

Open Geospatial Consortium Inc.

Date: 2006-12-21

Reference number of this OGC® document: **OGC 06-154**

Version: 0.1.0

Category: OpenGIS® Interoperability Program Report

Editors: David S. Burggraf, Ron Lake, Darko Androsevic

OWS 4 WFS Temporal Investigation

Copyright notice

Copyright © 2007 Open Geospatial Consortium, Inc. All Rights Reserved.
To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>.

Warning

This document is not an OGC Standard. This document is an OGC Discussion Paper and is therefore not an official position of the OGC membership. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard. Further, an OGC Discussion Paper should not be referenced as required or mandatory technology in procurements.

Document type: OGC® Interoperability Program Report
Document stage: Draft
Document language: English

Contents

Preface	v
i. Submitting organizations	v
ii. Document contributor contact points	v
iii. Revision history	v
iv. Changes to the OpenGIS® Specifications	vi
Foreword	vii
Introduction	viii
1 Scope	1
2 Conformance	1
3 Normative references	1
4 Terms and definitions	1
5 Conventions	2
5.1 Symbols (and abbreviated terms)	2
5.2 UML notation	3
6 Temporal Investigation	4
6.1 Temporal Support in GML v3.2	4
6.1.1 Temporal Types	4
6.1.2 Dynamic Features in GML	6
6.2 Comparison Operators and Temporal Queries	9
6.3 Review of OGC 06-076 Temporal Features	25
7 Temporal Query Experiments	28
7.1 Cartalinea on Oracle	28
7.1.1 Test Insert – TimePeriod using gml:begin/gml:end	28
7.1.2 Test Insert - TimePeriod2 – using gml:beginPosition/gml:endPosition	29
7.1.3 Test Insert - TimePeriod3 - beginPosition/endPosition/@indeterminatePosition	30
7.1.4 Test Get - Property filter – TM_During	31
7.1.5 Test Update - Temporal property	32
7.1.6 TM_After	33
7.1.7 TM_Before	34
7.1.8 TM_Overlaps	37
7.1.9 TM_Contains	38
8 Discussion	40
8.1 Application Semantics vs Physical Markup	40

8.2	Clarity of Expression - Complexity	41
8.3	Mapping to Underlying Database Types	41
8.4	Handling Time Zones in a Distributed Environment.....	42

Preface

Suggested additions, changes, and comments on this draft report are welcome and encouraged. Such suggestions may be submitted by OGC portal message, email message, or by making suggested changes in an edited copy of this document.

The changes made in this document version, relative to the previous version, are tracked by Microsoft Word, and can be viewed if desired. If you choose to submit suggested changes by editing this document, please first accept all the current changes, and then make your suggested changes with change tracking on.

i. Submitting organizations

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

- Galdos Systems Inc.

ii. Document contributor contact points

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

Contact	Company
D. Burggraf, PhD	Galdos Systems Inc.
Darko Androsevic	Galdos Systems Inc.
Ron Lake	Galdos Systems Inc.

iii. Revision history

Date	Release	Editor	Primary clauses modified	Description
2006-31-12	0.0.1	D. Burggraf	All	First draft

iv. Changes to the OpenGIS[®] Specifications

No changes to OGC Specifications are required by this document.

Foreword

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

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

Introduction

Many if not all geographic problems involve time. As a consequence, the ability to query a geographic database via WFS requests involving temporal and spatial filters is highly desirable. In fact, proposals for such operators have been proposed via change requests for Version 1.2 of the OGC Web Feature Service (WFS) Specification. Furthermore, Geography Markup Language (GML), on which WFS is based contains several key components in support of temporal geography, including:

- Temporal data types including in particular:
 - a. TimeInstant
 - b. TimePeriod

Additional attributes with these types allow the expression of time formats (e.g. default for timePosition is ISO 8601), user defined calendars and clocks, and ordinal (indeterminate) time expressions (e.g. before, after etc.).

- Dynamic Features. GML Dynamic Features support two temporal models, namely 1) a Dynamic Feature in GML has a history consisting of sequence of user defined TimeSlices which contain the time varying properties of the feature, and 2) a snap shot model where a validTime property is attached to a feature, thus providing a snap shot of the feature's properties at that moment in time. Dynamic Features can be applied to a wide class of geo-temporal problems such as:
 - a. Current and historical tracking of moving objects (e.g. vehicles, ships, aircraft).
 - b. Archeological digs and sites (e.g. culture succession, stages of building).
 - c. Evolution of the built environment – e.g. buildings, roadways, and related physical infrastructure.
 - d. Disaster impact areas from floods, fires etc.

The objective of the proposed temporal extensions to the WFS is to enable temporal/geospatial queries using the GML temporal types against GML dynamic features employing either the snapshot or time history model (time slices). One might anticipate that such functionality would depend on the base capabilities of the underlying data store (Although the objective of the WFS is to hide such capabilities or capability distinctions as much as possible), and that these base store capabilities would impact the difficulty or ease of implementation of temporal query support on a particular database platform.

In addition to these considerations, we note that WFS version 1.1 (and 1.0) provide support for queries that employ arithmetic tests on feature properties (via comparison operators), and that such properties could, in principle, be time valued. Use of these arithmetic tests might then provide an alternative in some cases to the implementation and support of temporal specific operators.

To investigate these questions, Galdos Systems conducted a series of experiments using the Oracle (10G) version of our Cartalinea Web Feature Service (WFS).

OWS 4 WFS Temporal Investigations

1 Scope

All of the work described in this document was conducted by Galdos Systems Inc. This work is comprised of the following parts:

- Review of temporal and dynamic feature constructs in GML v3.1.1/v3.2.
- Review of proposed temporal operators for WFS.
- Review of proposed temporal feature constructs for GML and WFS.
- Experiments with comparison operators to “simulate” proposed WFS temporal operators using ISO 19108:2003 semantics.

2 Conformance

Not required for an IP IPR, DIPR, or Discussion Paper.

3 Normative references

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

ISO DIS 19108, *Geographic Information – Temporal Schema*

OGC 03-105r1, OpenGIS® Geography Markup Language (GML) Implementation Specification, Version 3.1.1, April 2004.

4 Terms and definitions

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

4.1 Time stamp

The time that some data change is recorded in a database. Also known as database time.

4.2 Time zone

A region of the earth in which the time is uniform relative to UTC. The earth is divided into 24 time zones mostly by lines of longitude.

4.3 UTC

The acronym for “Coordinate Universal Time”. UTC is an atomic clock based time standard. With respect to time zones it is approximately the same as Greenwich Mean Time (GMT)

4.4 valid time

The time that some fact is true in the real world.

5 Conventions

5.1 Symbols (and abbreviated terms)

API	Application Program Interface
COM	Component Object Model
CORBA	Common Object Request Broker Architecture
COTS	Commercial Off The Shelf
DCE	Distributed Computing Environment
DCP	Distributed Computing Platform
DCOM	Distributed Component Object Model
GML	Geography Markup Language
ISO	International Organization for Standardization
OGC	Open GIS Consortium
UML	Unified Modeling Language
WFS	Web Feature Service
XML	eXtensible Markup Language
1D	One Dimensional

2D Two Dimensional

3D Three Dimensional

5.2 UML notation

Most diagrams that appear in this specification are presented using the Unified Modeling Language (UML) static structure diagram, as described in Subclause 5.2 of the OGC Web Services Common Implementation Specification [OGC 04-016r2].

6 Temporal Investigation

6.1 Temporal Support in GML v3.2

GML provides a variety of components for the support of time and time dependent geographic phenomena. This section provides a brief summary based on the GML v3.2 specification. Since current generations of WFS (WFS 1.1) are based on GML v3.1.1, differences between the two specifications (v3.2 and v3.1.1) respecting temporal issues are highlighted where relevant.

6.1.1 Temporal Types

GML v3.2 provides the separate schema `temporal.xsd`, that defines a set of elements, content models and attributes for the description of time. This section provides a brief overview of concepts that are of importance for temporal queries to a WFS. For a more detailed description see the GML v3.2 (reference) and GML v3.1 (reference) specifications.

The key temporal elements and content models of interest to the current discussion are as follows:

- `TimeInstant`
- `TimePeriod`
- `timePosition`
- `timeLength`
- `duration`
- `timeInterval`

TimeInstant

`gml:TimeInstant` is a zero-dimensional geometric primitive that represents an identifiable position in time.

EXAMPLE In an instance document, a `gml:TimeInstant` contains a `gml:timePosition` as follows:

```
<gml:TimeInstant gml:id="t11">
  <gml:description>Fire outbreak</gml:description>
  <gml:timePosition>2006-08-13</gml:timePosition>
</gml:TimeInstant>
```

TimePeriod

`gml:TimePeriod` is a one-dimensional geometric primitive that represents an identifiable extent in time.

The location in of a `gml:TimePeriod` is described by the temporal positions of the time instants at which it begins and ends. The length of the period is equal to the temporal distance between the two bounding temporal positions.

