNITSYMBOL and SHALL assign it the value of the null string, `UNITSYMBOL=""`.

Req. 42 Conformant WMS servers SHALL declare in the `elevation` dimension the attribute *`default`* and SHALL assign it the named surface that will be used for requests that do not specify an elevation value.

Req. 43 Conformant WMS servers **SHALL** assign to the values of the `elevation` dimension the named surface or named surfaces of the vertical dependency.

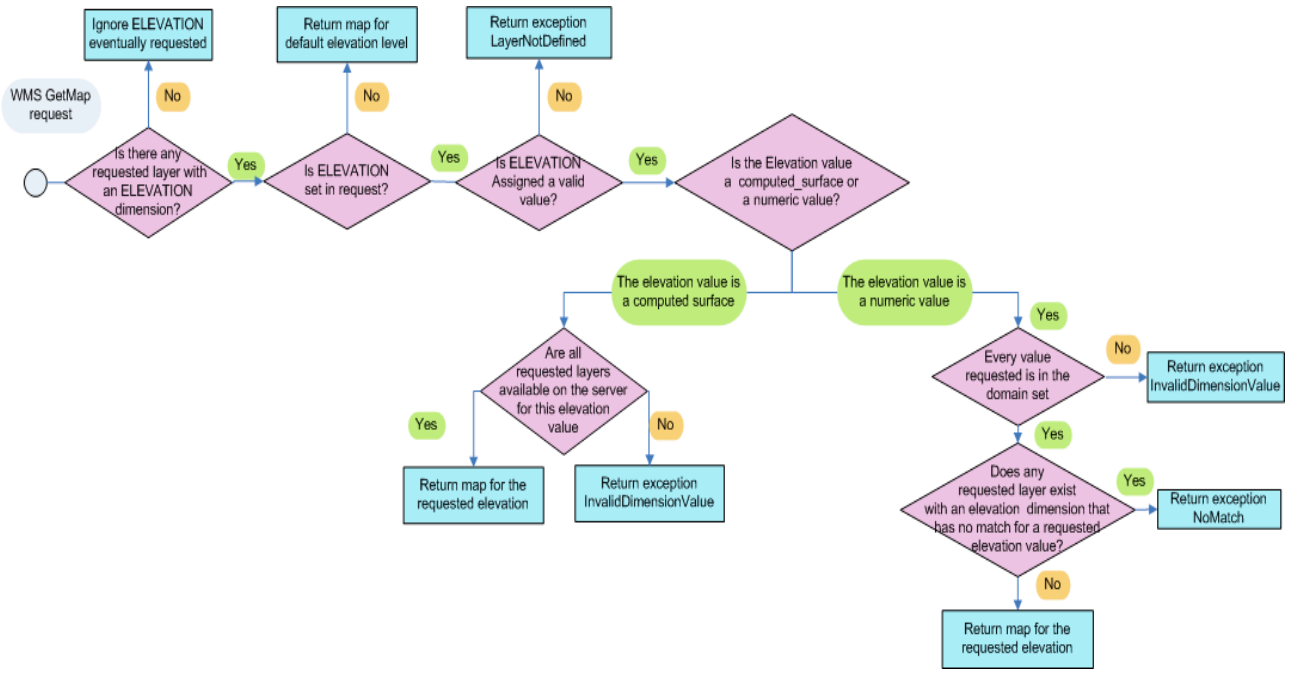

Figure 6: Decision tree for ELEVATION in a GetMap request

Annex A: Dimensions and Coordinates

In physics and mathematics the dimension of a space or object is informally defined as the minimum number of coordinates needed to specify any point within it (see http://en.wikipedia.org/wiki/Dimension). Thus a line has a dimension of one because only one coordinate is needed to specify the location of a point on it. A map has two dimensions because two coordinates are needed to locate a point on it. These coordinates are usually latitude and longitude.

