
Guía de la API del MeteoSIX

Versión v4

MeteoGalicia

30 de abril de 2021

1. Introducción	1
2. Modelos de predicción numérica	3
2.1. WRF (Weather Research Forecast)	4
2.2. WW3 (Wave Watch III)	5
2.3. SWAN (Simulating Waves Nearshore)	5
2.4. ROMS (Regional Ocean Modeling System)	5
2.5. MOHID	6
3. Cuestiones generales	7
3.1. Obtención de la clave para el uso de la API	7
3.2. Sistemas de coordenadas	7
3.3. Mayúsculas y minúsculas	7
3.4. Idiomas	8
3.5. Usos horarios	8
3.6. Formatos temporales	8
3.7. Formatos numéricos	9
3.8. Imágenes	9
3.9. Formatos de las respuestas	9
3.10. Excepciones	9
4. Operaciones	11
4.1. Introducción	11
4.2. Parámetros comunes a las operaciones <i>/getNumericForecastInfo</i> , <i>/getTidesInfo</i> , <i>/getSolarInfo</i>	11
4.3. Rango temporal	13
4.4. Formatos soportados por las operaciones <i>/getNumericForecastInfo</i> , <i>/getTidesInfo</i> y <i>/getSolarInfo</i>	13
4.5. Estructura de los datos devueltos por las operaciones <i>/getNumericForecastInfo</i> , <i>/getTidesInfo</i> y <i>/getSolarInfo</i>	13
4.6. Comportamiento en caso de Error	18
4.7. Comportamiento en caso de ausencia de datos	26
5. Operación <i>/findPlaces</i>	31
5.1. Introducción	31
5.2. Parámetros	31
5.3. Resultados	32
5.4. Ejemplos	35
6. Operación <i>/getNumericForecastInfo</i>	37
6.1. Introducción	37
6.2. Rango temporal	39

6.3. Parámetros.....	39
6.4. Resultados.....	42
6.5. Ejemplos.....	49
7. Operación /getTidesInfo	51
7.1. Introducción.....	51
7.2. Rango temporal.....	53
7.3. Parámetros.....	53
7.4. Resultados.....	54
7.5. Ejemplos.....	58
8. Operación /getSolarInfo	59
8.1. Introducción.....	59
8.2. Rango temporal.....	59
8.3. Parámetros.....	59
8.4. Resultados.....	59
8.5. Observaciones.....	62
8.6. Ejemplos.....	62
9. Excepciones	65
10. A1. Novedades de la versión v4.	67
11. A2. Acerca de este documento.	69
12. A3. Usos horarios.	71

Introducción

La API del MeteoSIX es un servicio web gratuito de libre uso para el acceso a la información meteorológica y oceanográfica. Ofrece información sobre:

- Los resultados de distintos modelos de predicción numérica, meteorológica y oceanográfica, procedentes directamente de las salidas de los modelos, sin supervisión por parte de los expertos.
- Las horas de salida y puesta de sol.
- La predicción de mareas.

Además, para facilitar la consulta, se proporcionan métodos para la localización de algunos tipos de entidades geográficas.

La información de predicción meteorológica y oceanográfica procede de los modelos ejecutados diariamente por MeteoGalicia. Estos se ejecutan sobre diferentes mallas, de distintas resoluciones y coberturas geográficas y temporales (ver *Modelos de predicción numérica*). La información sobre las horas de salida y puesta de sol está disponible para cualquier punto (ver *Operación /getSolarInfo*) mientras que la información sobre mareas se ofrece para la costa gallega (ver *Operación /getTidesInfo*).

La versión actual de la API, a la que se refiere este manual, es la **versión v4 (4.0.0)**. Esta versión es **compatible con** versiones anteriores, aunque requiere la solicitud de una nueva API_KEY.

Modelos de predicción numérica

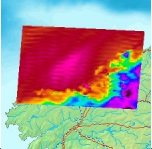
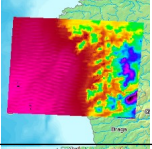
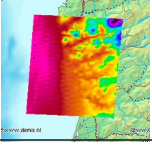

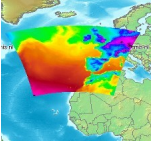
La información de predicción numérica que sirve la API procede directamente de las salidas de los modelos de predicción ejecutados diariamente por MeteoGalicia, sin supervisión por parte de expertos.

Existen distintos modelos, y cada uno se ejecuta sobre diferentes mallas (diferentes zonas de cobertura y resoluciones). Las ejecuciones duran generalmente varias horas, y la hora de finalización varía de un modelo/malla a otro. Para una misma malla, la hora de finalización puede variar de un día a otro, por lo que las horas de finalización que aparecen en las siguientes tablas son aproximadas. Una vez que terminan las ejecuciones, pasan unos minutos hasta que están disponibles a través de la API.


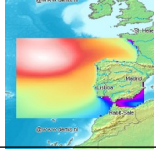
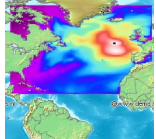
La frecuencia temporal para la que tanto los modelos como la API tienen datos de predicción numérica es de una hora.

los modelos y mallas de los que se sirve actualmente información son los siguientes (las horas están indicadas en UTC -Tempo Universal Coordinado):

2.1 WRF (Weather Research Forecast)

Cobertura	Malla	Resolución	Inicio ejecución	Fin ejecución aproximada	Primera hora predicción	Horizonte predicción
	Artabro 1Km	1km	00:00UTC 12:00UTC	07:30UTC 19:30UTC	01:00UTC 13:00UTC	72h 84h
	Rias Baixas 1Km	1km	00:00UTC 12:00UTC	07:30UTC 19:30UTC	01:00UTC 13:00UTC	72h 84h
	Norte Portugal 1Km	1km	00:00UTC 12:00UTC	07:30UTC 19:30UTC	01:00UTC 13:00UTC	72h 84h
	04km	4km	00:00UTC 12:00UTC	05:00UTC 17:00UTC	01:00UTC 13:00UTC	96h 84h
	12km	12km	00:00UTC 12:00UTC	05:00UTC 17:00UTC	01:00UTC 13:00UTC	96h 84h
	36km	36km	00:00UTC 12:00UTC	05:00UTC 17:00UTC	01:00UTC 13:00UTC	96h 84h

2.2 WW3 (Wave Watch III)

Cobertura	Malla	Resolución	Inicio ejecución	Fin ejecución aproximada	Primera hora predicción	Horizonte predicción
	Galicia	0,05°	00:00UTC 12:00UTC	05:00UTC 17:00UTC	12:00UTC 00:00UTC	109h 97h
	Iberica	0,25°	00:00UTC 12:00UTC	05:00UTC 17:00UTC	12:00UTC 00:00UTC	109h 97h
	AtlanticoNorte	0,5°	00:00UTC 12:00UTC	05:00UTC 17:00UTC	12:00UTC 00:00UTC	109h 97h

2.3 SWAN (Simulating Waves Nearshore)

Cobertura	Malla	Resolución	Inicio ejecución	Fin ejecución aproximada	Primera hora predicción	Horizonte predicción
	Galicia	Variable	00:00UTC	06:30UTC	00:00UTC	97h

2.4 ROMS (Regional Ocean Modeling System)

Cobertura	Malla	Resolución	Inicio ejecución	Fin ejecución aproximada	Primera hora predicción	Horizonte predicción
	Galicia	0,02°	00:00UTC	09:30UTC	00:00UTC	97h

2.5 MOHID

Cobertura	Malla	Resolución	Inicio ejecución	Fin ejecución aproximada	Primera hora predicción	Horizonte predicción
	Artabro	0,003°	00:00UTC	12:30UTC	00:00UTC	49h
	Arousa	0,003°	00:00UTC	12:30UTC	00:00UTC	49h
	Vigo	0,003°	00:00UTC	12:30UTC	00:00UTC	49h

Para obtener más información sobre las características de los distintos modelos de predicción puede consultarse en <https://www.meteogalicia.gal/modelos/index.action>

Cuestiones generales

La API ofrece una serie de operaciones que se ejecutan mediante peticiones *HTTPS GET* y *HTTPS POST* a la URL del servicio (dominio de la API):

`https://servizos.meteogalicia.gal/apiv4/`

La estructura común de todas las peticiones es la siguiente:

`https://servizos.meteogalicia.gal/apiv4/ruta_operación?[parámetros_operación]`

Por ejemplo:

`https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?coords=-8,44&variables=temperature,wind&API_KEY=***`

Cada operación admite distintos tipos de parámetros, que se detallan en las siguientes secciones.

3.1 Obtención de la clave para el uso de la API

El uso de la API es libre y gratuito pero se necesita una clave de usuario que debe ser incluida en todas las peticiones. La clave se puede solicitar siguiendo las instrucciones detalladas [en este apartado](#). (Si se olvida, se puede recuperar usando el mismo procedimiento, con el que recibirá un correo recordatorio).

Esta clave es de uso privado, **única para cada usuario de la API**, y se asocia a la dirección de correo electrónico del usuario. A las peticiones que no incluyen una clave válida se les devuelve una excepción.

Nota: Los usuarios de versiones anteriores de la API deben solicitar obligatoriamente una nueva para usar esta versión v4.

3.2 Sistemas de coordenadas

Algunas operaciones de la API devuelven resultados con geometrías. Del mismo modo, algunas operaciones tienen como parámetro valores geométricos (coordenadas). Actualmente el único sistema de coordenadas soportado para los datos geométricos es el WGS 84 (EPSG:4326).

3.3 Mayúsculas y minúsculas

Todos los parámetros excepto el parámetro `API_KEY` admiten los valores sin diferenciar entre mayúsculas o minúsculas, aunque se recomienda usar los valores tal como se indican en esta guía. Esto también se aplica a los nombres de los parámetros. Así, el parámetro:

`models=WRF, WW3`

es equivalente a:

`MODELS=wrf, ww3`

y también a:

`MoDeLS=wRF, Ww3`

3.4 Idiomas

La API soporta los siguientes idiomas:

- Gallego
- Español
- Inglés

El idioma por defecto es el inglés.

3.5 Usos horarios

La API permite especificar el uso horario en el que se quiere obtener los datos. Si en la petición no se especifica el uso, los datos se devuelven en el uso horario de Galicia, es decir, en UTC+1 en horario de invierno y UTC+2 en horario de verano.¹

En el apéndice *A3. Usos horarios* figuran los usos horarios permitidos.

3.6 Formatos temporales

El formato usado para las fechas y horas devueltas por la API es el siguiente:

`yyyy-MM-ddTHH:mm:ssZZ`

donde:

- *yyyy*: año.
- *MM*: mes.
- *dd*: día del mes.
- *HH*: hora (en formato de 24 horas).
- *mm*: minuto.
- *ss*: segundo.
- *ZZ*: desviación respecto a la hora UTC. Los caracteres *ZZ* se sustituirán por un código que representa la desviación de la hora respecto a la hora UTC. Por ejemplo, para un instante en hora local de Galicia, en horario de verano, los caracteres *ZZ* se sustituyen por el código +02; para un instante en hora local de Galicia, en horario de invierno, se sustituyen por el código +01. Para instantes en UTC, los caracteres *ZZ* se sustituyen por el código Z.

Ejemplos:

¿UTC es o TempoUniversalCoordinado (véxase http://es.wikipedia.org/wiki/Tiempo_universal_coordinado).

2011-10-24T05:00:00+02 : hora local de Galicia en verano.
2011-10-24T05:00:00+01 : hora local de Galicia en invierno.
2011-10-24T05:00:00Z : hora UTC.

Con respecto a los parámetros de las peticiones con valores temporales, el formato a emplear es:

yyyy-MM-ddTHH:mm:ss

Es el mismo formato, pero sin la diferencia horaria con respecto a la hora UTC.

3.7 Formatos numéricos

El separador decimal es siempre el punto, ".", independientemente del idioma.

3.8 Imágenes

Para algunas variables, como el viento o el estado del cielo, la API devuelve, además de los valores numéricos, referencias (URLs) a iconos de meteogramas que representan los valores de esas variables. Todas estas imágenes están en formato PNG y tienen un tamaño de 24x24 píxeles.