Both beginning and end may be described in terms of their direct position using `gml:TimePositionType` which is an XML Schema simple content type, or by reference to an identifiable time instant using `gml:TimeInstantPropertyType`

EXAMPLE 1 Within a `gml:TimePeriod`, a `gml:TimeInstant` may appear directly as the value of the begin and end as follows:

```
<gml:TimePeriod gml:id="p22">
  <gml:begin>
    <gml:TimeInstant gml:id="t11">
      <gml:timePosition>2001-05-23</gml:timePosition>
    </gml:TimeInstant>
  </gml:begin>
  <gml:end>
    <gml:TimeInstant gml:id="t12">
      <gml:timePosition>2001-06-23</gml:timePosition>
    </gml:TimeInstant>
  </gml:end>
</gml:TimePeriod>
```

Alternatively a limit of a `gml:TimePeriod` may use the conventional GML remote property model to make a reference to a time instant described elsewhere, or a limit may be indicated as a direct position.

EXAMPLE 2 The following mixed example shows both of these, as well as including the optional `gml:duration` property:

```
<gml:TimePeriod gml:id="p22">
  <gml:begin xlink:href="#t11"/>
  <gml:endPosition>2002-05-23</gml:endPosition>
  <gml:duration>P1Y</gml:duration>
</gml:TimePeriod>
```

Use of `xlink:ref`'s for remote references to temporal values is not considered in this document.

timePosition

Temporal positions are specified with respect to a selected temporal reference system. `gml:TimePositionType` supports the description of temporal position according to the subtypes described in ISO 19108.

Values based on calendars and clocks use lexical formats that are based on ISO 8601, as described in XML Schema Part 2:2001. A decimal value may be used with coordinate

systems such as GPS time or UNIX time. A URI may be used to provide a reference to some era in an ordinal reference system¹.

The various attributes that can be attached to `timePosition` enable a wide variety of time positions to be described as illustrated by the following list of examples.

Example 1 : Default Time Position using ISO 8601 Time String :

```
<gml:timePosition>2006-10-21T12:01:20Z</gml:timePosition>
```

Example 2 : Indeterminate Time Position using before/after/now :

```
<gml:timePosition indeterminatePosition="after">2001</gml:timePosition>
```

Example 3 : Time Position using a frame attribute – cardinal time:

```
<gml:timePosition frame="http://my.big.org/TRS/GPS">23846321.01</gml:timePosition>
```

Example 4 : Time Position using a frame attribute – ordinal time:

```
<gml:timePosition frame="http://my.big.org/TRS/archaeology"> http://my.history.org/eras/bronzeAge
</gml:timePosition>
```

Example 5 : Time Position using a frame attribute – ordinal time with calendar era

```
<gml:timePosition frame="http://my.big.org/TRS/calendars/japanese" calendarEraName="Meiji">0025-
03</gml:timePosition>
```

Only the ISO 8601 time positions are considered in this document.

timeLength, duration, timeInterval

The length of a time period is described using the group `gml:timeLength`, which is in effect a choice between two elements “duration” and “timeInterval” as shown in the following examples. Note that `gml:duration` conforms to the ISO 8601 syntax as implemented in the XML Schema duration type. The alternative `gml:timeInterval` conforms to the ISO 11404 syntax which uses floating point values for temporal length.

To express a time length length of 5 days, 14 hours, and 30 minutes any of the following instances are acceptable:

```
<duration>P5DT14H30M</duration>
```

```
<timeInterval unit="hour" radix="10" factor="0">134.5</timeInterval>
```

```
<timeInterval unit="other:week" radix="10" factor="0"> 0.800595</timeInterval>
```

6.1.2 Dynamic Features in GML

GML provides a basic model, called Dynamic Feature to represent time dependent phenomena, essentially through the description of the states of that phenomena. This readily include the motion of nominally rigid objects (aircraft, persons, vehicles, ships) as

¹ e.g. a geological epoch

well as the temporal evolution (change of shape, position, other properties) of complex objects such as fires, floods, or urban infrastructure.

Dynamic Features are typically described through the gml:history property which contains a sequence of TimeSlices describing the states of the feature. The TimeSlice which contains the time varying properties of the feature is concretely defined (like the time invariant properties) in an application schema. The structure of a dynamic feature is thus as shown in Figure 6.1.2-1.

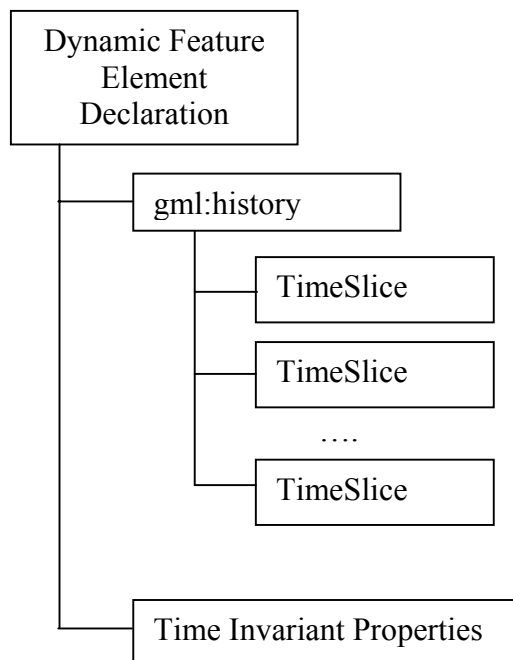


Figure 6.1-1 Structure of Dynamic Feature

The key component here is that of the gml:TimeSlice. A concrete TimeSlice is derived in an application schema from gml:AbstractTimeSlice which has the following content model.

```

<element name="AbstractTimeSlice"
type="gml:AbstractTimeSliceType" abstract="true"
substitutionGroup="gml:AbstractGML"/>

<complexType name="AbstractTimeSliceType" abstract="true">
  <complexContent>
    <extension base="gml:AbstractGMLType">
      <sequence>
        <element ref="gml:validTime"/>
        <element ref="gml:dataSource" minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
  
```

```

    </extension>
  </complexContent>
</complexType>

```

The `gml:validTime` property is attached to each `TimeSlice` (by inheritance) and hence allows the time associated with the `TimeSlice` to be a `TimePeriod` or a `TimeInstant` and to build on the rich variety of time positions supported in GML.

A sample Dynamic Feature with a history of time slices is as follows:

Example 1: Dynamic Feature

```

<app:Cyclone gml:id="c1">
  <gml:history>
    <gml:MovingObjectStatus>
      <gml:validTime><gml:TimeInstant>
        <gml:timePosition>2005-11-28T13:00:00Z</gml:timePosition>
      </gml:TimeInstant></gml:validTime>
      <gml:location><gml:Point gml:id="p1" srsName="urn:x-ogc:def:crs:EPSG:6.6:4326">
        <gml:pos>-35 140</gml:pos>
      </gml:Point></gml:location>
      <gml:speed uom="#kph">12</gml:speed>
      <gml:bearing>
        <gml:CompassPoint>SE</gml:CompassPoint>
      </gml:bearing>
    </gml:MovingObjectStatus>
    <gml:MovingObjectStatus>
      <gml:validTime><gml:TimeInstant>
        <gml:timePosition>2005-11-28T14:00:00Z</gml:timePosition>
      </gml:TimeInstant></gml:validTime>
      <gml:location><gml:Point>
        <gml:pos>-34.9 140.1</gml:pos gml:id="p1" srsName="urn:x-ogc:def:crs:EPSG:6.6:4326">
      </gml:Point></gml:location>
      <gml:speed uom="#kph">23</gml:speed>
      <gml:bearing>
        <gml:CompassPoint>ESE</gml:CompassPoint>
      </gml:bearing>
    </gml:MovingObjectStatus>
  </gml:history>
  <app:impactTime>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-11-28T18:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-11-28T22:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </app:impactTime>
</app:Cyclone>

```

Note that in the above example, the TimeSlice is a gml:MovingObjectStatus which provides properties to describe the position and velocity of a moving point object or rigid body.

6.2 Comparison Operators and Temporal Queries

Even without the addition of temporal operators, the WFS specification can be used to perform temporal queries. This section considers the capabilities and limitations of this approach.

The material in this section is based on the WFS 1.1 specification, although all functionality discussed in this section would also apply to WFS 1.0.

A simple wfs:GetFeature request looks as follows:

```
<wfs:GetFeature
  service="WFS"
  version="1.1.0"
  outputFormat="text/xml; subtype=gml/3.1.1"
  xmlns:app="http://www.appserver.com/stuff"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs ../wfs/1.1.0/WFS.xsd">
  <wfs:Query typeName="app:Cyclone">
    <ogc:Filter>
      <ogc:GmlObjectId gml:id="c1"/>
    </ogc:Filter>
  </wfs:Query>
</wfs:GetFeature>
```

This would return the complete Cyclone dynamic feature shown in section 6.1.2.