A dimension is described by a domain. The domain is a set of values having all the same type, e.g. number, string or date. A coordinate of the dimension D is a value in D's domain. For an example, consider a `time` dimension with a domain that contains all date values between January 1st 2010 and June 16th 2013. March 10th 2012 will be a valid coordinate in this dimension. Sometimes, a dimension is associated with a unit, e.g. meter or hectoPascal (hPa). This may allow the transformation of the domain into another domain with a different unit if a transformation formula exists.

A WMS is a standardized interface to visualize two-dimensional maps. However, the data behind a map, the so-called layer, may have more than two dimensions.

The WMS specification allows the definition of multi-dimensional layers. Then, each layer object is explicitly assigned a pair of coordinates for every dimension. This pair defines an interval on the dimension. The coordinate pairs for latitude and longitude specify the bounding box of the layer object. The bounding box is assigned to the layer rather than to the layer objects because the box has to be identical for all layer objects.

Let $[x_i,y_i]$ be the coordinate interval of a layer object L in the dimension i. A coordinate interval $[a_i,b_i]$ in dimension i matches L if and only if $[a_i,b_i]$ overlaps $[x_i,y_i]$, i.e. $x_i \leq b_i$ and $a_i \leq y_i$. Note that this statement can be simplified if $a_i = b_i$. I.e. The coordinate interval reduces to a single coordinate. A coordinate c_i in dimension i matches L if and only if $x_i \leq c_i \leq y_i$.

So, if a request is made to a WMS for a map on a coordinate c_i in dimension i, then the WMS server should produce this map from all layer objects which are matched by c_i.

Consider for example a layer with warnings. Each warning is for a geographic area and is associated with a time, when the warning becomes effective, and a time when the warning expires. Thus, this layer has a `time` dimension. If a user wants to get a map that shows all warnings that are effective today at 20:00 UTC, then a GetMap request with this time coordinate has to be submitted to the WMS server. The server selects all warnings that are matched by the time coordinate and produces a map that visualizes the geographic areas of these warnings.

Another example is the distribution of ozone concentration. Such a layer may be composed of four coverages. The first coverage contains the total sum of ozone concentration for the elevation between 0m and 10m. The second coverage contains the total sum of ozone concentration for the elevation between 10m and 100m. The third resp. forth coverage contains the total sum of ozone concentration for the elevation between 100m and 500m resp. 500m and 2000m. Thus, this layer has a third dimension, the elevation. The domain of this dimension consists of numbers. The unit is meter. If this is a request for the ozone concentration between 0m and 50m at a geographic point, then a

corresponding GetFeatureInfo request is sent to the WMS server. The interval [0,50] matches the first and the second coverage. Hence, the WMS server should return the value at the given point for both coverages.

The WMS specification provides the optional <Dimension> element for a declaration of additional dimensions. The mandatory fields of a <Dimension> element are the name, the units and the extent. The extent should indicate the available values of the corresponding dimension. The units element denotes the domain type of a dimension and the extent describes the values of the dimension's domain. For example, latitude and longitude can be declared as <Dimension> elements.

```
<Dimension name="latitude" units="degree" unitSymbol="°"
 default="0">-90/90/0</Dimension>
<Dimension name="longitude" units="degree" unitSymbol="°"
 default="0">-180/180</Dimension>
```

The extent of latitude, or longitude, is a range because every value in the range will be a valid coordinate of this dimension.

Each dimension of a layer, other than latitude and longitude, has to be declared in the Capabilities document.

The `time` dimension of the previous warning layer example may be declared as follows:

```
<Dimension name="time" units="ISO8601" default="2013-06-27T12:00:00Z"
>2013-06-26/2013-06-30</Dimension>
```

The extent is a time interval and every timestamp in this interval is considered to be a valid time coordinate for this `time` dimension.

If the upper and the lower bounds are equal for every coordinate interval of a dimension, i.e. each coordinate interval is in fact a single coordinate, then the extent of this dimension should be a list of these coordinates and not an interval, because only these coordinates will match with an object of the layer.