3.9 Formatos de las respuestas

Dependiendo del tipo de operación y del resultado, los formatos soportados son los siguientes:

- *GeoJSON*²
- *GML*
- *3.2.1HTML*
- *KML2.0*

El formato deseado se indica a través del correspondiente parámetro en la petición. Si no se indica, la respuesta se devolverá en el formato por defecto.

3.10 Excepciones

En caso de Error, la API devolverá información sobre este en la respuesta, en el formato de Error especificado en la petición o en el formato de Error por defecto. Los formatos soportados para las excepciones son:

- *JSON*²
- *XML*

² Aunque el MIME-TYPE que se declara es el de JSON (GeoJSON carece de MIME-TYPE específico), el formato usado es GeoJSON, que añade a JSON soporte para la información geográfica. En este documento se hablará normalmente de GeoJSON, aunque a veces aparecerá JSON.

4.1 Introducción

La API v4 dispone de cuatro operaciones:

- *Operación /findPlaces:*
Permite buscar lugares por su nombre.
- *Operación /getNumericForecastInfo:*
Devuelve información de predicción numérica (atmosférica y oceanográfica).
- *Operación /getTidesInfo:*
Devuelve información de mareas.
- *Operación /getSolarInfo:*
Devuelve información de las horas de salida y puesta de sol.

As operaciones */getNumericForecastInfo*, */getTidesInfo* y */getSolarInfo* comparten varias características comunes, como los parámetros de los formatos de salida. Se describen a continuación.

4.2 Parámetros comunes a las operaciones */getNumericForecastInfo*, */getTidesInfo*, */getSolarInfo*

Los parámetros comunes a estas operaciones son:

Nombre	Obligatorio	Posibles valores	Valor por defecto	Observaciones
API_KEY	SI	Clave de la API	-	Ver la sección <i>Obtención de la clave para el uso de la API</i> .
locationIds ¹	NON ²	Identificadores de lugar	-	Lista de identificadores de lugares sobre los que se desea obtener datos, separados por comas (ver <i>Operación /findPlaces</i>). Si no existen lugares con esos identificadores se devuelve una excepción. Actualmente, el límite son 20 localizaciones, en caso de indicar se devuelve una excepción.
coords ³	NO ²	ya,ya;xb,yb;... (ej: -8.32,44.5;-8.11,43.88)	-	Lista de coordenadas de los lugares sobre los que se desea obtener datos. Cada par de coordenadas se separa del siguiente por punto y coma. En cada par, a longitud y a latitud se separan por coma. El orden es longitud,latitud. Actualmente, el límite son 20 pares de coordenadas, en caso de indicar más se devuelve una excepción
startTime	NO	<i>Instante temporal</i> , con formato yyyy-MM-ddTHH:mm:ss	Instante actual	Primer instante temporal para el que se quiere información. Ver Rango temporal.
endTime	NO	<i>Instante temporal</i> , con formato yyyy-MM-ddTHH:mm:ss	-	Último instante temporal para el que se quiere información. Si no se indica, se devuelve hasta el último instante disponible, dentro de los límites establecidos para cada petición. Ver Rango temporal.
lang	NO	'gl','es','en'	'en'	Idioma en el que se devolverán los textos, incluyendo los textos de las excepciones.
tz	NO	<i>Identificador</i> del uso horario (ver Usos horarios)	'Europe/Madrid'	Uso horario en el que se mostrarán las horas.
CRS	NO	Identificador del sistema de coordenadas	'EPSG:4326'	<i>Actualmente</i> el único posible valor es EPSG:4326.
format	NO	'gml3' 'kml' 'application/json' 'text/html'	'application/json'	Formato en el que se devolverán los resultados.
style	NO	Nombre de estilo a aplicar en las respuestas con formato HTML	'default'	Sólo se debe <i>indicar</i> cuando el valor del parámetro format es text/html. Actualmente, el único posible valor es <i>default</i> .
exceptionsFormat	NO	'application/json' 'application/xml'	'application/json'	Formato en el que se devolverán las Excepciones.

¹ En la versión v2 de la API este parámetro se llamaba *locationId*.

² Uno y sólo un de los dos parámetros, *locationIds* o *coords*, tiene que estar presente.

³ En la versión v2 de la API este parámetro se llamaba *coord*.

Existen, por tanto, dos formas de indicar los puntos sobre los que se quiere realizar una consulta:

- A partir de un conjunto de pares de coordenadas x, y.
- A partir de un conjunto de *ides* de lugares obtenidos previamente mediante la operación `/findPlaces`.

4.3 Rango temporal

Los parámetros `startTime` y `endTime` indican el rango temporal para el que se quiere obtener los datos. Ambos deben tener el formato `yyyy-MM-ddTHH:mm:ss`, pero en algunas operaciones sólo se tiene en cuenta el día y no la hora. En las secciones dedicadas a cada operación se indica si se tiene en cuenta o no la hora.

En los casos en los que sí se tiene en cuenta la hora, únicamente se devuelven los datos comprendidos entre el `startTime` y el `endTime`. Así por ejemplo, si el `startTime` tuviese el valor `2014-05-06T07:01:00`, no se devolverían los valores correspondientes a las `07:00:00` del día 6, con lo cual, en el caso de la predicción numérica, el primer valor que se devolvería correspondería a las `08:00:00` de ese día (ya que sería el primer instante con datos posterior o igual a las `07:01:00`).

La cota inferior para el rango temporal de las peticiones es el primer instante del día actual (día en el que se hace petición). No existe una cota superior, pero para cada operación existe un límite en el número total de días que se pueden pedir (a contar desde el primer día que se pide, no desde el día actual; ver los apartados correspondientes a cada operación). Si `startTime` o `endTime` son anteriores al primer instante del día actual se devuelve una excepción. Si `endTime` es anterior al `startTime` también se devuelve una excepción. Si `endTime` es anterior al instante actual, es obligatorio especificar el `startTime` en la petición (salvo en las operaciones en las que no se tiene en cuenta la hora).

Si no se indica el parámetro `startTime`, se devuelven los datos disponibles desde el instante actual, bien hasta el `endTime` (si este se indicó en la petición), o bien hasta un número de días por defecto que depende de cada operación.

4.4 Formatos soportados por las operaciones

/getNumericForecastInfo, */getTidesInfo* y */getSolarInfo*

- **GML**: devuelve los datos en el formato GML, versión 3.2.1.
- **HTML**: devuelve un documento HTML con los datos pedidos. Incluye referencias a ficheros CSS e iconos en PNG.
- **JSON**: devuelve los datos en formato GeoJSON.
- **KML**: devuelve los datos en el formato KML 2.0. La información está contenida en el elemento *description* en forma de HTML (con el mismo contenido que cuando se usa el formato HTML).

4.5 Estructura de los datos devueltos por las operaciones

/getNumericForecastInfo, */getTidesInfo* y */getSolarInfo*

Los datos devueltos por estas operaciones se estructuran de la siguiente forma:

- **Localizaciones**: se devuelven datos para una o varias localizaciones.
 - **Días**: cada localización contiene información para un o varios días. Para cada día puede no devolverse información sobre el día completo, si no sobre un subintervalo de tiempo contenido en el día:
 - **Variables**: dentro de cada día, La información se organiza por variables
 - ◊ **Valores**: en cada variable, se incluyen los valores propiamente dichos, ordenados por los instantes temporales con los que se corresponden.

Para representar o conjunto de localizaciones se usan las siguientes estructuras de datos :

Formato	Representación
GML	Se emplea un elemento <i>gml:featureCollection</i> con tantos elementos <i>gml:featureMember</i> como localizaciones. Cada <i>gml:featureMember</i> contiene un elemento <i>location</i> que a su vez contiene toda la información.
HTML	La información se muestra en tablas HTML.
JSON	Se emplea un objeto <i>FeatureCollection</i> con tantos objetos <i>feature</i> como localizaciones.
KML	Para cada localización se añade un elemento <i>Placemark</i> dentro del <i>Document</i> .

Para las geometrías (coordenadas) de cada localización se usan las siguientes representaciones:

Formato	Representación
GML	Se representa con un elemento <i>gml:point</i> contenido en un elemento <i>geometry</i> .
HTML	Se representa como texto dentro del HTML.
JSON	Se representa con un objeto <i>geometry</i> de tipo punto (“ <i>type</i> ”: “ <i>Point</i> ”).
KML	Se representa con un elemento <i>Point</i> .

Cuando la petición contiene el parámetro *locationIds*, en la respuesta se incluyen los siguientes atributos en cada localización válida solicitada: id, nombre, municipio, provincia y tipo de lugar:

Formato	Representación
GML	Se representan como atributos del elemento <i>location</i> de cada <i>gml:featureMember</i> (<i>id</i> , <i>name</i> , <i>municipality</i> , <i>province</i> y <i>type</i>).
HTML	Se representa como texto dentro del HTML.
JSON	Se representan como atributos del objeto <i>properties</i> de cada <i>feature</i> (<i>id</i> , <i>name</i> , <i>municipality</i> , <i>province</i> y <i>type</i>).
KML	Cada atributo se representa como un elemento <i>Data</i> dentro del elemento <i>ExtendedData</i> del <i>Placemark</i> con los siguientes nombres (<i>name</i>): <i>location_id</i> , <i>location_name</i> , <i>location_municipality</i> , <i>location_province</i> , <i>location_type</i> .

Cada localización contiene un conjunto de uno o varios días (*days*)⁴. En HTML y KML la información de cada día se muestra en tablas HTML. En JSON y GML se indica el rango temporal para el que se están devolviendo datos dentro de cada día. Esto quiere decir que se devolvieron los datos disponibles para ese rango pero:

- Para parte de los instantes del rango temporal puede no haber datos disponibles, caso en el que el valor del instante correspondiente es nulo o vacío.
- Para la parte del día que queda fuera del rango temporal podría haber datos disponibles que no se estén devolviendo. Esto sucede, por ejemplo, cuando quedan fuera del rango temporal indicado en la petición o, se no se indicó rango temporal en la petición, se corresponden a instantes anteriores al instante temporal *actual*.

El rango temporal que cobre cada *day* se especifica del siguiente modo:

Formato	Representación
GML	Se usa un elemento <i>gml:TimePeriod</i> que contiene un elemento <i>gml:begin</i> y un elemento <i>gml:end</i> con los instantes inicial y final del período, respectivamente.
JSON	Se usa un elemento “ <i>timePeriod</i> ” que contiene objetos <i>begin</i> y <i>end</i> con los instantes inicial y final del período, respectivamente.

En ambos casos, los instantes temporales tendrán el formato *yyyy-MM-ddTHH:mm:ss+XX*, según se indica en la sección *Formatos temporales*. La información de cada día se agrupa en una o varias variables contenidas en un elemento el objeto variables.

Más en detalle para cada formato de respuesta:

- **JSON:**

Se devuelve un objeto GeoJSON con la siguiente estructura⁴:

```
{
  "type": "FeatureCollection",
  "crs": {
    "type": "name",
    "properties": {
      "name": CRS
    }
  },
  "features": [
    ...
    {
      "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates": [
          X,
          Y
        ]
      },
      "properties": {
        "id": ID,
        "name": NAME,
        "municipality": MUNICIPALITY,
        "province": PROVINCE,
        "type": TYPE,
        "days": DAYS_ARRAY
      }
    }
    ...
  ]
}
```

Donde:

- CRS indica el sistema de coordenadas usado (actualmente es siempre urn:ogc:def:crs:OGC:1.3:CRS84).
- X y Y son las coordenadas de la localización (longitud y latitud respectivamente).
- ID, NAME, MUNICIPALITY, PROVINCE y TYPE son atributos de la localización. Sólo aparecen cuando la operación se invoca con el parámetro locationIds.
- DAYS_ARRAY es un array JSON que contiene tantos elementos como días para los que se devuelve la información. Cada día tiene la siguiente estructura:

```
{
  "timePeriod":
    { "begin": {
      "timeInstant": START_DATE
    },
      "end": {
        "timeInstant": END_DATE
      }
    },
  "variables": VARIABLES_ARRAY
}
```

Donde:

- ◻ START_DATE y END_DATE representan, respectivamente, el inicio y a fin del rango temporal para el

que se están devolviendo datos dentro del día.