To select the TimeSlice that meets a specific time restriction we need to phrase a temporal query, which might be handled as follows:

```
<?xml version="1.0" ?>
<GetFeature
  version="1.1.0"
  service="WFS"
  xmlns="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:app="http://www.someserver.com/myns"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs ../wfs/1.1.0/WFS.xsd
  http://www.someserver.com/myns Person.xsd">
  <Query typeName="Cyclone">
    <ogc:Filter>
      <ogc:PropertyIsGreaterThan>
        <ogc:PropertyName>app:Cyclone/gml:history/gml:MovingObjectStatus/gml:validTime/gml:TimeInst
          ant/gml:timePosition</ogc:PropertyName>
        <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
      </ogc:PropertyIsGreaterThan>
    </ogc:Filter>
  </Query>
</GetFeature>
```

This query would be “logically” equivalent to the following query using WFS 1.2 temporal operators, namely:

```
<?xml version="1.0" ?>
<GetFeature
  version="1.2.0"
  service="WFS"
  xmlns="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:app="http://www.someserver.com/myns"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs ../wfs/1.1.0/WFS.xsd
  http://www.someserver.com/myns Person.xsd">
  <Query typeName="Cyclone">
    <ogc:Filter>
      <ogc:TM_After>
        <ogc:PropertyName>app:Cyclone/gml:history/gml:MovingObjectStatus/gml:validTime</ogc:Property
          Name>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </ogc:TM_After>
    </ogc:Filter>
  </Query>
</GetFeature>
```

Note that processing the first query requires that the WFS be able to use the PSV (Post Schema Validation) typing information to cast the literal string “2005-05-17T00:00:00Z” to an ISO 8601 dateTime (XML Schema), and to correctly perform the magnitude comparison.

The specific semantics for the interpretation of the temporal operators in terms of WFS 1.0 Comparison operators is provided in ISO 19108:2003. This mapping depends on the temporal value type of the property in the comparison operation.

There are a number of cases as enumerated in ISO 19108:2003 and partially repeated in 05-093r2.

Case 1: Property and Literal both have Time Instant Values

Example 1: TM_After

```
<ogc:Filter>
  <ogc:PropertyIsGreaterThanOrEqualTo>
    <ogc:PropertyName>gml:history/gml:MovingObjectStatus/gml:validTime/
      gml:TimeInstant/gml:timePosition</ogc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
  </ogc:PropertyIsGreaterThanOrEqualTo>
</ogc:Filter>
```

Corresponds to:

```
<ogc:Filter>
  <ogc:TM_After>
    <ogc:PropertyName>gml:history/gml:MovingObjectStatus/gml:validTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_After>
```

```
</ogc:Filter>
```

Example 2: TM_Before

```
<ogc:Filter>
  <ogc:PropertyIsLessThan>
    <ogc:PropertyName>gml:history/gml:MovingObjectStatus/gml:validTime/
      gml:TimeInstant/gml:timePosition</ogc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
  </ogc:PropertyIsLessThan>
</ogc:Filter>
```

Corresponds to:

```
<ogc:Filter>
  <ogc:TM_Before>
    <ogc:PropertyName>gml:history/gml:MovingObjectStatus/gml:validTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_Before>
</ogc:Filter>
```

Example 3: TM_Equals

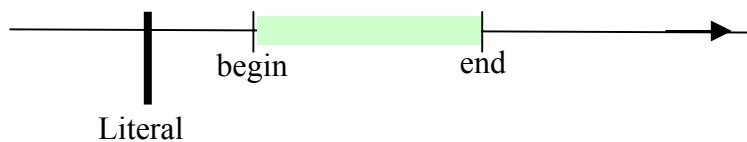
```
<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>gml:history/gml:MovingObjectStatus/gml:validTime/
      gml:TimeInstant/gml:timePosition</ogc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
  </ogc:PropertyIsEqualTo>
</ogc:Filter>
```

Corresponds to:

```
<ogc:Filter>
  <ogc:TM_Equals>
    <ogc:PropertyName>gml:history/gml:MovingObjectStatus/gml:validTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_Equals>
</ogc:Filter>
```

Case 2: Property has Time Period Value and Literal has Time Instant Value

Example 4: TM_After (Simplified Version)



```
<ogc:Filter>
  <ogc:PropertyIsGreaterThan>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</
      ogc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
  </ogc:PropertyIsGreaterThan>
```

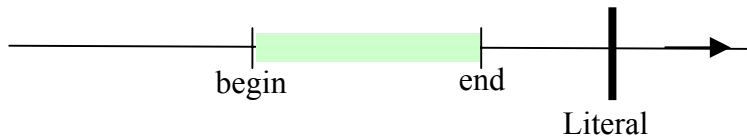
```
<ogc:PropertyIsGreaterThan>
</ogc:Filter>
```

Corresponds to:

```
<ogc:Filter>
  <ogc:TM_After>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_After>
</ogc:Filter>
```

Note that the query using comparison operators is a significant simplification relative to what would really be required since the gml:TimePeriod admits a variety of expressions for the limits of the time interval.

Example 5: TM_Before (Simplified Version)

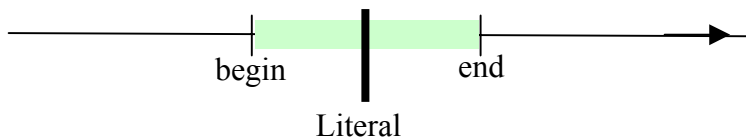


```
<ogc:Filter>
  <ogc:PropertyIsLessThan>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePosition</o
    gc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
  </ogc:PropertyIsLessThan>
</ogc:Filter>
```

Corresponds to:

```
<ogc:Filter>
  <ogc:TM_Before>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_Before>
</ogc:Filter>
```

Example 6: TM_Contains (simplified version)



```

<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
    </ogc:PropertyIsGreaterThan>
    <ogc:PropertyIsLessThan>
      <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T05:00:00Z</ogc:Literal>
    </ogc:PropertyIsLessThan>
  </ogc:And>
</ogc:Filter>

```

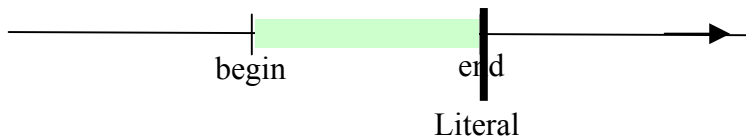
Corresponds to:

```

<ogc:Filter>
  <ogc:TM_Contains>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_Contains>
</ogc:Filter>

```

Example 7: TM_EndedBy



```

<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePosition</ogc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
  </ogc:PropertyIsEqualTo>
</ogc:Filter>

```

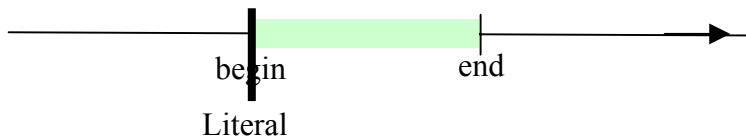
Corresponds to:

```

<ogc:Filter>
  <ogc:TM_EndedBy>
    <ogc:PropertyName>app:Cyclone/app:impactTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_EndedBy>
</ogc:Filter>

```

Example 8: TM_BegunBy (simplified version)



```

<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</
      ogc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
  </ogc:PropertyIsEqualTo>
</ogc:Filter>

```

Corresponds to:

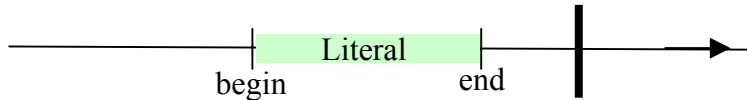
```

<ogc:Filter>
  <ogc:TM_BegunBy>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_BegunBy>
</ogc:Filter>

```

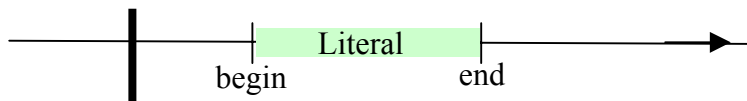
Case 3: Property has Time Instant Value and Literal has Time Period Value

Example 9: TM_After (Simplified Version)



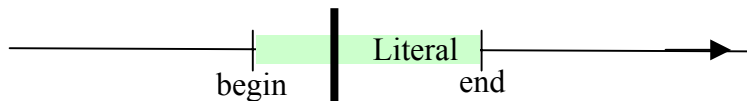
Note that this temporal comparison cannot be specified using the WFS 1.0 comparison operators, since there is no means of expressing the temporal literal. If the end of the temporal interval is known and can be expressed as a scalar literal, this of course reduces to the case of comparing Time Instants (Case 1).

Example 10: TM_Before (Simplified Version)



This can be treated as for Example 9.