Annex B: GetFeatureInfo

GetFeatureInfo is an optional operation. This operation is designed to provide clients of a WMS with more information about features in the pictures of maps that were returned by previous GetMap requests. It is only supported for those layers for which the attribute queryable="1" (true) has been defined or inherited.

The canonical use case for GetFeatureInfo is that a user sees the response of a GetMap request and chooses a point (I,J) on that map for which to obtain more information. The basic operation provides the ability for a client to specify which pixel is being queried, which layer(s) should be investigated, and in what format the information should be returned. In the case where the requested layers have additional dimensions for time or elevation the GetFeatureInfo operation can be used to ask for time series or other series at the chosen point.

This chapter defines additional requirements and recommendations for the use of the GetFeatureInfo operation. Further, those requirements from Section 6, which apply not only to a GetMap request but also to a GetFeatureInfo request, are listed.

The WMS 1.3.0 specification defines the following two requirements for GetFeatureInfo:

WMS 1.3 requirement	Conformant WMS clients SHALL not issue a GetFeatureInfo request for layers for which the attribute queryable="1" (true) has not been defined or inherited.
WMS 1.3 requirement	Conformant WMS servers SHALL answer with an *`OperationNotSupported`* exception to a GetFeatureInfo request if the server will not support this operation.

For this Best Practice document we define further requirements. These are:

Req. 44 Conformant WMS servers SHALL set the attribute *`queryable`* to "*1*", indicating true, for all layers declared in the Capabilities document which have any dimension involving time or elevation.

Req. 45 If a time dimension has been set for a layer, conformant queryable WMS servers SHALL return a feature collection to a GetFeatureInfo request where each feature includes the element "time" with a timestamp associated to the feature.

Req. 46 If a reference time dimension has been set for a layer, conformant WMS servers SHALL return a feature collection to a GetFeatureInfo request where each feature includes the element "`reference_time`" with a timestamp associated to the feature.

Req. 47 If an `elevation` dimension has been set for a layer, conformant WMS servers SHALL return a feature collection to a GetFeatureInfo request where each feature includes the

element “`elevation`” with an according elevation value.

The following requirements and recommendations which apply to a GetMap request also apply to a GetFeatureInfo request: **Reqs: 11, 12**, **20**, **22**, **23**, **38**, **39** and **Recs: e, f, j, k**. These are copied below for convenience.

Req. 11 Conformant WMS servers SHALL respond with an *InvalidDimensionValue* exception to a GetMap request that has a TIME request element set to a valid value but that value is not in the domain set of the time dimension for any of the layers specified in the request.

Example: Consider two layers, one for temperature observations and one for precipitation observations. The temperature values are measured every 5 minutes while the precipitation values are measured every 15 minutes. The measurement times of the observations are specified in a time dimension. The time dimension of the temperature layer may be:

```
<Dimension name="time" units="ISO8601"
default="2013-09-24T23:55:00Z" nearestValue="0" current="0"
>2013-09-22T12:00:00Z/2013-09-24T12:00:00Z/PT5M</Dimension>
```

The time dimension of the precipitation layer may be:

```
<Dimension name="time" units="ISO8601"
default="2013-09-24" nearestValue="0" current="0"
>2013-09-22T12:00:00Z/2013-09-24T12:00:00Z/PT15M</Dimension>
```

If a user specifies a GetMap request for both layers with the TIME parameter values 2013-09-23T12:00:00Z and 2013-09-23T12:15:00Z, then these timestamps are in the domain of both time dimensions. The WMS server shall return a valid map. But if the user specifies the timestamp 2013-09-23T:12:25:00Z as a third TIME parameter value, then this value is not in the domain of the precipitation layer. The WMS server shall respond with an *InvalidDimensionValue* exception.

Req. 12 Conformant WMS servers SHALL respond with a *NoMatch* exception to a GetMap request that includes a TIME value for which a match could not be found for at least one layer. This requirement addresses the case where the time value specified in the request is found in the domain sets of some but not all of the layers in the request.