◻ VARIABLES_ARRAY es un array JSON de objetos variable. La estructura del objeto variable varía para cada operación concreta y se detalla en las secciones dedicadas a cada operación

■ GML:

Se devuelve un documento GML con la siguiente Estructura⁴:

```
<?xmlversion="1.0"encoding="UTF-8"?>
<gml:FeatureCollectionxmlns="https://www.meteogalicia.gal/meteosix"xmlns:gml="http://www.opengis
  <gml:boundedBy>
    <gml:EnvelopeName=CRS>
      <gml:pos>MIN_YMIN_X</gml:pos>
      <gml:pos>MAX_YMAX_X</gml:pos>
    </gml:Envelope>
  </gml:boundedBy>
  ...
  <gml:featureMember>
    <locationid=IDname=NAMEmunicipality=MUNICIPALITYprovince=PROVINCEtype=TYPE>
      <geometry>
        <gml:PointName=CRS>
          <gml:pos>
            YX
          </gml:pos>
        </gml:Point>
      </geometry>
      <days>
        DAY_1DAY_2
        ...
      </days>
    </location>
  </gml:featureMember>
  ...
</gml:FeatureCollection>
```

Donde:

- CRS indica el sistema de coordenadas usado (actualmente es siempre EPSG:4326).
- MIN_Y, MIN_X, MAX_Y, MAX_X: son las coordenadas mínimas y máximas del Envelope del conjunto de todas las localizaciones. El orden es latitud, longitud.
- ID, NAME, MUNICIPALITY, PROVINCE y TYPE son atributos de la localización. Sólo aparecen cuando la operación se invoca con el parámetro locationIds
- X y Y son las coordenadas de la localización (longitud y latitud respectivamente).
- DAY_1, DAY_2... son elementos de tipo day. Cada día tiene la siguiente estructura:

```
<day>
  <gml:TimePeriod>
    <gml:begin>
      <gml:TimeInstant>
        <gml:timePosition>START
          _DATE
        </gml:timePosition>
      </gml:TimeInstant>
    </gml:begin>
```

⁴ En el caso de la operación /getTidesInfo, existen más elementos dentro del objeto properties (JSON) y del elemento location (GML).

```

    <gml:end>
      <gml:TimeInstant>
        <gml:timePosition>END_DATE
          ATE
        </gml:timePosition>
      </gml:TimeInstant>
    </gml:end>
  </gml:TimePeriod>
</variables>
  VARIABLE1
  VARIABLE2
  ...
</variables>
</day>

```

Donde:

- START_DATE y END_DATE representan, respectivamente, el inicio y a fin del rango temporal para el que se están devolviendo datos dentro del día.
- VARIABLE 1, VARIABLE 2... son elementos de tipo variable. La estructura de un elemento variable, varía, nuevamente, para cada operación concreta y se detalla en las secciones dedicadas a cada operación.

■ **HTML:**

La respuesta es un documento HTML con el correspondiente CSS. Por ejemplo, para el caso de predicción numérica tendría el siguiente aspecto:

Predicción para latitude=44, lonxitude=-8

Luns abril 29, 2013

	Resumo	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
Estado do ceo		-	-									
Vento	Min Max 15 53	-	-	 19	 19	 18	 17	 15	 15	 17	 17	 16

Martes abril 30, 2013

	Resumo	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
Estado do ceo												
Vento	Min Max 46 51	 50	 49	 51	 49	 50	 50	 50	 49	 48	 48	 49

Mércores maio 01, 2013

	Resumo	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
Estado do ceo												
Vento	Min Max 35 51	 51	 50	 49	 48	 46	 45	 44	 44	 44	 44	 43

■ **KML:**

La respuesta es un documento KML con la siguiente estructura:

```

<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://earth.google.com/kml/2.0">

```

<Document>

```

<Placemarkid="point_1">
  <ExtendedData>
    <Dataname="location_id">
      <value>ID</value>
    </Data>
    <Dataname="location_name">
      <value>NAME</value>
    </Data>
    <Dataname="location_province">
      <value>PROVINCE</value>
    </Data>
    <Dataname="location_municipality">
      <value>MUNICIPALITY</value>
    </Data>
    <Dataname="location_type">
      <value>TYPE</value>
    </Data>
  </ExtendedData>
  <description>DESCRIPTION
</description>
  <Point>
    <coordinates>X,Y</coordinates>
  </Point>
</Placemark>
...
</Document>
</kml>

```

Donde:

- ID, NAME, MUNICIPALITY, PROVINCE y TYPE son atributos de la localización. Sólo aparecen cuando la operación se invoca con el parámetro locationIds
- DESCRIPTION es un elemento de tipo description que contiene un documento HTML igual que o que se obtendría se el formato de respuesta fose HTML. El HTML va dentro de un bloque CDATA
- X y Y son las coordenadas de la localización (longitud y latitud respectivamente)

Cada Placemark tiene un atributo id con valores point_1, point_2... y así sucesivamente para cada Placemark contenido en la respuesta.

4.6 Comportamiento en caso de Error

En el caso de que se produzca un Error o excepción y no se puedan devolver todos los datos, se diferencian entre dos casos:

- **Errorres que afectan a todas las localizaciones solicitadas:** en este caso la respuesta es un mensaje de excepción en el formato de Excepciones por defecto o en el formato de Excepciones indicado en la petición. Ver Comportamiento en caso de Error global.
- **Errorres que afectan a localizaciones concretas:** en este caso, se devuelven los datos para as localizaciones que no están afectadas polos Errorres, y para aquellas que si o estén se devuelve un mensaje de excepción. La respuesta estará en el formato de datos por defecto o en el formato indicado en la petición (no en el formatel para Excepciones).En los bloques correspondientes a las localizaciones afectadas por Errorres se incluirá un mensaje de excepción. Ver *Comportamiento en caso de Error para algunhas localizációs.*

Si todas las localizaciones están afectadas por Errores, pero se trata de Errores distintos y no de un Error común a todas, se seguirá o indicado en el segundo caso.

4.6.1 Comportamiento en caso de Error global

Cuando se produce un Error que afecta a todas las localizaciones para las que se solicitó información, se devolverá un mensaje de excepción. El formato usado será el formato de Excepciones por defecto o bien el formato de Excepciones indicado en el correspondiente parámetro de la petición. En todos los formatos el mensaje contiene:

- Un código de excepción
- un mensaje descriptivo.

En el Apéndice *Excepciones* se listan los tipos de Excepciones contemplados. La estructura de los mensajes de excepción es:

- **JSON:**

```
{
  "exception":
    { "code": "CODE",
      "message": "MESSAGE"
    }
}
```

- **XML:**

```
<?xmlversion="1.0"encoding="UTF-8"?>
<Exceptioncode="CODE">
  <message>MESSAGE</message>
</Exception>
```

4.6.2 Comportamiento en caso de Error para algunas localizaciones

Cuando se produzcan Errores que afecten a una o varias localizaciones, pero no de un mismo Error común a todas ellas, se devolverán los datos para las localizaciones que no estén afectadas por los Errores, y para aquellas que sí lo estén se devolverá un mensaje de excepción. La respuesta seguirá el formato de datos por defecto o el formato de datos indicado en la petición (no el formato para Excepciones). En los bloques correspondientes a las localizaciones afectadas por Errores irá un mensaje de excepción en lugar de datos. La estructura de los mensajes de excepción para una localización concreta es la siguiente:

- **GML:**

En el caso del formato GML, en las localizaciones en las que se produzca un Error, el mensaje de Error se incluye dentro del bloque *location* de los *featureMember* correspondientes, en elementos *exception* con la siguiente forma:

```
<?xmlversion="1.0"encoding="UTF-8"?>
<gml:FeatureCollectionxmlns="https://www.meteogalicia.gal/
meteosix"xmlns:gml="http://www.opengis.net/gml">
  ...
  <gml:featureMember>
    <location>
      ...
      <exceptioncode="CODE"value="MESSAGE"/>
      ...
    </location>
  </gml:featureMember>
  ...
</gml:FeatureCollection>
```

El elemento *geometry* se incluye siempre. Si la localización no se corresponde con unas coordenadas válidas, su valor es nulo:

```
<geometry>
  <gml:null>missing</gml:null>
</geometry>
```

El elemento *days* también se incluye, en este caso sin contenido:

```
<days/>
```

en las peticiones en las que se use el parámetro *locationIds*, en la respuesta GML se incluirán los atributos del elemento *location* que sea posible obtener (*id*, *name*...). Si todas las localizaciones están afectadas por Errores (pero no por un Error común a todas, caso en el que se procedería según el *Comportamiento en caso de Error global*) y ninguna tiene coordenadas válidas, el elemento *gml:boundedBy* será nulo:

```
<gml:boundedBy>
  <gml:null>missing</gml:null>
</gml:boundedBy>
```

Por ejemplo, la respuesta a la petición

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?
locationIds=aa,33,71941,1002197&variables=sea_water_temperature
,significant_wave_height&lang=gl&format=gml3&API_KEY=*
**
```

es:

```
<?xmlversion="1.0"encoding="UTF-8"?>
<gml:FeatureCollectionxmlns:gml="http://www.opengis.net/
gml"xmlns="https://www.meteogalicia.gal/meteosix">
  <gml:boundedBy>
    <gml:EnvelopeSrsName="EPSG:4326">
      <gml:pos>42.52197-8.85002</gml:pos>
      <gml:pos>42.6397-7.511</gml:pos>
    </gml:Envelope>
  </gml:boundedBy>
  <gml:featureMember>
    <locationid="aa">
      <geometry>
        <gml:null>missing</gml:null>
      </geometry>
      <days/>
      <exceptioncode="207"message="
Ovalor'aa'nonéunidificadorde lugar válido."/>
    </location>
  </gml:featureMember>
  <gml:featureMember>
    <locationid="33">
      <geometry>
        <gml:null>missing</gml:null>
      </geometry>
      <days/>
      <exceptioncode="210"
message="Nonseencontróningún lugar conidigual a33"/>
    </location>
  </gml:featureMember>
  <gml:featureMember>
    <locationid="71941"name="MonfortedeLemos"
```



```

municipality="MONFORTE DE
LE MOS" province="Lugo" type="locality">
  <geometry>
    <gml:Point srsName="EPSG:4326">
      <gml:pos>42.52197-7.511</gml:pos>
    </gml:Point>
  </geometry>
</days>
<exceptioncode="216"
message="El punto indicado cae fuera de los límites geográficos para los cuales hay
datos."/>
</location>
</gml:featureMember>
<gml:featureMember>
  <locationid="1002197" name="Praia de
Suaigrexa" municipality="BOIRO" province="A
Coruña" type="beach">
    <geometry>
      <gml:Point srsName="EPSG:4326">
        <gml:pos>42.6397-8.85002</gml:pos>
      </gml:Point>
    </geometry>
  <days>
    ...
  </days>
</location>
</gml:featureMember>
</gml:FeatureCollection>

```

- **HTML:**

En el caso de HTML, en las localizaciones en las que se produzca Error, se incluirá la información sobre la excepción en la posición del HTML correspondiente a dichas localizaciones. Por ejemplo, la respuesta a la petición:

```

https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?coords=44,-9;aa;-
8.393145,43.437223&models=WW3,WW3,SWAN,WRF&variables=significant_wave_height,significant_wave_h
eight,significant_wave_height,temperature&grids=Galicia,Iberica,Artabro,Artabro1Km&lang=gl&f
ormat=text/html&API_KEY=***

```

es:

Predición para latitude=-9, lonxitude=44:
 O punto indicado cae fóra dos límites xeográficos para os cales hai datos.

O valor 'aa' non son unhas coordenadas válidas.

Predición para latitude=43.43722, lonxitude=-8.39314^A

Lui

	Resumo	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h
Temperatura	Min Max	-	-	-	-	-	-	-	-	-	-
	13 14	-	-	-	-	-	-	-	-	-	-
Altura significativa de onda	Min Max	-	-	-	-	-	-	-	-	-	-
	2 3	-	-	-	-	-	-	-	-	-	-
	2 2	-	-	-	-	-	-	-	-	-	-
	Min Max	-	-	-	-	-	-	-	-	-	-
	2 3	-	-	-	-	-	-	-	-	-	-

Ma

	Resumo	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h
Temperatura	Min Max	13	13	13	14	14	14	14	14	14	14
	13 14	13	13	13	14	14	14	14	14	14	14
Altura significativa de onda	Min Max	2	2	2	2	2	2	2	2	2	2
	2 2	2	2	2	2	2	2	2	2	2	2
	2 2	2	2	2	2	2	2	2	2	2	2
	Min Max	2	2	2	2	2	2	2	2	2	2
	2 3	2	2	2	2	2	2	2	2	2	2

Mé

Se trata de una petición sobre tres localizaciones. Para dos de ellas – 44,-9 y aa – se devuelven mensajes de excepción y para la otra – -8.393145,43.4372239 – se devuelven datos.

■ **JSON:**

En el caso de JSON, en las localizaciones en las que se produzca Error, la información sobre el Error se describe en el objeto JSON *exception* dentro del objeto *Feature* correspondiente. Como se indicó anteriormente, el objeto JSON *exception* consta de dos elementos, *code* y *message*. El elemento *geometry* se incluye siempre. Si la localización no se corresponde con unas coordenadas válidas, su valor es nulo:

`geometry: null`

El elemento *days* también se incluye, en este caso con valor nulo:

`days:null`

en las peticiones que contengan el parámetro *locationIds*, en la respuesta se incluyen los atributos *id*, *name*, *municipality*... (aquellos que no sea posible obtener tendrán el valor null). Por ejemplo, la respuesta a la petición:

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?
locationIds=33,71941,1002197&variables=sea_water_temperature,significant_wave_height&lang=gl&format=application/json&API_KEY=***
```

es:

```
{
  type:"FeatureCollection",
  rs:{
    type:"name",
```

```

    properties:
      {name:"urn:ogc:def:crs:OGC:1.3:CRS84"}
  },
  features:[
    {
      type:"Feature",
      geometry: null,
      properties:{
        id:"33",
        name:null,
        municipality:null,
        province:null,
        type:null,
        days:null
      },
      exception:{
        code:"210",
        message:"No se encontró ningún
        lugar con id igual a 33"
      }
    },
    {
      type:"Feature",
      geometry:{
        type:"Point",
        coordinates:[
          -7.511,
          42.52197
        ]
      },
      properties:
        {id:"71941",
        name:"Monforte de Lemos",
        municipality:"MONFORTEDELEMOS",
        province:"Lugo",
        type:"locality",
        days:null
        },
      exception:
        {code:"216",
        message:"
        El punto indicado cae fuera de los límites geográficos para los cales hay datos."
        }
    },
    {
      type:"Feature",
      geometry:{
        type:"Point", coordinates
        :[
          -8.85002,
          42.6397
        ]
      },
      properties:
        {id:"1002197",
        name:"Praia de Suaigrexa",
        municipality:"BOIRO",

```

```

        province:"ACoruña", ty
        pe:"beach",
        days:[
            ...
        ]
    }
}
]
}

```

- **KML:**

En el caso de KML, en las localizaciones en las que se produzca Error, el mensaje de Error se incluye dentro del bloque *ExtendedData* de los *Placemark* correspondientes, en elementos *Data* con *name* igual a *exceptionCode* y *exceptionMessage*:

```

<?xmlversion="1.0"encoding="UTF-8"?>
<kmlxmlns="http://earth.google.com/kml/2.0">
  <Document>
    <Placemarkid="point_1">
      <ExtendedData>
        ...
        <Dataname="exceptionCode">
          <value>CODE</value>
        </Data>
        <Dataname="exceptionMessage">
          <value>MESSAGE</value>
        </Data>
        ...
      </ExtendedData>
      ...
    </Placemarkid="point_2">
      ...
    </Placemark>
    ...
  </Document>
</kml>

```

Si el Error no se debe a que las coordenadas del punto no son válidas, se incluye también el elemento *Point*. En las peticiones que contengan el parámetro *locationIds*, en la respuesta también se incluyen los elementos *Dataparalocation_id*, *location_name*... que sea posible obtener..Por ejemplo, la respuesta a la petición:

```

https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?
locationIds=33,71941,1002197&variables=sea_water_temperature,significant_wave_height&lang
=gl&format=kml&API_KEY=***

```

es:

```

<?xmlversion="1.0"encoding="UTF-8"?>
<kmlxmlns="http://earth.google.com/kml/2.0">
  <Document>
    <Placemarkid="point_1">
      <ExtendedData>
        <Dataname="location_id">
          <value>33</value>
        </Data>
        <Dataname="exceptionCode">
          <value>210</value>
        </Data>

```

```

        <Dataname="exceptionMessage">
            <value>No se encontró ningún lugar con id igual a 33</value>
        </Data>
    </ExtendedData>
</Placemark>
<Placemarkid="point_2">
    <ExtendedData>
        <Dataname="location_id">
            <value>71941</value>
        </Data>
        <Dataname="location_name">
            <value>MonfortedeLemos</value>
        </Data>
        <Dataname="location_municipality">
            <value>MONFORTEDELEMONS</value>
        </Data>
        <Dataname="location_province">
            <value>Lugo</value>
        </Data>
        <Dataname="location_type">
            <value>locality</value>
        </Data>
        <Dataname="exceptionCode">
            <value>216</value>
        </Data>
        <Dataname="exceptionMessage">
            <value>El punto indicado cae fuera de los límites geográficos para los cales hay
            datos.
            </value>
        </Data>
    </ExtendedData>
    <Point>
        <coordinates>-7.511,42.52197</coordinates>
    </Point>
</Placemark>
<Placemarkid="point_3">
    <ExtendedData>
        <Dataname="location_id">
            <value>1002197</value>
        </Data>
        <Dataname="location_name">
            <value>Praia de Suaigrexa</value>
        </Data>
        <Dataname="location_municipality">
            <value>BOIRO</value>
        </Data>
        <Dataname="location_province">
            <value>ACoruña</value>
        </Data>
        <Dataname="location_type">
            <value>beach</value>
        </Data>
    </ExtendedData>
    <description>
        ...
    </description>
    <Point>
        <coordinates>-8.85002,42.6397</coordinates>
    </Point>

```

```
</Placemark>
</Document>
</kml>
```

4.7 Comportamiento en caso de ausencia de datos

En este apartado se describe el comportamiento en los casos en los que no se pueden devolver todos los datos solicitados. A falta de datos puede darse a tres niveles:

- Días para los que no se puede devolver ningún dato.
- Variables para las que no se puede devolver ningún dato dentro de alguno de los días.
- Instantes horarios, contenidos en el rango temporal declarado para el día, para los que no hay datos.

A continuación se indica cómo se gestionan estos casos en cada formato:

- **JSON:**

Si no existen datos para ninguno de los días solicitados se devuelve una excepción, salvo en el caso de que se pidieran datos de más de una localización. En ese caso, se devuelve un mensaje de excepción dentro de cada localización sin datos. Por ejemplo:

```
{
  "type": "FeatureCollection",
  "crs": {
    "type": "name",
    "properties": {
      "name": "urn:ogc:def:crs:OGC:1.3:CRS84"
    }
  },
  "features": [
    {
      "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates": [
          -8.29504,
          43.47569
        ]
      },
      "properties": null,
      "exception": {
        "code": "216", "message": "
        El punto indicado cae fuera de los límites geográficos para los cuales hay datos."
      }
    },
    {
      "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates": [
          -8.19505,
          43.47569
        ]
      },
      "properties": {
```

```

    "days": [
      {
        "timePeriod":
          {"begin":{
            "timeInstant":"2014-05-07T14:46:31+02"
          },
        ...

```

Si no existen datos para algún de los días solicitados pero si para otros, o comportamiento es el siguiente

- Para cada día sin datos comprendido entre el instante actual (o en su caso el startDate) y el último día con datos, se devuelve un objeto con la siguiente estructura:

```

{
  "timePeriod":
    {"begin":{
      "timeInstant":"yyyy-MM-ddTHH:mm:ss"
    },
    "end":{
      "timeInstant":"yyyy-MM-ddTHH:mm:ss"
    }
  },
  "variables":null
}

```

- Si se invocó la operación especificando el parámetro endTime, para los días sin datos comprendidos entre el último día con datos y el endTime, se devuelve un objeto con la Estructura anterior.
- Si se invocó la operación sin especificar el parámetro endTime, los días sin datos posteriores al último día con datos no figuran en la respuesta

Cuando una variable no tiene datos para ningún instante horario dentro del rango temporal del objeto *day*, el objeto *values* será nulo. La estructura del objeto *day* será la siguiente:

```

{
  "timePeriod":{"begin":{
    "timeInstant":"yyyy-MM-ddTHH:mm:ss"
  },
  "end":{
    "timeInstant":"yyyy-MM-ddTHH:mm:ss"
  }
},
  "variables":[
    ...
    {
      "name":VARIABLE_NAME,
      ...
      "values":null,
      ...
    }
    ...
  ]
}

```

El caso de ausencia de datos para un instante temporal concreto se describe en las secciones dedicadas a cada operación.

■ **GML:**

Si no existen datos para ninguno de los días solicitados se devuelve una excepción, salvo en el caso de que se pidieran datos de más de una localización. En ese caso, se devuelve un mensaje de excepción dentro de cada localización sin datos. Por ejemplo

```

<?xmlversion="1.0"encoding="UTF-8"?>
<gml:FeatureCollectionxmlns="https://www.meteogalicia.gal/
meteosix"xmlns:gml="http://www.opengis.net/gml">
  <gml:boundedBy>
    <gml:EnvelopeSrsName="EPSG:4326">
      <gml:pos>43.47569-8.29504</gml:pos>
      <gml:pos>43.47569-8.19505</gml:pos>
    </gml:Envelope>
  </gml:boundedBy>
  <gml:featureMember>
    <location>
      <geometry>
        <gml:PointsrsName="EPSG:4326">
          <gml:pos>43.47569-8.29504</gml:pos>
        </gml:Point>
      </geometry>
      <days/>
      <exceptioncode="216"
        message="El punto indicado cae fuera de los límites geográficos para los cales hay
        datos."/>
    </location>
  </gml:featureMember>
  <gml:featureMember>
    <location>
      <geometry>
        <gml:PointsrsName="EPSG:4326">
          <gml:pos>43.47569-8.19505</gml:pos>
        </gml:Point>
      </geometry>
      <days>
        <day>
          <gml:TimePeriod>
            <gml:begin>
              <gml:TimeInstant>
                <gml:timePosition>2014-05-07T19:41:36+02</gml:timePosition>
              </gml:TimeInstant>
            </gml:begin>
          </gml:TimePeriod>
        </day>
      </days>
    </location>
  </gml:featureMember>
  ...

```

Si no existen datos para algún de los días solicitados pero si para otros, o comportamiento es el siguiente:

- Para cada día sin datos comprendido entre el instante actual (o en su caso o *startDate*) y o último día con datos, se devuelve un elemento *day* con la siguiente estructura:

```

<day>
  <gml:TimePeriod>
    <gml:begin>
      <gml:TimeInstant>
        <gml:timePosition>yyyy-MM-ddTHH:mm:ss</gml:timePosition>
      </gml:TimeInstant>
    </gml:begin>
    <gml:end>
      <gml:TimeInstant>
        <gml:timePosition>yyyy-MM-ddTHH:mm:ss</gml:timePosition>
      </gml:TimeInstant>
    </gml:end>
  </gml:TimePeriod>
</variables/>

```


</day>

- Si se invocó la operación especificando el parámetro *endTime*, para los días sin datos comprendidos entre el último día con datos y el *endTime*, se devuelve un elemento *day* con la Estructura anterior.
- Si se invocó la operación sin especificar el parámetro *endTime*, los días sin datos posteriores al último día con datos no figuran en la respuesta.