Example 11: TM_During (simplified version)



```

<ogc:Filter>
  <ogc:PropertyIsBetween>
    <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePosition</ogc:PropertyName>

```



```

    <ogc:LowerBoundary>
      <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal> <!--corresponds to begin -->
    </ogc:LowerBoundary>
    <ogc:UpperBoundary>
      <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal> <!--corresponds to end -->
    </ogc:UpperBoundary>
  </ogc:PropertyIsBetween>
</ogc:Filter>

```

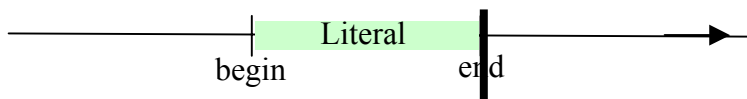
Corresponds to:

```

<ogc:Filter>
  <ogc:TM_During>
    <ogc:PropertyName>app:Cyclone/app:impactTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_During>
</ogc:Filter>

```

Example 12: TM_Ends



This expression cannot be strictly represented using the WFS 1.0 comparison operators as there is no representation for the literal temporal value.

```

<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePosition</o
    gc:PropertyName>
    <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to end -->
  </ogc:PropertyIsEqualTo>
</ogc:Filter>

```

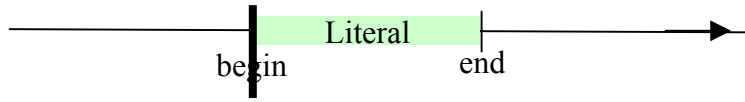
Corresponds to:

```

<ogc:Filter>
  <ogc:TM_Ends>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_Ends>
</ogc:Filter>

```

Example 13: TM_Begins (simplified version)



This expression cannot be strictly represented using the WFS 1.0 comparison operators as there is no representation for the literal temporal value.

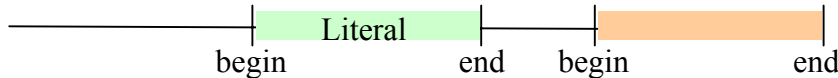
```
<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</
      ogc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z </ogc:Literal>
  </ogc:PropertyIsEqualTo>
</ogc:Filter>
```

Corresponds to:

```
<ogc:Filter>
  <ogc:TM_Begins>
    <ogc:PropertyName>app:Cyclone/app:impactTime </ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>.
    </gml:TimePeriod>
  </ogc:TM_Begins>
</ogc:Filter>
```

Case 4: Property has Time Period Value and Literal has Time Period Value

Example 14: TM_After (Simplified Version)



Note that this temporal comparison cannot be strictly specified using the WFS 1.0 comparison operators, since there is no means of expressing the temporal literal. If the end of the temporal interval is known and can be expressed as a scalar literal, the case reduces to the comparison of a Property with Time Period Value and this of course reduces to the case of comparing Time Instants (Case 1).

```
<ogc:Filter>
  <ogc:PropertyIsGreaterThan>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</
      ogc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal>
  </ogc:PropertyIsGreaterThan>
</ogc:Filter>
```

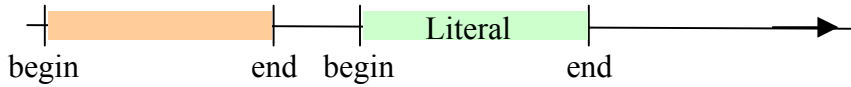
Corresponds to:

```

<ogc:Filter>
  <ogc:TM_After>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_After>
</ogc:Filter>

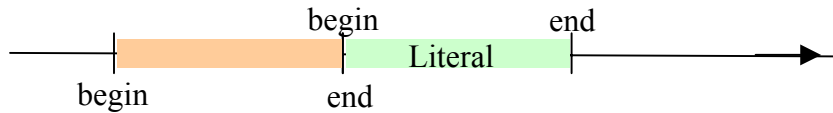
```

Example 15: TM_Before (Simplified Version)



This can be treated as for Example 14

Example 16: TM_Meets (simplified version)



This example cannot strictly be represented as there is no way to represent the literal temporal interval.

```

<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePosition</o
gc:PropertyName>
    <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal> !—corresponds to begin
  </ogc:PropertyIsEqualTo>
</ogc:Filter>

```

Corresponds to:

```

<ogc:Filter>
  <ogc:TM_Meets>

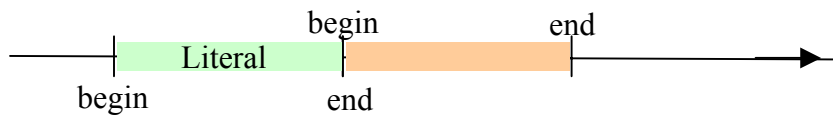
```

```

    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_Meets>
</ogc:Filter>

```

Example 17: TM_MetBy



This example cannot strictly be represented as a WFS 1.0 comparison as there is no way to represent the literal temporal interval.

```

<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</
    ogc:PropertyName>
    <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!-- corresponds to end -->
  </ogc:PropertyIsEqualTo>
</ogc:Filter>

```

Corresponds to:

```

<ogc:Filter>
  <ogc:TM_MetBy>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_MetBy>
</ogc:Filter>

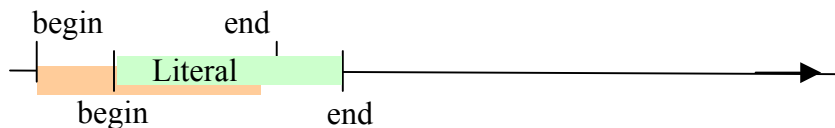
```

```

    </ogc:TM_MetBy>
  </ogc:Filter>

```

Example 18: TM_Overlaps (simplified version)



This example cannot strictly be represented as a WFS 1.0 comparison as there is no way to represent the literal temporal interval. A rough approximation is as follows:

```

<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsLessThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/
gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T00:00:00Z </ogc:Literal> <!--corresponds to begin -->
    </ogc:PropertyIsLessThan>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/g
ml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to begin -->
    </ogc:PropertyIsGreaterThan>
    <ogc:PropertyIsLessThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/g
ml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to end -->
    </ogc:PropertyIsLessThan>
  </ogc:And>
</ogc:Filter>

```

Corresponds to:

```

<ogc:Filter>
  <ogc:TM_Overlaps>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_Overlaps>
</ogc:Filter>

```

Example 19: TM_BegunBy (simplified version)



This example cannot strictly be represented as a WFS 1.0 comparison as there is no way to represent the literal temporal interval. A rough approximation is as follows:

```

<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/
gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal> <!--corresponds to begin -->
    </ogc:PropertyIsEqualTo>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/g
mml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to end -->
    </ogc:PropertyIsGreaterThan>
  </ogc:And>
</ogc:Filter>

```

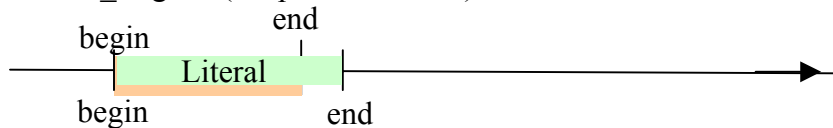
Corresponds to:

```

<ogc:Filter>
  <ogc:TM_BegunBy>
    <ogc:PropertyName>app:Cyclone/app:impactTime </ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_BegunBy>
</ogc:Filter>

```

Example 19: TM_Begins (simplified version)



This example cannot strictly be represented as a WFS 1.0 comparison as there is no way to represent the literal temporal interval. A rough approximation is as follows:

```

<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/
gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal> <!--corresponds to begin -->
    </ogc:PropertyIsEqualTo>

```

```

    <ogc:PropertyIsLessThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/g
ml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to end -->
    </ogc:PropertyIsLessThan>
  </ogc:And>
</ogc:Filter>

```

Corresponds to:

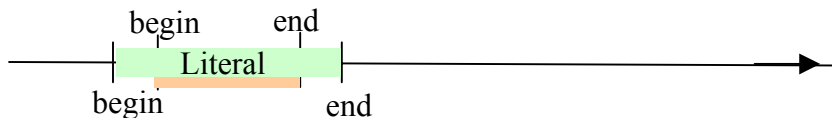
```

<ogc:Filter>
  <ogc:TM_Begins>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_Begins>
</ogc:Filter>

```

TM_Ends, TM_EndedBy are handled in a similar manner.