Example: Consider the warnings shown in Figure 1. If all warnings are represented in a single layer with a time dimension, then this dimension may be defined as follows:

```
<Dimension name="time" units="ISO8601"
default="2012-08-16" nearestValue="0" current="0"
>2012-08-10/2012-08-16</Dimension>
```

If a user specifies a GetMap request for that layer with the TIME parameter values 2012-08-12T00:00:00Z and 2012-08-12T13:00:00Z, then these timestamps are in the domain of the time dimension. However, the timestamp 2012-08-12T00:00:00Z will match no warning. The GetMap request shall respond with a *NoMatch* exception.

Rec. e Conformant WMS clients SHOULD specify a value for the TIME parameter in any GetMap request including a WMS Layer for which a time dimension has been defined in the Capabilities document.

This recommendation to include the TIME parameter in every GetMap request rather than relying on the default value is intended to increase precision and is primarily intended for specialized WMS client software. The use of a default value for TIME is intended for mass-market WMS client applications.

Rec. f Conformant WMS clients SHOULD specify the value of the DIM_REFERENCE_TIME in the GetMap request if the Capabilities document declares a REFERENCE_TIME dimension for the layer in the request.

Req. 20 Conformant WMS clients SHALL express the value of a REFERENCE_TIME parameter in a GetMap request either as a comma separated list of timestamps or as a time interval with a resolution value of zero.

Req. 22 Conformant WMS servers SHALL respond with an InvalidDimensionValue exception to a GetMap request which has a DIM_REFERENCE_TIME request element set to a valid value but that value is not declared in the Capabilities document as available for any of the layers specified in the request.

Req. 23 Conformant WMS servers SHALL respond with a *NoMatch* exception to a GetMap request that includes a DIM_REFERENCE_TIME value for which a match could not be found for at least one of the layers specified in the request. This requirement addresses the case where the DIM_REFERENCE_TIME value specified in the request is found in the domain sets of some but not all of the layers in the request.

Example: Consider a sequence of one month of successive forecasts, some of which are shown in Table 1. All the forecasts may be represented in a single layer with both `time` and `reference-time` dimensions. The `reference-time` dimension may be defined to represent the nominal start times of the forecasts, as follows:

```
<Dimension name="reference_time" units="ISO8601"
 default="2009-11-30T12:00" nearestValue="0" current="0"
>20009-11-01T00:00/2009-11-30T12:00/PT12H</Dimension>
```

If a user specifies a GetMap request for that layer with the REFERENCE_TIME parameter values 2009-11-26T00:00Z, 2009-11-26T12:00Z and 2009-11-27T00:00Z, then these timestamps are in the domain of the time dimension. If however, the forecast for 2009-11-27T00:00Z, with its four validity times, is missing because of operational production problems, the timestamp 2009-11-27T00:00Z will not match any available forecast. The GetMap request shall respond with a *NoMatch* exception.

Rec. j Conformant WMS clients SHOULD specify in GetMap requests only metric values declared in the elevation dimension. Where the metric value declared in the Capabilities document represents a layer of non-zero vertical thickness, the GetMap requests SHOULD specify ELEVATION using the pairs of values exactly as stated in the list of available elevations.

Rec. k Conformant WMS servers offering any data that have a vertical dependency that can be referenced to metric values SHOULD sort lists of elevation metric values in order from nearest the surface of the Earth (or ocean) to farthest from the surface. In other words,

lower elevations should come before higher elevations, and shallower depths should come before greater depths.

Req. 38 Conformant WMS servers SHALL respond with an InvalidDimensionValue exception to a GetMap request that has an ELEVATION request element set to a valid value but that value is not in the domain set of the elevation dimension for any of the layers specified in the request

Req. 39 Conformant WMS servers SHALL respond with a `NoMatch` exception to a GetMap request that includes an ELEVATION value for which a match could not be found for at least one layer. This requirement addresses the case where the elevation value specified in the request is found in the domain sets of some but not all of the layers in the request.