Cuando una variable no tiene datos para ningún instante horario dentro del rango temporal del elemento *day*, el elemento *variable* estará vacío. La estructura del elemento *day* será la siguiente:

```
<day>
  <gml:TimePeriod>
    <gml:begin>
      <gml:TimeInstant>
        <gml:timePosition>yyyy-MM-ddTHH:mm:ss</gml:timePosition>
      </gml:TimeInstant>
    </gml:begin>
    <gml:end>
      <gml:TimeInstant>
        <gml:timePosition>yyyy-MM-ddTHH:mm:ss</gml:timePosition>
      </gml:TimeInstant>
    </gml:end>
  </gml:TimePeriod>
  <variables>
    ...
    <variablename="VARIABLE_NAME".../>
    ...
  </variables>
</day>
```

El caso de ausencia de datos para un instante temporal concreto se describe en las secciones dedicadas a cada operación

■ HTML:

Na respuesta sólo se incluyen los días para los que hay datos. Dentro de cada día, sólo se incluyen las variables para las que existen datos para cuando menos un instante temporal. En los instantes del día para los que no haya datos (o no se pidieran), se pone un guión “-”

Si no existen datos para ninguno de los días solicitados se devuelve una excepción, salvo en el caso de que se pidieran datos de más de una localización. En este caso, se devuelve un documento HTML con contenido específico para cada localización, como se ve en la siguiente imagen:

Predición para latitude=43.47569, lonxitude=-8.19504:
Neste intre non hai datos dispoñibles para o intervalo temporal indicado.

Predición para latitude=43.47569, lonxitude=-8.19505:
Neste intre non hai datos dispoñibles para o intervalo temporal indicado.

^aTodas as horas en hora local de Galicia

■ KML:

Para el caso del formato KML, lo comentado para el formato HTML se aplica al elemento *description* del documento KML. En el caso de que se pidieran datos de más de una localización y no haya datos para algunas (o ninguna), se devuelve un mensaje de excepción dentro de cada localidad sin datos. Por ejemplo:

```
<?xmlversion="1.0"encoding="UTF-8"?>
<kmlxmlns="http://earth.google.com/kml/2.0">
  <Document>
    <Placemarkid="point_1">
      <ExtendedData>
        <Dataname="exceptionCode">
          <value>216</value>
        </Data>
        <Dataname="exceptionMessage">
          <value>El punto indicado cae fuera de los límites geográficos para los cales hay
          datos.
          </value>
        </Data>
      </ExtendedData>
      <Point>
        <coordinates>-8.29504,43.47569</coordinates>
      </Point>
    </Placemark>
    <Placemarkid="point_2">
      <description>&lt;![CDATA[&lt;![DOCTYPEhtmlPUBLIC...
...

```

Operación / f i n d P l a c e s

5.1 Introducción

Esta operación sirve para localizar lugares a partir de una cadena de caracteres. La información devuelta para cada lugar se incluyen atributos alfanuméricos (nombre del lugar, nombre del municipio al que pertenece, etc.), un identificador, que permitirá hacer referencia al lugar en otras operaciones, y sus coordenadas.

Se puede indicar o tipo o tipos de lugares que se desea buscar. Se no se establece este parámetro, se busca entre todos los tipos de lugares. Os tipos de lugares contemplados actualmente son:

Nombre	Descripción	Ámbito
locality	Entidades de población	Galicia
beach	Playas	Galicia

El criterio de coincidencia es el siguiente: se devolverán todos los lugares nombre comience, contenga el final con la cadena de caracteres especificada, sea en mayúsculas o minúsculas. El número máximo de resultados que se devuelve en una respuesta es 1000.

La información sobre las entidades de población fue proporcionada por el SITGA (Sistema de Información Territorial de Galicia, <http://sitga.xunta.es>) y la localización de las playas fue facilitada por INTECMAR (Instituto Tecnológico para el Control del Medio Marino de Galicia, <http://www.intecmar.gal/>)

5.2 Parámetros

Los parámetros que admite esta operación son:

Nombre	Obligatorio	Posibles valores	Valor por defecto	Comentarios
API_KEY	SI	Clave de la API	-	Ver la sección <i>Obtención de la clave para el uso de la API</i> .
location	SI	Cualquier cadena de caracteres	-	Por ejemplo “ouren” o “coru”.
types	NO	Secuencia separada por comas de los tipos de lugares entre los que buscar (ver a <i>tabla de tipos de lugares</i>)	-	
lang	NO	‘gl’‘es’‘en’	‘en’	Idioma en el que se devolverán los textos, incluyendo los textos de las excepciones.
format	NO	‘gml3’‘kml’‘application/json’	‘application/json’	Formato en el que se devolverán los resultados.
exceptionsFormat	NO	‘application/xml’ ‘application/json’	‘application/json’	Formato en el que se devolverán las excepciones.

5.3 Resultados

Esta operación devuelve los resultados siguiendo el formato especificado por el parámetro *format*. Se trata de un conjunto de uno o varios lugares. En función del formato especificado, a Estructura de datos devuelta será:

- Un objeto *FeatureCollection*, en el caso de GeoJSON, con tantos objetos *Feature* como coincidencias se encuentren.
- Un elemento *gml:FeatureCollection* en el caso de GML, con tantos elementos *gml:featureMember* como coincidencias se encuentren.
- Un documento KML, en el caso de KML, con tantos elementos *Placemark* como coincidencias se encuentren.

En cualquier caso, las geometrías devueltas serán siempre puntos. Las coordenadas de los lugares pueden ser modificadas ligeramente de vez en cuando sin previo aviso con el fin de mejorar su colocación o ajustarlas a las geometrías de las mallas de los modelos para optimizar la predicción devuelta, con lo cual no se garantiza que sean siempre las mismas.

Cada lugar tiene los siguientes atributos, presentados de distinta manera según el formato de respuesta elegido:

Nombre	Comentarios
id	Identificador(para usar en futuras buscas por <i>id</i>)
name	Nombre del lugar
municipality	Nombre del municipio al que pertenece
province	Nombre da provincia a la que pertenece
type	Tipo de lugar
geometry	Las coordenadas del lugar

Para las geometrías (coordenadas) de cada localización se usan las siguientes representaciones:

Formato	Representación
GML	Se representa con un elemento gml:point contenido en un elemento geometry.
HTML	Se representa como texto dentro del HTML.
JSON	Se representa con un objeto <i>geometry</i> de tipo punto (“type”:“Point”).
KML	Se representa con un elemento <i>Point</i> .

Más en detalle para cada formato de respuesta:

- **JSON:**

Devuelve un objeto GeoJSON con la siguiente estructura:

```
{
  "type": "FeatureCollection",
  "crs": {
    "type": "name",
    "properties": {
      "name": "CRS"
    }
  },
  "features": FEATURES_ARRAY
}
```

Donde:

- CRS indica el sistema de coordenadas utilizado (actualmente siempre es urn:ogc:def:crs:OGC:1.3:CRS84).
- FEATURES_ARRAY es un *array* JSON ([elemento1, elemento2...]), donde cada elemento es un objeto *feature* con la siguiente forma

```
{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [
      X,
      Y
    ]
  },
  "properties": {
    "id": ID,
    "name": NAME,
    "municipality": MUNICIPALITY,
    "province": PROVINCE,
    "type": TYPE
  }
}
```

Donde:

- X y Y son las coordenadas de la localización (longitud y latitud respectivamente).
- ID, NAME, MUNICIPALITY, PROVINCE y TYPE son los atributos, ya mencionados, de la localización.

- **GML:**

Devuelve un documento GML en el que la raíz es un elemento gml:featureCollection con la siguiente estructura

```
<gml:FeatureCollection xmlns="https://www.meteogalicia.gal/meteosix" xmlns:gml="http://www.opengis.net/gml">
  <gml:boundedBy>
    <gml:EnvelopesrsName=CSR>
```

```

        <gml:pos>MIN_YMIN_X</gml:pos>
        <gml:pos>MAX_YMAX_X</gml:pos>
    </gml:Envelope>
</
    gml:boundedBy>FE
    ATURE_MEMBER_1FEA
    TURE_MEMBER_2
    ...
</gml:FeatureCollection>

```

Donde:

- CRS indica el sistema de coordenadas utilizado (actualmente siempre es EPSG:4326)
- MIN_Y, MIN_X, MAX_Y, MAX_X son las coordenadas mínimas y máximas del *Envelope* del conjunto de todas las localizaciones. El orden es latitud, longitud
- FEATURE_MEMBER_1, FEATURE_MEMBER_2... son elementos de tipo gml:featureMember, que a su vez presentan la siguiente estructura:

```

<gml:featureMember>
  <location>
    <geometry>
      <gml:PointsrsName=CSR>
        <gml:pos>YX</gml:pos>
      </gml:Point>
    </geometry>
    <id>ID</id>
    <name>NAME</name>
    <municipality>MUNICIPALITY</municipality>
    <province>PROVINCE</province>
    <type>TYPE</type>
  </location>
</gml:featureMember>

```

Donde:

- Y y X son las coordenadas de la localización (latitud y longitud respectivamente)
- ID, NAME, MUNICIPALITY, PROVINCE y TYPE son los atributos, ya mencionados, de la localización.

■ KML:

Devuelve un documento KML con la siguiente estructura:

```

<kml xmlns="http://earth.google.com/kml/2.0">
  <Document>
    <Placemark id="point_1">
      <description>NAME-MUNICIPALITY (PROVINCE)</description>
      <Point>
        <coordinates>X,Y</coordinates>
      </Point>
      <ExtendedData>
        <Data name="location_id">
          <value>ID</value>
        </Data>
        <Data name="location_name">
          <value>NAME</value>
        </Data>
        <Data name="location_municipality">
          <value>MUNICIPALITY</value>
        </Data>
      </ExtendedData>
    </Placemark>
  </Document>
</kml>

```

```

        <Dataname="location_province">
            <value>PROVINCE</value>
        </Data>
        <Dataname="location_type">
            <value>TYPE</value>
        </Data>
    </ExtendedData>
</Placemark>
<Placemarkid="point_2">
    ...
</Placemark>
    ...
</Document>
</kml>

```

Donde:

- X y Y son las coordenadas de la localización (longitud y latitud respectivamente).
- ID, NAME, MUNICIPALITY, PROVINCE y TYPE son los atributos, ya mencionados, de la localización. Cada *Placemark* tiene un atributo id con los valores son point_1, point_2... y así sucesivamente para cada *Placemark* contenido en la respuesta

Se no se encuentra ningún resultado se devuelve el siguiente:

■ **JSON:**

```

{
  "crs":
    { "properties":
      {
        "name": "urn:ogc:def:crs:OGC:1.3:CRS84"
      },
      "type": "name"
    },
  "features": null,
  "type": "FeatureCollection"
}

```

■ **GML:**

```

<?xmlversion="1.0"encoding="UTF-8"?>
<gml:FeatureCollectionxmlns="https://www.meteogalicia.gal/
meteosix"xmlns:gml="http://www.opengis.net/gml"/>

```

■ **KML:**

```

<?xmlversion="1.0"encoding="UTF-8"?>
<kmlxmlns="http://earth.google.com/kml/2.0">
  <Document/>
</kml>

```

5.4 Ejemplos

Buscar lugares que contengan el texto ‘oure’:

https://servizos.meteogalicia.gal/apiv4/findPlaces?location=oure&API_KEY=***

Buscar lugares que contengan el texto ‘lanza’:

https://servizos.meteogalicia.gal/apiv4/findPlaces?location=lanza&API_KEY=***

Buscar playas que contengan el texto 'lanza':

https://servizos.meteogalicia.gal/apiv4/findPlaces?location=lanza&types=beach&API_KEY=***

Operación / getNumericForecastInfo

6.1 Introducción

Esta operación permite obtener información sobre a predicción numérica para distintas variables meteorológicas y oceanográficas. Algunas variables están disponibles en varios modelos de predicción y mallas (ver la sección *Modelos de predicción numérica*). La siguiente tabla muestra las variables disponibles. Para cada una se indica los modelos en los que se encuentra, las unidades en las que se pueden devolver los datos y si los datos tienen o no un símbolo asociado.