Example 20: TM_During (simplified version)



This example cannot strictly be represented as a WFS 1.0 comparison as there is no way to represent the literal temporal interval. A rough approximation is as follows:

```

<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosi
tion</ogc:PropertyName>
      <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal> <!--corresponds to begin
    </ogc:PropertyIsGreaterThan>
    <ogc:PropertyIsLessThan>
      <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePositi
on</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to end
    </ogc:PropertyIsLessThan>
  </ogc:And>
</ogc:Filter>

```

Corresponds to:

```

<ogc:Filter>
  <ogc:TM_During>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>

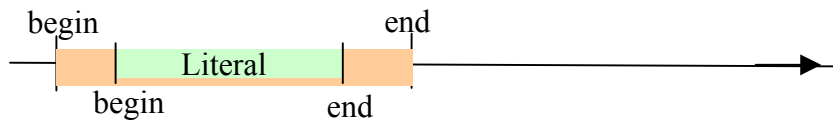
```

```

    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_During>
</ogc:Filter>

```

Example 21: TM_Contains (simplified version)



This example cannot strictly be represented as a WFS 1.0 comparison as there is no way to represent the literal temporal interval. A rough approximation is as follows:

```

<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsLessThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/
gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal> <!--corresponds to begin -->
    </ogc:PropertyIsLessThan>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/g
ml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to end -->
    </ogc:PropertyIsGreaterThan>
  </ogc:And>
</ogc:Filter>

```

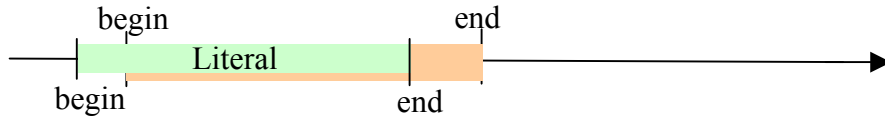
Corresponds to:

```

<ogc:Filter>
  <ogc:TM_Contains>
    <ogc:PropertyName>app:impactTime </ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_Contains>
</ogc:Filter>

```


Example 22: TM_OverlappedBy (simplified version)



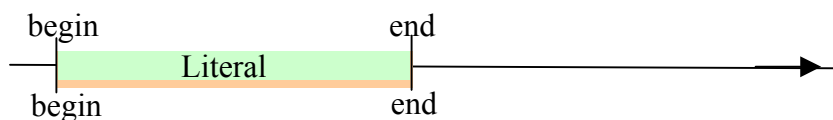
This example cannot strictly be represented as a WFS 1.0 comparison as there is no way to represent the literal temporal interval. A rough approximation is as follows:

```
<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T00:00:00Z </ogc:Literal> <!--corresponds to begin -->
    </ogc:PropertyIsGreaterThan>
    <ogc:PropertyIsLessThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z </ogc:Literal> <!--corresponds to end -->
    </ogc:PropertyIsLessThan>
    <ogc:PropertyIsGreaterThan>
      <ogc:PropertyName>app:Cyclone/app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to end -->
    </ogc:PropertyIsGreaterThan>
  </ogc:And>
</ogc:Filter>
```

Corresponds to:

```
<ogc:Filter>
  <ogc:TM_OverlappedBy>
    <ogc:PropertyName>app:Cyclone/app:impactTime</ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_OverlappedBy>
</ogc:Filter>
```

Example 21: TM_Equals (simplified version)



This example cannot strictly be represented as a WFS 1.0 comparison as there is no way to represent the literal temporal interval. A rough approximation is as follows:

```

<ogc:Filter>
  <ogc:And>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T00:00:00Z</ogc:Literal> <!--corresponds to begin -->
    </ogc:PropertyIsEqualTo>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>app:impactTime/gml:TimePeriod/gml:end/gml:TimeInstant/gml:timePosition</ogc:PropertyName>
      <ogc:Literal>2005-05-17T20:00:00Z</ogc:Literal> <!--corresponds to end -->
    </ogc:PropertyIsEqualTo>
  </ogc:And>
</ogc:Filter>

```

Corresponds to:

```

<ogc:Filter>
  <ogc:TM_Equals>
    <ogc:PropertyName>app:impactTime</ogc:PropertyName>
    <gml:TimePeriod>
      <gml:begin>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T00:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:begin>
      <gml:end>
        <gml:TimeInstant>
          <gml:timePosition>2005-05-17T20:00:00Z</gml:timePosition>
        </gml:TimeInstant>
      </gml:end>
    </gml:TimePeriod>
  </ogc:TM_Equals>
</ogc:Filter>

```

6.3 Review of OGC 06-076 Temporal Features

This section provides review commentary on the Temporal Features Discussion Paper 06-076 relevant to the issue of Temporal Queries for a WFS.

- As in most temporal database discussions, 06-076 distinguishes Transaction Time (also known as timestamp), the time at which the database is updated, and Valid Time (the real world time relative to which feature changes take place). We note that GML only supports valid time (`gml:validTime`), and the proposal for WFS temporal operators does not include any support for a database time.
- The paper discusses temporal evolution in terms of a series of “revisions” of a feature. This could be confused with versions of the feature. The dynamic feature model in GML is based on a history consisting of a sequence of time slices which contain the values of the time varying properties of the feature. One could understand the “revision” of 06-076 as meaning the feature and its time invariant properties together with any selected time slice in the feature history.
- The paper appears to imply that a features identity might vary in time, and it is clear that such a thing does happen in practice. The current GML dynamic feature model assumes that the feature’s identity is time invariant, and this is reflected in the proposed changes for temporal support for WFS. It is suggested that the GML temporal topology be considered as a basis for building a future evolutive feature type that incorporates changes in feature identity.
- The paper introduces the notion of an Event which is defined as a kind of feature “with an enumerated temporal domain”. This is very similar to the notion proposed for parent of Observation in the O&M Paper (05-087r2). It is suggested that these two Event models be analyzed and harmonized and incorporated into GML.
- A notion of “Feature Observation” is introduced which is considerably more restrictive than GML Observation or O&M Observation. It is suggested that this be replaced by on the existing models.
- A Moving Feature is defined by reference to ISO 19141 as a moving rigid body (time series of position, orientation and associated derivatives). This is more restrictive than the GML notion of dynamic feature. Work is underway to harmonize ISO 19141 and the Moving Object Status (a special case of GML Dynamic Feature).
- A notion of a non-feature “Publication” is introduced although this seems to be the same thing as a GML Feature Collection which is in turn a Feature. No additional ideas appear to be introduced in the concept of Publication.
- The paper notes that for Feature Histories that “While this model differs from an timeslice application schema defined in GML dynamic features, by using this

approach, the model follows a “supersede-not-delete” approach capable of working with WFS and a relational database”. This statement is not substantiated. There is no reason why the GML timeslice model will not work with WFS and with relational DBMS and a number of such implementations exist today. We would argue that GML time slice model is a “supersede-no-delete” model.

- Figure 4-3 shows the TransactionTime as a property of the feature object. While this time should be accessible it is not a characteristic of the feature and should be shown as metadata.
- The paper provides a suggested list of actions. Actions for which we have specific comments are repeated here:

WFS Changes Proposed from 06-076	Comment
The GetCapabilities will have additional optional elements to contain the temporal metadata of each temporal feature type	We believe that rather than overload Capabilities that a new interface(s) should be added – e.g. GetTemporalExtent()
A transaction to store/update/delete temporal features should be introduced (presumably for WFS).	This can be handled through the combination of the Filter 1.2 Temporal operators and the WFS 1.2 fine grained object update. Update of a dynamic feature (e.g. insert a new time slice, edit a time slice) is a significant use case for the fine grained object update change proposal for WFS 1.2. We believe that this requires further “testing” within the ISO and OGC environments.
Upon insertion, Revision_Number is calculated automatically based upon the sequence of revision of the feature.	We believe that this Revision_Number should be replaced by the GML id of the TimeSlice.

A number of issues are also raised. These are discussed below.

Issue from 06-076	Comment
1. Use of indefinite time value for valid and transaction end time in feature history. The value “9999-12-31T23:59:59” was used by TEA in order work with filter queries on time range. Determine if this is an acceptable method of representing	This is handled in ISO 19136 using the indeterminate attribute on startPosition, endPosition in a TimePeriod. This attribute can take the values unknown, before, after. This should be supported in WFS requests that wish to use indefinite time values.

<p>acceptable method of representing indefinite time or if other values can be used that are compatible with temporal filter queries.</p>	
---	--

7 Temporal Query Experiments

A series of temporal query experiments were conducted by Galdos using the Galdos Cartalinea products.

- o Galdos Cartalinea v1. on Oracle

This is a compliant implementation of the WFS 1.0 Specification and which implement many features of WFS 1.1.

7.1 Cartalinea on Oracle

For the tests in this section we used a variant of a sensor schema, with the WFS collecting real time sensor data. All tests were conducted using the Galdos WFS test framework. The material reported below is copied from that test framework.

This first test uses a WFS request to insert a point feature (abc:GenericDataSource). This object has a temporal valued property, namely abc:dateTime which has a value which is a gml:TimePeriod.

7.1.1 Test Insert – TimePeriod using gml:begin/gml:end

Transaction

```
<wfs:Transaction version="1.0.0" service="WFS"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml feature.xsd
http://www.galdosinc.com/wfs/testpointfeatures TestPointFeature.xsd
http://www.opengis.net/wfs WFS-transaction.xsd ">
  <wfs:Insert handle="insert TestPointFeature features"
srsName="EPSG:4269">
    <GenericDataSource gml:id="GALDOS.3-100000012"
xmlns="http://www.sensors.org/abc" xmlns:is="http://www.sensors.org/is"
xmlns:abc="http://www.sensors.org/abc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.sensors.org/abc ../../xsd/abc/abc.xsd">
      <abc:location>
        <gml:Point>
          <gml:coordinates>35.893040,-84.195691</gml:coordinates>
        </gml:Point>
      </abc:location>
      <abc:datetime>
        <gml:TimePeriod>
          <gml:begin>
            <gml:TimeInstant>
              <gml:timePosition>2002-06-06T02:54:05-
03:00</gml:timePosition>
```

```

        </gml:TimeInstant>
    </gml:begin>
    <gml:end>
        <gml:TimeInstant>
            <gml:timePosition>2002-06-06T02:54:06-
03:00</gml:timePosition>
        </gml:TimeInstant>
    </gml:end>
</gml:TimePeriod>
</abc:datetime>
</GenericDataSource>
</wfs:Insert>
</wfs:Transaction>

```

Result: Pass

The next test is the same as 7.1.1 except that begin is replaced by beginPosition and end is replaced by endPosition.