Example: Consider a data set recording wind speed of several atmospheric surfaces. Each surface or 'thick slice' is defined by the space between two isobaric levels (e.g. 1000-900 hPa). All surfaces are between 1000 hPa and 100 hPa. The data set should be represented as a single layer with an elevation dimension. This dimension may be defined as follows:

```
<Dimension name="elevation" units="someCRS:xxxx" unitSymbol="hPa"
default="1000" multipleValues="0" nearestValue="0"
current="0">100/1000</Dimension>
```

Assume that the surface or 'thick slice' defined by the isobaric levels 700-600 hPa is missing. If a user specifies a GetMap request for the layer with the elevation parameter values 550 and 650, then these values are in the domain of the elevation dimension. However, the value 650 will match no surface. The GetMap request shall respond with a `NoMatch` exception.

Annex C: Summary of Best Practice Impacts on WMS1.3 Implementations

Requirements fall into four categories:

C.1 How data must be mapped into WMS layers

Temporal			**Elevation**		
Atemporal	**Having a 'validity time' concept**	**Having other time concept than 'validity_time'**	**Not vertical dependency**	**numeric**	**named surface**
	Must use '`time`'	Must use Reference_time		Must use 'elevation'	Must use 'elevation'

C.2 How domain clients must issue GetMap Requests (Table 8 from WMS1.3 specification, **changes indicated in Bold**)

Table 8 — The Parameters of a GetMap request of the specification is changed subsequently

Request parameter	**Mandatory /optional**	**Description**
VERSION=1.3.0	M	Request version.
REQUEST=GetMap	M	Request name.
LAYERS=layer_list	M	Comma-separated list of one or more map layers.
STYLES=style_list	M	Comma-separated list of one rendering style per requested layer.
CRS=namespace:identifier	M	Coordinate reference system.
BBOX=minx,miny,maxx,maxy	M	Bounding box corners (lower left, upper right) in CRS units.
WIDTH=output_width	M	Width in pixels of map picture.
HEIGHT=output_height	M	Height in pixels of map picture.
FORMAT=output_format	M	Output format of map.
TRANSPARENT=TRUE|FALSE	O	Background transparency of map (default=FALSE).
BGCOLOR=color_value	O	Hexadecimal red-green-blue colour value for the background colour (default=0xFFFFFF).
EXCEPTIONS=exception_format	O	The format in which exceptions are to be reported by the WMS (default=XML).
TIME=time	**M**	**Validity Time value of layer desired.**
REFERENCE_TIME	**O**	**Second temporal dimension. Semantics can vary**
ELEVATION=elevation	O	Elevation of layer desired.
Other sample dimension(s)	O	Value of other dimensions as appropriate.

C.3 How the WMS layers must be declared in the capabilities

Contents of the dimension elements of this Best Practice document

	TIME		**REFERENCE_TIME**		**ELEVATION**	
Field	**Mandatory/ optional**	**Meaning**	**Mandatory/optional**	**Meaning**	**Mandatory/ optional**	**Meaning**
name	**TIME**	Name of dimensional axis.	**REFERENCE_TIME**	Name of dimensional axis.	**ELEVATION**	Name of dimensional axis.
units	ISO8601	Attribute indicating units of dimensional axis.	ISO8601	Attribute indicating units of dimensional axis.	M	Attribute indicating units of dimensional axis.
unitSymbol	“”	Attribute specifying symbol.	“”	Attribute specifying symbol.	M	Attribute specifying symbol.
default	**M**	**Attribute indicating default value that will be used if GetMap request does not specify a value. The default value should be the value that is closest to now and in any case documented into the Abstract property**	**M**	**Attribute indicating default value that will be used if GetMap request does not specify a value. The default value should be the value that is closest to now and in any case documented into the Abstract property**	**M**	**Attribute indicating default value that will be used if GetMap request does not specify a value.**
multipleValues	O	Boolean attribute indicating whether multiple values of the dimension may be requested. 0 (or “false”) = single values only; 1 (or “true”) = multiple values permitted. Default = 0.	O	Boolean attribute indicating whether multiple values of the dimension may be requested. 0 (or “false”) = single values only; 1 (or “true”) = multiple values permitted. Default = 0.	**O**	**Boolean attribute indicating whether multiple values of the dimension may be requested. If a value is assigned to this attribute , it has to be set tto 0 (or “false”) = single values only; Default = 0.**
nearestValue	**M**	**Boolean attribute indicating whether nearest value of the dimension will be returned in response to a request for a nearby value. Has to be set to 0 (or “false”) = request value(s) must correspond exactly to declared extent value(s);**	**M**	**Boolean attribute indicating whether nearest value of the dimension will be returned in response to a request for a nearby value. Has to be set to 0 (or “false”) = request value(s) must correspond exactly to declared extent value(s);**	**M**	**Boolean attribute indicating whether nearest value of the dimension will be returned in response to a request for a nearby value. Has to be set to 0 (or “false”) = request value(s) must correspond exactly to declared extent value(s);**