Nombre	Descripción	Modelos	Valores	unidades	unidades por defecto	Símbolo asociado
sky_state	Estado del ceo	<i>WRF</i>	SUNNY, HIGH_CLOUDS, PARTLY_-CLOUDY, OVERCAST, CLOUDY,FOG, SHOWERS, OVERCAST_AND_SHOWERS, INTERMITTENT_SNOW, DRIZZLE, RAIN, SNOW, STORMS, MIST, FOG_BANK, MID_CLOUDS, WEAK_RAIN, WEAK_SHOWERS, STORM_THEN_CLOUDY, MELTED_SNOW, RAIN_HayL	-1	-	SI
temperature	Temperatura	<i>WRF</i>	Número entero	degC degK degF	degC	NO
precipitation_amount	Precipitación acumulada durante la hora anterior	<i>WRF</i>	Número real (con 2 decimais)	lm2	lm2	NO

wind	viento	<i>WRF</i>	Devuelve los valores de módulo y dirección del viento, ambos números reais con 2 decimais	kmh_deg ms_deg mph_deg kt_deg	kmh_deg	SI
------	--------	------------	---	--	---------	----

1 Si se especifica el parámetro units, la unidad para esta variable tiene que quedar vacía (sin espacios en blanco)

relative_humidity	Humedad relativa	<i>WRF</i>	Número real (con 2 decimales)	perc	perc	NO
cloud_area_fraction	Cobertura de nubes	<i>WRF</i>	Número real (con 2 decimales)	perc	perc	NO
air_pressure_at_sea_level	Presión al nivel del mar	<i>WRF</i>	Número entero	hpapaatm	hpa	NO
snow_level	Cota de nieve	<i>WRF</i>	Número entero	mft	m	NO
sea_water_temperature	Temperatura del agua	<i>ROMS, MOHID</i>	Número entero	degC degK degF	degC	NO
significant_wave_height	Altura de ola	<i>WW3, SWAN</i>	Número real (con 2 decimales)	mft	m	NO
mean_wave_direction	Dirección del mar	<i>WW3, SWAN</i>	Número real (con 2 decimales)	deg	deg	SI
relative_peak_period	Período de ola	<i>WW3, SWAN</i>	Número entero	s	s	NO
sea_water_salinity	Salinidad del agua	<i>ROMS, MOHID</i>	Número real (con 2 decimales)	psu	psu	NO

Donde:

e:

- *degC*: grados Celsius ($^{\circ}C$)
- *degK*: grados Kelvin ($^{\circ}K$)
- *degF*: grados Fahrenheit ($^{\circ}F$)
- *kmh-deg*: kilómetros por hora (*km/h*) – grados ($^{\circ}$)
- *mdeg*: metros por segundo (*m/s*) – grados ($^{\circ}$)
- *mph-deg*: millas por hora (*mph*) – grados ($^{\circ}$)
- *kt-deg*: nús (*kt*) – grados ($^{\circ}$)
- *m*: metros (*m*)
- *ft*: pés (*ft*)
- *lm2*: litros por metro cuadrado (*l/m2*)
- *perc*: porcentaje
- *(%)hpa*: hectopascales (*hP*)
- *a)pa*: pascales (*Pa*)
- *atm*: atmosferas (*atm*)
- *s*: segundos (*s*)
- *deg*: grados ($^{\circ}$)
- *psu*: unidades prácticas de salinidad (*psu*)

6.2 Rango temporal

Esta operación tiene en cuenta el valor de las horas, minutos y segundos de los parámetros `startTime` y `endTime`. El número máximo de días que se puede solicitar en una petición es 7. Si no se indica el parámetro `endTime`, por defecto se devuelven todos los datos disponibles (desde el instante actual o desde el instante indicado en el parámetro `startTime`) hasta o máximo de 7 días. En la práctica, los datos disponibles dependen de cada modelo y malla (ver la sección Modelos de predicción numérica)

6.3 Parámetros

Además de los parámetros indicados en la *tabla de parámetros comunes*, esta operación admite los siguientes parámetros:

Nombre	Obligatorio	Posibles valores	Valorpordefecto	Comentarios
autoAdjustPostion	NO	<i>True o false</i>	<i>true</i>	Si el valor es true, en los puntos próximos a la costa y con el fin de dar una predicción más fiable, para algunas variables (temperatura del aire, viento y variables oceanográficas) se realiza de forma automática un ajuste de posición de donde se toman los datos. Así, o punto de donde se toman los datos puede ser ligeramente distinto del punto indicado en la petición. 2
variables	NO	Lista de nombres de variables, separados por comas, de entre los nombres indicados en la <i>tabla de variables de predicción</i>	<i>sky_state, temperature, wind, precipitation_amount</i>	
models	NO	Lista de nombres de modelos separados por comas. Los posibles nombres son: <i>WRF, WW3, SWAN, ROMS y MOHID</i> .	-	Si se establece, tiene que tener tantos elementos como variables se indiquen no parámetro variables. Cada modelo se refiere a la variable que ocupa a misma posición en la lista de variables. Un elemento puede ser la cadena vacía y en ese caso se usará o mejor modelo disponible.
grids	NO	Lista de nombres de mallas, separados por comas, de entre los nombres indicados en la columna <i>Malla</i> de las tablas: <i>WRF, WW3, SWAN, ROMS y MOHID</i> .	-	Si se establece, tiene que tener tantos elementos como variables se indiquen no parámetro variables. Cada malla se refiere a la variable y al modelo que ocupan la misma posición en sus respectivos parámetros. Un elemento puede ser la cadena vacía y en ese caso se usará a mejor malla disponible.
units	NO	Lista de nombres, separados por comas, de entre los nombres que figuran en la columna unidades de las tablas unidades en la <i>tabla de variables de predicción</i>	Ver a <i>tabla de variables de predicción</i>	Si se establece, tiene que tener tantos elementos como variables se indiquen no parámetro variables. Cada unidad se refiere a la variable que ocupa a misma posición nel parámetro variables. Un elemento puede ser a cadena vacía y en ese caso se usará a unidad por defecto.

En función del que se indique en la petición, el comportamiento es el siguiente:

- Si en la petición no se especifican las variables, las variables que se toman por defecto son: sky_state, temperature, wind, precipitation_amount.
- Si en la petición no se especifican los modelos ni las mallas, para cada variable se devolverán los valores correspondientes a las ejecuciones de los mejores modelos y mallas que estén disponibles. Es posible que a mejor combinación modelo-malla no tenga datos a partir de un determinado instante, pero si los tengan otros modelos-mallas. la API comprueba, para cada día y para cada variable, cantos instantes con datos tiene cada par modelo-malla disponible, y se usa el par modelo-malla que disponga de más datos. Cuando hay varios pares modelo-malla que tienen el mismo número de datos se usa a mejor malla. Por ejemplo, la petición:

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?coords=-8.4275,43.4336&variables=significant_wave_height,sea_water_temperature&API_KEY=***
```

devolverá:

- Los valores de la variable significant_wave_height de la malla Artabro del modelo SWAN para los primeros días y de la malla AtlanticoNorte del WW3 para los últimos días.
 - Los valores de la variable sea_water_temperature de la malla Artabro del modelo MOHID para los primeros días y de la malla Galicia del ROMS para los últimos días.
- Si en la petición si se especifican las variables y los modelos, pero no las mallas, se aplica lo comentado en el punto anterior, pero considerando sólo los modelos indicados en la petición (no otros modelos). Por ejemplo, la petición:

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?coords=-.838158,43.489911&variables=temperature,significant_wave_height,sea_water_temperature&models=WRF,WW3,ROMS&API_KEY=***
```

devolverá:

- Los valores de la variable temperature de la malla Artabro1Km del modelo WRF para los primeros días y de la malla 04km del WRF para los últimos días.
 - Los valores de la variable significant_wave_height de la malla AtlanticoNorte del modelo WW3.
 - Los valores de la variable sea_water_temperature de la malla Galicia del modelo ROMS.
- Si en la petición se especifican las variables, los modelos y las mallas, para cada variable se devolverán los valores correspondientes a los modelos especificados sobre las mallas indicadas.
 - Es posible pedir información de una misma variable en varias mallas y modelos. Por ejemplo, la petición:

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?coords=-8.4275,43.4336&variables=sky_state,sky_state,temperature,significant_wave_height,significant_wave_height,sea_water_temperature&models=WRF,WRF,WW3,WW3,ROMS&gridids=04km,12km,36km,Galicia,AtlanticoNorte,Galicia&API_KEY=***
```

devolverá:

- Los valores de la variable sky_state en la malla 04km del modelo WRF.
- Los valores de la variable sky_state en la malla 12km del modelo WRF.

²Para realizar este ajuste se usa información derivada de datos procedentes de las siguientes fuentes:

- Línea de costa da Península Ibérica: Instituto Hidrográfico de la Armada (España): http://www.armada.mde.es/ihtm/Aplicaciones/pruebas/costa/LINEA_-DE_COSTA.zip
- Límites de las comunidades autónomas de España: Centro Nacional de Información Geográfica (España) http://centrode.lascargas.cbig.es/CentroDescargas/equipamiento/lineas_limite.zip
- Límites NUT1 de Portugal: Direção-Geral del Território (Portugal): <http://mapas.igeo.pt/ows/caop/continente?SERVICE=WFS&VERSION=1.0.0&REQUEST=Get2013>

- Los valores de la variable `temperature` en la malla 36km del modelo WRF.
 - Los valores de la variable `significant_wave_height` en la malla Galicia del modelo WW3.
 - Los valores de la variable `significant_wave_height` en la malla AtlanticoNorte del modelo WW3.
 - Los valores de la variable `sea_water_temperature` en la malla Galicia del modelo ROMS.
- Es posible especificar el modelo y malla a usar para algunas variables y no facelel para otras (que usarían los mejores disponibles, tal como se explicó más arriba). Para eso se conservará la posición donde iría el nombre del modelo o de la malla sin poner ningún texto (cadena vacía). Por ejemplo, la petición:

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?coords=-8.4275,43.4336&variables=significant_wave_height,sea_water_temperature&models=WW3,&grids=AtlanticoNorte,&API_KEY=***
```

devolverá:

- Los valores de la variable `significant_wave_height` de la malla AtlanticoNorte del modelo WW3.
- Los valores de la variable `sea_water_temperature` de la malla Artabro del modelo MOHID para los primeros días y de la malla Galicia del ROMS para los últimos días.

Observaciones:

- Si en la petición se especifica algún modelo, es obligatorio especificar las variables.
- Si se especifica alguna malla, es obligatorio especificar el modelo con el que se corresponde la malla.
- Se o número de variables especificadas no coincide con el número de modelos o con el número de mallas (contando as cadenas vacías) se devuelve una excepción.

6.4 Resultados

La estructura de los datos devueltos por esta operación es la indicada en el apartado *Estructura de los datos devueltos por las operaciones /getNumericForecastInfo, /getTidesInfo y /getSolarInfo*. Si una variable se pide para más de una combinación modelo-malla, cada día contendrá un objeto *variable* para cada combinación. Cada *variable* contiene la siguiente información:

Nombre	Comentarios
<code>name</code>	El nombre de la variable.
<code>model</code>	El nombre del modelo del que proceden los datos.
<code>grid</code>	El nombre de la malla de la que proceden los datos.
<code>units</code>	El nombre de la unidad de medida en la que se devuelven los datos. No aparecen las variables <code>sky_state</code> nin <code>wind</code> .
<code>moduleUnits</code>	El nombre de la unidad de medida en la que se devuelve o módulo del viento. Sólo aparece en la variable <code>wind</code> .
<code>directionUnits</code>	El nombre de la unidad de medida en la que se devuelve a dirección del viento. Sólo aparece en la variable <code>wind</code> .
<code>geometry</code>	Las coordenadas del punto finalmente usados para tomar los datos. Este no tiene porqué coincidir exactamente con el punto que se indicó en la petición.
<code>values</code>	Es el conjunto de valores de la variable (combinación variable-modelo-malla) contenidos dentro del rango temporal del objeto <code>day</code> . Cada valor contiene los datos indicados en la siguiente tabla.