7.1.2 Test Insert - TimePeriod2 – using gml:beginPosition/gml:endPosition**Transaction**

```

<wfs:Transaction version="1.0.0" service="WFS"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml feature.xsd
http://www.galdosinc.com/wfs/testpointfeatures TestPointFeature.xsd
http://www.opengis.net/wfs WFS-transaction.xsd ">
  <wfs:Insert srsName="EPSG:4269">
    <GenericDataSource gml:id="TRP.ABC.3-100000012"
xmlns="http://www.sensors.org/abc" xmlns:is="http://www.sensors.org/is"
xmlns:abc="http://www.sensors.org/abc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.sensors.org/abc ../../xsd/abc/abc.xsd">
      <abc:location>
        <gml:Point>
          <gml:coordinates>35.893040,-84.195691</gml:coordinates>
        </gml:Point>
      </abc:location>
      <abc:datetime>
        <gml:TimePeriod>
          <gml:beginPosition>2002-06-06T02:54:05-
03:00</gml:beginPosition>
          <gml:endPosition>2002-06-06T02:54:06-03:00</gml:endPosition>
        </gml:TimePeriod>
      </abc:datetime>
    </GenericDataSource>
  </wfs:Insert>
</wfs:Transaction>

```

Result-PASS**7.1.3 Test Insert - TimePeriod3 - beginPosition/endPosition/@indeterminatePosition**

This next test is the same as that in 7.1.2, except that the endPosition is given as indeterminate. Note that the expected response, namely a service exception, is obtained.

Transaction

```
<wfs:Transaction version="1.0.0" service="WFS"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml feature.xsd
http://www.galdosinc.com/wfs/testpointfeatures TestPointFeature.xsd
http://www.opengis.net/wfs WFS-transaction.xsd ">
  <wfs:Insert srsName="EPSG:4269">
    <GenericDataSource gml:id="GALDOS.3-100000012"
xmlns="http://www.sensors.org/abc" xmlns:is="http://www.sensors.org/is"
xmlns:abc="http://www.sensors.org/abc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.sensors.org/abc ../../xsd/abc/abc.xsd">
      <abc:location>
        <gml:Point>
          <gml:coordinates>35.893040,-84.195691</gml:coordinates>
        </gml:Point>
      </abc:location>
      <abc:datetime>
        <gml:TimePeriod>
          <gml:beginPosition>2002-06-06T02:54:05-
03:00</gml:beginPosition>
          <gml:endPosition indeterminatePosition="unknown"/>
        </gml:TimePeriod>
      </abc:datetime>
    </GenericDataSource>
  </wfs:Insert>
</wfs:Transaction>
```

Result-FAIL

```
<ServiceExceptionReport xmlns="http://www.opengis.net/ogc"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <ServiceException code="There was an error processing the
event:TransactionException: null"/>
</ServiceExceptionReport>
```


7.1.4 Test Get - Property filter – TM_During

The next test queries the database using the mapping of temporal operators to the comparison operators of WFS 1.0. In this case the “model” query is a temporal comparison of a gml:TimeInstant and a gml:TimePeriod. This corresponds to Case 3, Example 10, i.e. TM_During in Section 6.2 above. Note that the TimeInstant used was (gml:TimePeriod/gml:beginPosition) rather than a gml:TimeInstant/gml:timePosition, however these are logically equivalent.

Transaction

```
<wfs:Transaction version="1.0.0" service="WFS"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml feature.xsd
http://www.galdosinc.com/wfs/testpointfeatures TestPointFeature.xsd
http://www.opengis.net/wfs WFS-transaction.xsd ">
  <wfs:Insert srsName="EPSG:4269">
    <GenericDataSource gml:id="GALDOS.3-100000012"
xmlns="http://www.sensors.org/abc" xmlns:is="http://www.sensors.org/is"
xmlns:abc="http://www.sensors.org/abc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.sensors.org/abc ../../xsd/abc/abc.xsd">
      <abc:location>
        <gml:Point>
          <gml:coordinates>35.893040,-84.195691</gml:coordinates>
        </gml:Point>
      </abc:location>
      <abc:datetime>
        <gml:TimePeriod>
          <gml:beginPosition>2002-06-06T02:54:05-
03:00</gml:beginPosition>
          <gml:endPosition>2002-06-06T02:54:06-03:00</gml:endPosition>
        </gml:TimePeriod>
      </abc:datetime>
    </GenericDataSource>
  </wfs:Insert>
</wfs:Transaction>
```

The insert succeeds, then

```
<wfs:GetFeature service="WFS" version="1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:abc="http://www.sensors.org/abc"
xmlns:p21="http://www.sensors.org/abc/p2142"
xmlns:is="http://www.sensors.org/is"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs
../../../../../../../../xml/xsd/WFS-1.0.0/WFS_Wrapper.xsd">
  <wfs:Query typeName="abc:GenericDataSource">
```

```

      <ogc:Filter>
        <ogc:PropertyIsBetween>
          <ogc:PropertyName>abc:datetime/gml:TimePeriod/gml:beginPosition<
/ogc:PropertyName>
          <ogc:LowerBoundary><ogc:Literal>2002-06-
06T02:54:04-03:00</ogc:Literal></ogc:LowerBoundary>
          <ogc:UpperBoundary><ogc:Literal>2002-06-
06T02:54:07-03:00</ogc:Literal></ogc:UpperBoundary>
        </ogc:PropertyIsBetween>
      </ogc:Filter>
    </wfs:Query>
  </wfs:GetFeature>

```

Result-PASS

7.1.5 Test Update - Temporal property

Transaction

```

<wfs:Transaction version="1.0.0" service="WFS"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:gml="http://www.opengis.net/gml"
xmlns:abc="http://www.sensors.org/abc"
xmlns:p21="http://www.sensors.org/abc/p2142"
xmlns:is="http://www.sensors.org/is" >
  <wfs:Update typeName="abc:GenericDataSource">
    <wfs:Property>
      <wfs>Name>abc:datetime/gml:TimeInstant/gml:timePosition</wfs>Name>
      <wfs:Value>2002-06-06T01:02:03.99Z</wfs:Value>
    </wfs:Property>
    <ogc:Filter>
      <ogc:FeatureId fid="GALDOS.3-100000012"/>
    </ogc:Filter>
  </wfs:Update>
</wfs:Transaction>

```

Result-FAIL

Decimal Seconds get truncated (i.e. 3.99 in seconds field in update comes back as 3.000). Note the time zone of response is the server time zone (expected behaviour).

```

<wfs:FeatureCollection >
  <gml:boundedBy>
    <gml:Envelope srsName="EPSG:4269">
      <gml:lowerCorner>35.89304 -84.195691</gml:lowerCorner>
      <gml:upperCorner>35.89304 -84.195691</gml:upperCorner>
    </gml:Envelope>
  </gml:boundedBy>
  <gml:featureMember>
    <abc:GenericDataSource gml:id="GALDOS.3-100000012">
      <abc:location>
        <gml:Point srsName="EPSG:4269">

```

```

    <gml:pos srsDimension="2">35.89304 -84.195691</gml:pos>
  </gml:Point>
</abc:location>
<abc:datetime>
  <gml:TimeInstant frame="#ISO-8601" gml:id="GML_TT_100271">
    <gml:timePosition frame="#ISO-8601">2002-06-05T18:02:03.000-
07:00</gml:timePosition>
  </gml:TimeInstant>
</abc:datetime>
</abc:GenericDataSource>
</gml:featureMember>
</wfs:FeatureCollection>

```

7.1.6 TM_After

This experiment is based on comparison of two time instants.