current	**M**	**Boolean attribute valid only for temporal extents (i.e. if attribute name="time"). This attribute, if it either 1 or "true", indicates (a) that temporal data are normally kept current and (b) that the request parameter TIME may include the keyword "*current*" instead of an ending value (see C.4.1). Current =1 if data are continually updated Default= 0**	**M**	**Boolean attribute valid only for temporal extents (i.e. if attribute name="time"). This attribute, if it either 1 or "true", indicates (a) that temporal data are normally kept current and (b) that the request parameter TIME may include the keyword "*current*" instead of an ending value (see C.4.1). Current =1 if data are continually updated Default= 0**	**O**	**Boolean attribute valid only for temporal extents (i.e. if attribute name="time"). If it is assigned, this attribute, has to be set to 0 or "false", meaning that the vertical axis SHALL be static, i.e. the domain set values of the axis are not continually updated. Default = 0.**
extent	O	Text content indicating available value(s) for dimension.	O	Text content indicating available value(s) for dimension.	O	Text content indicating available value(s) for dimension.

C.4 Requirements that target WMS Layer declaration (I.E <Dimension> or <Abstract> properties) and GetMap requests content

	time	**reference_time**	**elevation**	
Dimension			**Numeric**	**Named surface**
name=	time Semantics = 'validity_time' □ Servers: Req 2 □ Clients: Req 3	reference_time Semantics can vary □ Servers: Req 14 □ Clients: Req 15	elevation □ Servers: Req 25 □ Servers: Req 26	
units=	ISO8601 □ Servers: Req 5	ISO8601 □ Servers: Req 16	Requested □ Servers: Req 31	Requested □ Servers: Req 39
unitSymbol=	ISO8601 □ Servers: Req 6	ISO8601 □ Servers: Req 17	Requested □ Servers: Req 32	Requested □ Servers: Req 40
default=	Required □ Servers: Req 7 Documented □ Servers: Rec a Closest to now □ Servers: Rec b	Required □ Servers: Req 18 Documented □ Servers: Rec g	Required □ Servers: Req 34	Required □ Servers: Req 42
multipleValues=			Required and set to false □ Servers: Req 30	
nearestValue=	Required and set to false □ Servers: Req 10	Required and set to false □ Servers: Req 21	Required and set to false □ Servers: Req 31	
current=	Required and set to 1 (true) □ Servers: Req 13	Required and set to 1 (true) □ Servers: Req 24	Required and set to 0 (false) □ Servers: Req 35	
Values	interval with non zero resolution or list □ Servers: Req 8 □ Servers: Req 9	interval with non zero resolution or list □ Servers: Req 19 □ Clients: Req 20	□ Servers: Rec l □ Servers: Req 36 □ Servers: Req 37	□ Servers: Req 43
Abstract	Explain default □ Servers: Rec a	Explain semantics and default □ Servers: Rec g		