Contenido de los *valores de las variables*:

Nombre	Comentarios
timeInstant	La hora a la que corresponde el valor, en formato <i>yyyy-MM-ddTHH:mm:ss+XX</i> .
modelRun	La hora de ejecución del modelo del cual proceden los valores, en formato <i>y y y y - M M - ddTHH:mm:ss+XX</i> .
value	El valor de la variable. puede ser un valor numérico o una cadena de caracteres, dependiendo de la variable de la que se trate. No aparece en la variable <i>wind</i> .
moduleValue	El valor del módulo del viento. Sólo aparece en la variable <i>wind</i> .
directionValue	El valor de la dirección del viento. Sólo aparece en la variable <i>wind</i> .
iconURL	En las variables con símbolo asociado (ver la tabla de variables) se incluye este atributo con una URL de donde obtener a icono que representa el valor. Los iconos son parte del MeteoSIX.

Más en detalle para cada formato de respuesta:

▪ **JSON:**

El array VARIABLES_ARRAY contendrá tantos objetos variable como combinaciones variable-modelo-malla solicitáronse (salvo que no haya datos para ninguna de las variables solicitadas, en tal caso se sigue lo comentado en [Comportamiento en caso de ausencia de datos](#)). La estructura de cada variable es la siguiente:

- Variable *sky_state*:

```
{
  "name":VARIABLE_NAME,
  "model":MODEL_NAME, "gr
  id":GRID_NAME,
  "geometry":
    { "type": "Point", "
      coordinates": [
        X,
        Y
      ]
    }
  "values":VALUES_ARRAY
}
```

- Variable *wind*:

```
{
  "name":VARIABLE_NAME,
  "model":MODEL_NAME,
  "grid":GRID_NAME,
  "moduleUnits":MODULE_UNIT_NAME,
  "directionUnits":DIRECTION_UNIT_NAME,
  "geometry":{
    "type": "Point",
    "coordinates": [
      X,
      Y
    ]
  }
  "values":VALUES_ARRAY
}
```

- Resto de las variables:

```
{
  "name":VARIABLE_NAME,
  "model":MODEL_NAME,
  "grid":GRID_NAME,
  "units":UNIT_NAME,
  "geometry":{
    "type": "Point",
```

```

    "coordinates": [
      X,
      Y
    ]
    "values": VALUES_ARRAY
  }

```

Donde
:

- *VARIABLE_NAME* es el nombre de la variable.
- *MODEL_NAME* es el modelo del que proceden los datos.
- *GRID_NAME* es la malla utilizada en la ejecución de la que proceden los datos.
- *MODULE_UNIT_NAME* es la unidad de medida en la que se devuelven los datos de módulo del viento.
- *DIRECTION_UNIT_NAME* es la unidad de medida en la que se devuelven los datos de la dirección del viento.
- *UNIT_NAME* es la unidad de medida en la que se devuelven los datos.
- *X* y *Y* son las coordenadas del punto del que proceden los datos (longitud y latitud respectivamente). Este punto no tiene porqué coincidir exactamente con el punto indicado en la petición.
- *VALUES_ARRAY* es un array JSON con los datos para esa combinación variable-modelo-malla comprendidos dentro del rango temporal del objeto *today*.

Los elementos del *VALUES_ARRAY* tienen la siguiente estructura:

- En variables sin símbolo asociado:

```

{
  "timeInstant": TIME_INSTANT,
  "modelRun": MODEL_RUN,
  "value": VALUE
}

```

- En variables con símbolo asociado, salvo la variable *wind*:

```

{
  "timeInstant": TIME_INSTANT,
  "modelRun": MODEL_RUN,
  "value": VALUE,
  "iconURL": ICON_URL
}

```

- En la variable *wind*:

```

{
  "timeInstant": TIME_INSTANT,
  "modelRun": MODEL_RUN,
  "moduleValue": MOD_VALUE,
  "directionValue": DIR_VALUE,
  "iconURL": ICON_URL
}

```

Donde
:

- *TIME_INSTANT* es la hora a la que corresponde el dato, con formato *yyyy-MMddTHH:mm:ss+XX*.
- *MODEL_RUN* es la hora del inicio de ejecución del modelo de la que proceden los datos, con formato *yyyy-MMddTHH:mm:ss+XX*.

- *VALUE* es el valor. El seu tipo de dato dependerá de la variable de la que se trate (ver a *tabla de variables de predicción*).
- *ICON_URL* es unhaURL, de acceso público, que apunta a la icono que representa el valor (proporcionada por lo MeteoSIX).
- *MOD_VALUE* es el valor del módulo del viento.
- *DIR_VALUE* es el valor de la dirección del viento.

En el caso de no disponer de datos para un instante concreto de entre los que podrían tener datos (actualmente, las horas en punto), el formato para ese instante será:

- en las variables sin símbolo asociado:

```
{
  "timeInstant": TIME_INSTANT,
  "modelRun": null,
  "value": null
}
```

- en las variables con símbolo asociado, salvo la variable *wind*:

```
{
  "timeInstant": TIME_INSTANT,
  "modelRun": null,
  "value": null,
  "iconURL": null
}
```

- En la variable *wind*:

```
{
  "timeInstant": TIME_INSTANT,
  "modelRun": null,
  "moduleValue": null,
  "directionValue": null,
  "iconURL": null
}
```

Donde

:

- *TIME_INSTANT* es el instante temporal.

Si un objeto *variable* no tiene ningún dato dentro del rango temporal del objeto *day* en el que se encuentra, el contenido del objeto *variable* será el siguiente:

- *name* contendrá el nombre de la variable.
- *model* contendrá el nombre del modelo si en la petición se especificó un modelo, si no será nulo (*null*).
- *grid* contendrá el nombre de la malla si en la petición se especificó una malla, si no será nulo (*null*).

units será nulo (*null*) salvo en el caso de *wind*, no que no estará presente.

- *moduleUnits* será nulo (*null*) en el caso de *wind* y en el resto de variables no estará presente.
-

directionUnits será nulo (*null*) en el caso de *wind* y en el resto de variables no estará presente.

geometry será siempre nulo (*null*).

- *values* será siempre nulo (*null*).

- **GML:**

Cada elemento *variable* es de la siguiente forma:

- Variable *sky_state*:

```
<variablename=VARIABLE_NAMEmodel=MODEL_NAMEgrid=GRID_NAME>
  <geometry>
    <gml:Point srsName=CRS>
      <gml:pos>Y X</gml:pos>
    </gml:Point>
  </geometry>
  <values>
    VALUE_1
    VALUE_2
    ...
  </values>
</variable>
```

- Variable *wind*:

```
<variable name=VARIABLE_NAME model=MODEL_NAME grid=GRID_NAME
moduleUnits=MODULE_UNIT_NAME directionUnits=DIRECTION_UNIT_NAME>
  <geometry>
    <gml:Point srsName=CRS>>
      <gml:pos>Y X</gml:pos>
    </gml:Point>
  </geometry>
  <values>
    VALUE_1
    VALUE_2
    ...
  </values>
</variable>
```

- Resto de las variables:

```
<variablename=VARIABLE_NAME model=MODEL_NAME grid=GRID_NAME units=UNIT_NAME>
  <geometry>
    <gml:Point srsName=CRS>
      <gml:pos>Y X</gml:pos>
    </gml:Point>
  </geometry>
  <values>
    VALUE_1
    VALUE_2
    ...
  </values>
</variable>
```

Donde:

- *VARIABLE_NAME* es el nombre de la variable.
- *MODEL_NAME* es el modelo del que proceden los valores.
- *GRID_NAME* es la malla utilizada en la ejecución de la que proceden los datos.
- *CRS* indica el sistema de coordenadas usado (actualmente es siempre *EPSG:4326*).
- *MODULE_UNIT_NAME* es la unidad de medida en la que se devuelven los datos de módulo del viento.
- *DIRECTION_UNIT_NAME* es la unidad de medida en la que se devuelven los datos de la dirección del viento.
- *UNIT_NAME* es la unidad de medida en la que se devuelven los datos.

- *Y y X* son las coordenadas del punto del que proceden los valores (latitud y longitud respectivamente). Este punto no tiene porqué coincidir exactamente con el punto indicado en la petición.
- *VALUE_1, VALUE_2...* son los valores para esa combinación variable-modelo-malla comprendidos dentro del rango temporal del objeto *day*.

a su vez, cada un de estos elementos *VALUE_1, VALUE_2...* presenta la siguiente estructura:

- en las variables sin símbolo asociado:

```
<hourValue>
  <gml:TimeInstant>
    <gml:timePosition>TIME_INSTANT</gml:timePosition>
  </gml:TimeInstant>
  <modelRun>
    <gml:TimeInstant>
      <gml:timePosition>MODEL_RUN</gml:timePosition>
    </gml:TimeInstant>
  </modelRun>
  <value>VALUE</value>
</hourValue>
```

- en las variables con símbolo asociado, salvo la variable *wind*:

```
<hourValue>
  <gml:TimeInstant>
    <gml:timePosition>TIME_INSTANT</gml:timePosition>
  </gml:TimeInstant>
  <modelRun>
    <gml:TimeInstant>
      <gml:timePosition>MODEL_RUN</gml:timePosition>
    </gml:TimeInstant>
  </modelRun>
  <value>VALUE</value>
  <iconURL>ICON_URL</iconURL>
</hourValue>
```

- En la variable *wind*:

```
<hourValue>
  <gml:TimeInstant>
    <gml:timePosition>TIME_INSTANT</gml:timePosition>
  </gml:TimeInstant>
  <modelRun>
    <gml:TimeInstant>
      <gml:timePosition>MODEL_RUN</gml:timePosition>
    </gml:TimeInstant>
  </modelRun>
  <moduleValue>MOD_VALUE</moduleValue>
  <directionValue>DIR_VALUE</directionValue>
  <iconURL>ICON_URL</iconURL>
</hourValue>
```

Donde:

- *TIME_INSTANT* es la hora a la que corresponde el dato, con formato *yyyy-MMddTHH:mm:ss+XX*.
- *MODEL_RUN* es la hora del inicio de ejecución de la que proceden los datos, con formato *yyyy-MmddTHH:mm:ss+XX*.
- *VALUE* es el valor. Su tipo de dato dependerá de la variable de la que se trate (ver la [tabla de variables de predicción](#)).

- *ICON_URL* es una URL, de acceso público, que apunta al icono que representa el valor (proporcionada poloMeteoSIX).
- *MOD_VALUE* es el valor del módulo del viento.
- *DIR_VALUE* es el valor de la dirección del viento.

En el caso de no disponer de datos para un instante concreto de entre los que podrían tener datos (actualmente, las horas en punto), el formato para ese instante será:

- en las variables sin símbolo asociado:

```
<hourValue>
  <gml:TimeInstant>
    <gml:timePosition>TIME_INSTANT</gml:timePosition>
  </gml:TimeInstant>
  <modelRun/>
  <value/>
</hourValue>
```

- en las variables con símbolo asociado, salvo la variable *wind*:

```
<hourValue>
  <gml:TimeInstant>
    <gml:timePosition>TIME_INSTANT</gml:timePosition>
  </gml:TimeInstant>
  <modelRun/>
  <value/>
  <iconURL/>
</hourValue>
```

- En la variable *wind*:

```
<hourValue>
  <gml:TimeInstant>
    <gml:timePosition>TIME_INSTANT</gml:timePosition>
  </gml:TimeInstant>
  <modelRun/>
  <moduleValue/>
  <directionValue/>
  <iconURL/>
</hourValue>
```

Donde:

- TIME_INSTANT es el instante temporal para el que no hay datos.

Si una variable no tiene ningún dato dentro del rango temporal del elemento *day* en el que se encuentra, la variable no contendrá los elementos *geometry* ni *values*, y en cuanto a los atributos:

- *name* contendrá el nombre de la variable.
- *model* contendrá el nombre del modelo si en la petición se especificó un modelo, si no no aparecerá.
- *grid* contendrá el nombre de la malla si en la petición se especificó una malla, si no no aparecerá.

units no aparecerá.

- *moduleUnits* no aparecerá.
- *directionUnits* no aparecerá.