WFS request:

```

<?xml version="1.0" encoding="UTF-8"?>
<wfs:GetFeature service="WFS" version="1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:app="http://www.sensors.org/app"
xmlns:abc="http://www.sensors.org/app/abc"
xmlns:is="http://www.sensors.org/is"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs
../../../../../../../../xml/xsd/WFS-1.0.0/WFS_Wrapper.xsd">
  <wfs:Query typeName="app:GenericDataSource">
    <ogc:Filter>
      <ogc:PropertyIsGreaterThan>

<ogc:PropertyName>app:datetime/gml:TimePeriod/gml:begin/gml:TimeInstant
/gml:timePosition</ogc:PropertyName>
<ogc:Literal>2002-06-05T22:54:04.000-
07:00</ogc:Literal></ogc:PropertyIsGreaterThan>

    </ogc:Filter>
  </wfs:Query>
</wfs:GetFeature>

```

Response: Pass

```

<wfs:FeatureCollection xmlns:gml="http://www.opengis.net/gml"
xmlns:is="http://www.sensors.org/is" xmlns:abc="http://www.sensors.org/app/abc"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sensors.org/app test.xsd">
  <gml:boundedBy>
    <gml:Envelope srsName="EPSG:4269">
      <gml:lowerCorner>35.89304 -84.195691</gml:lowerCorner>

```

```

    <gml:upperCorner>35.89304 -84.195691</gml:upperCorner>
  </gml:Envelope>
</gml:boundedBy>
<gml:featureMember>
  <app:GenericDataSource gml:id="TRP.GALDOS.3-100000022">
    <app:location>
      <gml:Point srsName="EPSG:4269">
        <gml:pos srsDimension="2">35.89304 -84.195691</gml:pos>
      </gml:Point>
    </app:location>
    <app:datetime>
      <gml:TimePeriod frame="#ISO-8601" gml:id="GML_TD_100046">
        <gml:begin xlink:type="simple">
          <gml:TimeInstant frame="#ISO-8601" gml:id="GML_TT_100412">
            <gml:timePosition frame="#ISO-8601">2002-06-05T22:54:05.000-
07:00</gml:timePosition>
          </gml:TimeInstant>
        </gml:begin>
        <gml:end xlink:type="simple">
          <gml:TimeInstant frame="#ISO-8601" gml:id="GML_TT_100413">
            <gml:timePosition frame="#ISO-8601">2002-06-05T22:54:06.000-
07:00</gml:timePosition>
          </gml:TimeInstant>
        </gml:end>
      </gml:TimePeriod>
    </app:datetime>
  </app:GenericDataSource>
</gml:featureMember>
</wfs:FeatureCollection>

```

7.1.7 TM_Before

The first test uses the same time interval and time position as in the previous case (7.1.9) and hence should return an empty result as it does. Note that we are now using the comparison operator as in Example 2., Case 1.

WFS Request:

```

<?xml version="1.0" encoding="UTF-8"?>
<wfs:GetFeature service="WFS" version="1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:app="http://www.sensors.org/app"
xmlns:abc="http://www.sensors.org/app/abc"
xmlns:is="http://www.sensors.org/is"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

```

```

xsi:schemaLocation="http://www.opengis.net/wfs
../../../../../../../../xml/xsd/WFS-1.0.0/WFS_Wrapper.xsd">
  <wfs:Query typeName="app:GenericDataSource">
    <ogc:Filter>
      <ogc:PropertyIsLessThan>
        <ogc:PropertyName>app:datetime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:time
        Position</ogc:PropertyName>
        <ogc:Literal>2002-06-05T22:54:05.000-07:00</ogc:Literal>
      </ogc:PropertyIsLessThan>    </ogc:Filter>
    </wfs:Query>
  </wfs:GetFeature>

```

Response: Pass

```

<wfs:FeatureCollection
xmlns:gml="http://www.opengis.net/gml"
xmlns:is="http://www.sensors.org/is"
xmlns:abc="http://www.sensors.org/app/abc"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:app="http://www.sensors.org/app"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sensors.org/app test.xsd"
  <gml:boundedBy>
    <gml:Null>inapplicable</gml:Null>
  </gml:boundedBy>
</wfs:FeatureCollection>

```

The time in the test was then adjusted so that property time instant was less than the test interval (gml:begin).

```

<wfs:FeatureCollection
xmlns:gml="http://www.opengis.net/gml"
xmlns:is="http://www.sensors.org/is"
xmlns:abc="http://www.sensors.org/app/abc"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:app="http://www.sensors.org/app"
<?xml version="1.0" encoding="UTF-8"?>
<wfs:GetFeature service="WFS" version="1.0.0"
xmlns:wfs="http://www.opengis.net/wfs" xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc" xmlns:app="http://www.sensors.org/app"
xmlns:abc="http://www.sensors.org/app/abc" xmlns:is="http://www.sensors.org/is"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs
../../../../../../../../xml/xsd/WFS-1.0.0/WFS_Wrapper.xsd">

```

```

<wfs:Query typeName="app:GenericDataSource">
  <ogc:Filter>
    <ogc:PropertyIsLessThan>
      <ogc:PropertyName>app:datetime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:time
      Position</ogc:PropertyName>
      <ogc:Literal>2002-08-05T22:54:07.000-07:00</ogc:Literal>
    </ogc:PropertyIsLessThan>  </ogc:Filter>
  </wfs:Query>
</wfs:GetFeature>

```

Result: Pass

```

<wfs:FeatureCollection xmlns:gml="http://www.opengis.net/gml"
xmlns:is="http://www.sensors.org/is" xmlns:abc="http://www.sensors.org/app/abc"
xmlns:ogc="http://www.opengis.net/ogc" xmlns:pr23="http://www.icao.int/IAXSTypes"
xmlns:app="http://www.sensors.org/app" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sensors.org/app test.xsd">
  <gml:boundedBy>
    <gml:Envelope srsName="EPSG:4269">
      <gml:lowerCorner>35.89304 -84.195691</gml:lowerCorner>
      <gml:upperCorner>35.89304 -84.195691</gml:upperCorner>
    </gml:Envelope>
  </gml:boundedBy>
  <gml:featureMember>
    <app:GenericDataSource gml:id="TRP.GALDOS.3-100000022">
      <app:location>
        <gml:Point srsName="EPSG:4269">
          <gml:pos srsDimension="2">35.89304 -84.195691</gml:pos>
        </gml:Point>
      </app:location>
      <app:datetime>
        <gml:TimePeriod frame="#ISO-8601" gml:id="GML_TD_100046">
          <gml:begin xlink:type="simple">
            <gml:TimeInstant frame="#ISO-8601" gml:id="GML_TT_100412">
              <gml:timePosition frame="#ISO-8601">2002-06-05T22:54:05.000-
              07:00</gml:timePosition>
            </gml:TimeInstant>
          </gml:begin>
          <gml:end xlink:type="simple">
            <gml:TimeInstant frame="#ISO-8601" gml:id="GML_TT_100413">
              <gml:timePosition frame="#ISO-8601">2002-06-05T22:54:06.000-
              07:00</gml:timePosition>
            </gml:TimeInstant>
          </gml:end>
        </gml:TimePeriod>
      </app:datetime>
    </app:GenericDataSource>
  </gml:featureMember>

```

```

    </app:datetime>
  </app:GenericDataSource>
</gml:featureMember>
</wfs:FeatureCollection>

```

7.1.8 TM_Overlaps

```

<?xml version="1.0" encoding="UTF-8"?>
<wfs:GetFeature service="WFS" version="1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:snet="http://www.sensors.org/app"
xmlns:abc="http://www.sensors.org/app/abc"
xmlns:is="http://www.sensors.org/is"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs
../../../../../../../../xml/xsd/WFS-1.0.0/WFS Wrapper.xsd">
  <wfs:Query typeName="snet:GenericDataSource">
    <ogc:Filter>
      <ogc:And>
        <ogc:PropertyIsLessThan>
<ogc:PropertyName>snet:datetime/gml:TimePeriod/gml:beginPosition</ogc:Pro
pertyName>
          <ogc:Literal>2002-06-05T18:05:00.000-07:00</ogc:Literal>
        </ogc:PropertyIsLessThan>
        <ogc:PropertyIsLessThan>
<ogc:PropertyName>snet:datetime/gml:TimePeriod/gml:endPosition</ogc:Pro
pertyName>
          <ogc:Literal>2002-06-05T21:00:00.000-07:00</ogc:Literal>
        </ogc:PropertyIsLessThan>
        <ogc:PropertyIsGreaterThan>
<ogc:PropertyName>snet:datetime/gml:TimePeriod/gml:endPosition</ogc:Pro
pertyName>
          <ogc:Literal>2002-06-05T18:05:00.000-07:00</ogc:Literal>
        </ogc:PropertyIsGreaterThan>
      </ogc:And>
    </ogc:Filter>
  </wfs:Query>
</wfs:GetFeature>

```

Result: Pass

```

<wfs:FeatureCollection xmlns:gml="http://www.opengis.net/gml"
xmlns:is="http://www.sensors.org/is"
xmlns:abc="http://www.sensors.org/app/abc"
<wfs:FeatureCollection xmlns:gml="http://www.opengis.net/gml"
xmlns:is="http://www.sensornet.gov/is"
xmlns:abc="http://www.sensornet.gov/app/abc"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sensornet.gov/app test.xsd">
  <gml:boundedBy>

```

```

    <gml:Envelope srsName="EPSG:4269">
      <gml:lowerCorner>35.89304 -84.195691</gml:lowerCorner>
      <gml:upperCorner>35.89304 -84.195691</gml:upperCorner>
    </gml:Envelope>
  </gml:boundedBy>
  <gml:featureMember>
    <app:GenericDataSource gml:id="TRP.GALDOS.3-100000035">
      <app:location>
        <gml:Point srsName="EPSG:4269">
          <gml:pos srsDimension="2">35.89304 -84.195691</gml:pos>
        </gml:Point>
      </app:location>
      <app:datetime>
        <gml:TimePeriod frame="#ISO-8601" gml:id="GML_TD_100067">
          <gml:beginPosition frame="#ISO-8601">2002-06-05T17:00:00.000-07:00</gml:beginPosition>
          <gml:endPosition frame="#ISO-8601">2002-06-05T19:00:00.000-07:00</gml:endPosition>
        </gml:TimePeriod>
      </app:datetime>
    </app:GenericDataSource>
  </gml:featureMember>
</wfs:FeatureCollection>

```

7.1.9 TM_Contains

This test uses the same a similar time interval to 7.1.10 and tests the containment of a Literal value (Time Position).

WFS Request:

```

<?xml version="1.0" encoding="UTF-8"?>
<wfs:GetFeature service="WFS" version="1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc" xmlns:app="http://www.sensors.org/app"
xmlns:abc="http://www.sensors.org/app/abc" xmlns:is="http://www.sensors.org/is"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs
../../../../../../xml/xsd/WFS-1.0.0/WFS_Wrapper.xsd">
  <wfs:Query typeName="app:GenericDataSource">
    <ogc:Filter>
      <ogc:And>
        <ogc:PropertyIsLessThan>
<ogc:PropertyName>app.datetime/gml:TimePeriod/gml:beginPosition</ogc:PropertyName>
          <ogc:Literal>2002-06-06T03:54:06-07:00</ogc:Literal>
        </ogc:PropertyIsLessThan>
        <ogc:PropertyIsGreaterThan>
<ogc:PropertyName>app.datetime/gml:TimePeriod/gml:endPosition</ogc:PropertyName>
          <ogc:Literal>2002-06-06T03:54:06-07:00</ogc:Literal>
        </ogc:PropertyIsGreaterThan>
      </ogc:And>
    </ogc:Filter>

```



```
</wfs:Query>
</wfs:GetFeature>
```

Result: Pass

```
<wfs:FeatureCollection xmlns:gml="http://www.opengis.net/gml"
xmlns:is="http://www.sensors.org/is"
xmlns:abc="http://www.sensors.org/app/abc"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:pr23="http://www.icao.int/IAXSTypes"
xmlns:app="http://www.sensors.org/app"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sensors.org/app test.xsd">
  <gml:boundedBy>
    <gml:Envelope srsName="EPSG:4269">
      <gml:lowerCorner>35.89304 -84.195691</gml:lowerCorner>
      <gml:upperCorner>35.89304 -84.195691</gml:upperCorner>
    </gml:Envelope>
  </gml:boundedBy>
  <gml:featureMember>
    <app:GenericDataSource gml:id="TRP.GALDOS.3-100000014">
      <app:location>
        <gml:Point srsName="EPSG:4269">
          <gml:pos srsDimension="2">35.89304 -84.195691</gml:pos>
        </gml:Point>
      </app:location>
      <app:datetime>
        <gml:TimePeriod frame="#ISO-8601" gml:id="GML_TD_100057">
          <gml:beginPosition frame="#ISO-8601">2002-06-06T02:54:05.000-
07:00</gml:beginPosition>
          <gml:endPosition frame="#ISO-8601">2002-06-06T04:54:06.000-
07:00</gml:endPosition>
        </gml:TimePeriod>
      </app:datetime>
    </app:GenericDataSource>
  </gml:featureMember>
</wfs:FeatureCollection>
```

8 Discussion

In general the comparison operators of WFS 1.0 can be used to realize all of the functionality of the Temporal Operators to be introduced in WFS 1.2. It should be noted, however, that there are a number of deficiencies with this approach and which do indeed warrant the introduction of temporal operators.

8.1 Application Semantics vs Physical Markup

In Section 7.1.1 and 7.1.2 we inserted two different features, each with an application defined “dateTime” property. In Section 7.1.1 we used the gml:begin and gml:end encoding, while in Section 7.1.2 we used the gml:beginPosition and gml:endPosition encoding. Semantically these are equivalent, and are just different syntaxes for expressing the same time interval. If we issue a query using the property name clause value:

```
app:datetime/gml:TimePeriod/gml:beginPosition
```

should we get the same or a different response as when we use the property name clause:

```
app:datetime/gml:TimePeriod/gml:begin/gml:TimeInstant/gml:timePosition
```

From a semantic standpoint we would expect the same result. From an XML standpoint we might expect a different result since these are two different allowed encodings for the property, and in there is no way of knowing the semantics from an inspection of the XML schema. Note that the required semantics can be achieved using only comparison operators, but the request becomes more complex since we must test to see if either version of the encoding (gml:begin, gml:beginPosition) satisfies the temporal constraint.

When we introduce the Temporal operators as proposed in WFS 1.2, this problem goes away, or at least is now hidden from the client, since only the dateTime property appears as an argument of the operation, as in:

```
<ogc:Filter>
  <ogc:TM_After>
    <ogc:PropertyName>app:dateTime</ogc:PropertyName>
    <gml:TimeInstant>
      <gml:timePosition> .. </gml:timePosition>
    </gml:TimeInstant>
  </ogc:TM_After>
</ogc:Filter>
```

It is then perfectly reasonable to require that the implementation of TM_After correctly apply the necessary temporal semantics.

8.2 Clarity of Expression - Complexity

The use of comparison operators to implement temporal comparisons requires that the person or program constructing the WFS requests be aware of the semantic mapping per ISO 19108:2003. If such queries are always constructed programmatically the complexity may be a moot point, however, at some point someone will likely need to read and understand the query, and here the use of the temporal operators is much clearer, and reduces the expression complexity.

8.3 Mapping to Underlying Database Types

The most common time format used in GML is ISO 8601. This format provides a standard time encoding (ISO 8601 : 1988 (E)), however, this encoding provides several different ways of representing the same time. It is unlikely that underlying databases (e.g. Oracle, MS SQL Server, X-Hive) will implement all of the features of ISO 8601, hence compromising interoperability, and placing additional burden on the WFS developer to manage the temporal mappings appropriately. This is made more complex by the fact that there may be multiple temporal casts between the database and the GML serialization (e.g. Java objects, XML data binding etc). For example,

`(1994-11-05T08:15:30-05:00` corresponds to November 5, 1994, 8:15:30 am, US Eastern Standard Time.

`1994-11-05T13:15:30Z` corresponds to the same instant.

In the case of Oracle 10G as the underlying database, the following temporal data types are supported:

- DATE
- TIMESTAMP
- TIMESTAMP WITH TIME ZONE
- TIMESTAMP WITH LOCAL TIME ZONE

Using the DATE Datatype

Use the DATE datatype to store point-in-time values (dates and times) in a table. The DATE datatype stores the century, year, month, day, hours, minutes, and seconds.

Note that this time representation is not appropriate for the representation of `gml:timePosition` (ISO 8601) as it does not incorporate timezone.

Using the TIMESTAMP Datatype

Use the TIMESTAMP datatype to store values that are precise to fractional seconds. For example, an application that must decide which of two events occurred first might use TIMESTAMP. An application that specifies the time for a job might use DATE.

Note that `gml:timePosition` might be mapped to TIMESTAMP or DATE depending on the value of the `gml:frame` attribute. This sort of attribute dependent mapping requires further investigation and was not examined in this IPR.

Using the `TIMESTAMP WITH TIME ZONE` Datatype Because `TIMESTAMP WITH TIME ZONE` can also store time zone information, it is particularly suited for recording date information that must be gathered or coordinated across geographic regions.

This is the most appropriate temporal type for the representation of `gml:timePosition` with the default frame attribute value (8601).

Using the `TIMESTAMP WITH LOCAL TIME ZONE` Datatype Use `TIMESTAMP WITH LOCAL TIME ZONE` when the time zone is not significant. For example, you might use it in an application that schedules teleconferences, where participants each see the start and end times for their own time zone.

The `TIMESTAMP WITH LOCAL TIME ZONE` type is appropriate for two-tier applications in which you want to display dates and times that use the time zone of the client system. It is generally inappropriate in three-tier applications because data displayed in a Web browser is formatted according to the time zone of the Web server, not the time zone of the browser. The Web server is the database client, so its local time is used.

Use of this temporal type was not investigated in this IPR.

8.4 Handling Time Zones in a Distributed Environment

When a `gml:timePosition` property value is captured by a client and submitted through a WFS transaction the time zone and time may be converted so that the time is expressed in the time zone of the WFS to which the transaction is sent.

For most applications this does not present any problems since the time handling is correct in any event. The time zone of the client is however lost in the WFS transaction. For example a WFS located in time zone (UTC-7) which receives a time from a client (located in UTC-4) that is expressed in UTC-4 will be converted to UTC-7 and the time zone of the client will not be visible in the database. Of course the temporal value reported by the client may be in any time zone and not necessarily that of the client.

It may be useful to have the WFS support requests for times in different time zones. The specific grammar for such requests was not, however, investigated.

Bibliography

- [1] OGC 03-105r1, OpenGIS[®] Geography Markup Language (GML) Implementation Specification, Version 3.1.1, April 2004.
- [2] OGC 04-095, Filter Encoding Specification Version 1.1.0, Dec 2004
- [3] OGC 05-093r2, Filter Encoding Specification Change Request Version 1.1.0, 2005
- [4] ISO TC/211 - 19108 : 2003 Temporal Schema, 2003