■ HTML:

A mayores de la información devuelta en otros formatos, en HTML se incluyen resúmenes diarios para cada variable. Si una variable no tiene valor para un instante temporal (porque no hay datos disponibles o porque es un instante pasado) se indica con un guión-. Así, por ejemplo, la respuesta ante una petición de las variables *wind* y *sky_state* tendría el siguiente aspecto:

Predicción para latitude=44, lonxitude=-8

Luns abril 29, 2013

	Resumo	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
Estado do ceo		-	-									
Vento	Min Max 15 53	-	-	19	19	18	17	15	15	17	17	16

Martes abril 30, 2013

	Resumo	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
Estado do ceo												
Vento	Min Max 46 51	50	49	51	49	50	50	50	49	48	48	49

Mércores maio 01, 2013

	Resumo	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
Estado do ceo												
Vento	Min Max 35 51	51	50	49	48	46	45	44	44	44	44	43

- KML

La respuesta es un documento KML con la Estructura indicada en el apartado *Estructura de los datos devueltos por las op-eraci3ns /getNumericForecastInfo, /getTidesInfo y /getSolarInfo.*

6.5 Ejemplos

Obtener los datos por defecto para un punto determinado:

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?
coords=-8.350573861318628,43.3697102138535&API_KEY=***
```

Obtener los datos por defecto para un punto determinado en formato HTML:

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?
coords=-8.350573861318628,43.3697102138535&format=text/html&API_KEY=***
```

Obtener, para Loureda (Arteixo), lugar con id 42917, a predicci3n para la temperatura en las mallas de 4 y 12km, y para o y estado del cielo en la malla de 36km, todas del modelo WRF y en formato GML:

```
https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?
locationIds=42917&variables=temperature,temperature,sky_state&models=WRF,WRF,WRF&
grids=04km,12km,36km&format=gml3&API_KEY=***
```

Obtener los datos anteriores en formato KML y con la temperatura en grados Kelvin:

[https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?
locationIds=42917&variables=temperature,temperature,sky_state&models=WRF,WRF,WRF
&grids=04km,12km,36km&format=kml&units=degk,degk,&API_KEY=***](https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?locationIds=42917&variables=temperature,temperature,sky_state&models=WRF,WRF,WRF&grids=04km,12km,36km&format=kml&units=degk,degk,&API_KEY=***)

Obtener los datos anteriores en formato HTML y para un rango de tiempo determinado:

[https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?
locationIds=42917&variables=temperature,temperature,sky_state&models=WRF,WRF,WRF&
grids=04km,12km,36km&format=text/html&units=degk,degk,&startTime=2013-03-07T15:00:00
&endTime=2013-03-08T08:00:00&API_KEY=***](https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?locationIds=42917&variables=temperature,temperature,sky_state&models=WRF,WRF,WRF&grids=04km,12km,36km&format=text/html&units=degk,degk,&startTime=2013-03-07T15:00:00&endTime=2013-03-08T08:00:00&API_KEY=***)

Nota: para ejecutar esta última petición es necesario ajustar los parámetros *startTime* y *endTime* a valores actuales.

Operación /getTidesInfo

7.1 Introducción

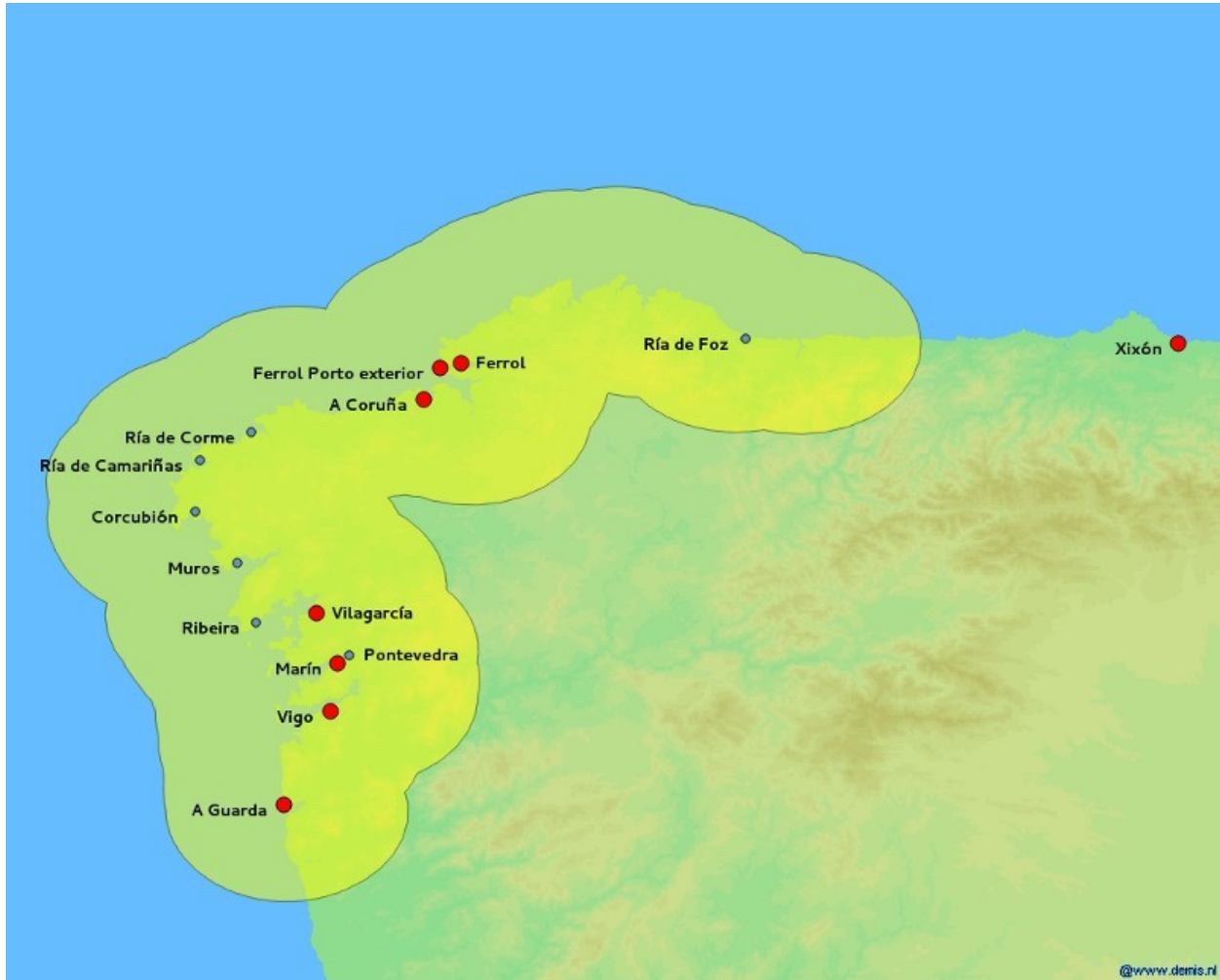
Esta operación ofrece información sobre las mareas en puntos de la costa gallega o próximos a ella. Para dar la información, se toman como base los siguientes 15 puertos:

Identificador	Nombre
1	A Coruña
3	Vigo
4	Vilagarcía
6	Ría de Foz
7	Corcubión
8	Ría de Camariñas
9	Ría de Corme
10	A Guarda
11	Ribeira
12	Muros
13	Pontevedra
14	FErrorl puerto exterior
15	Marín
16	FErrorl

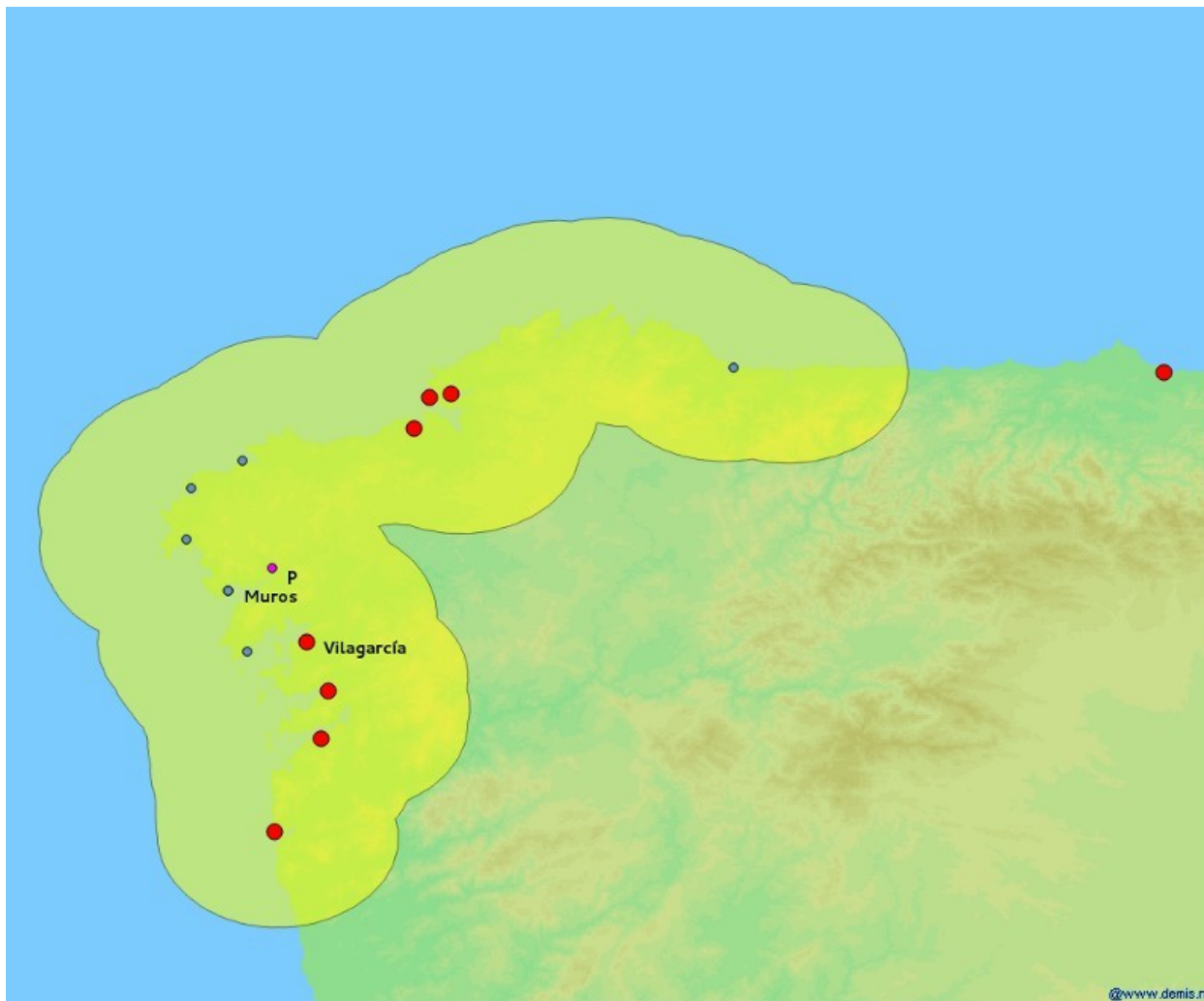
La información de la que se dispone para estos puertos es el resumen diario de mareas: hora y altura de marea para cada *pleamar* y *bajamar* del día. Cada uno de los 15 puertos tiene asociado, por proximidad, un *puerto de referencia* de entre los siguientes:

Identificador	Nombre
1	A Coruña
2	Xixón
3	Vigo
4	Vilagarcía
10	A Guarda
14	FErrorl puerto exterior
15	Marín
16	FErrorl

Para estos puertos de referencia tiene la estimación de la altura de marea para cada 30 minutos. En la figura siguiente se puede ver la localización de los puertos. Los puertos de referencia se señalan en rojo.



Cuando se invoca esta operación para un punto p (ver figura siguiente) próximo a la costa gallega (en la figura, dentro da área amarela), se recibirá por una parte, el resumen diario del puerto más próximo a p de entre los 15 considerados, el puerto **1**, que en este caso corresponde a Muros, y Además los valores de altura da marea cada 30 minutos correspondientes al puerto de referencia más próximo al puerto **1**, el puerto **2**, que en este caso corresponde a Vilagarcía.



7.2 Rango temporal

El número máximo de días que se pueden solicitar datos en una petición es 30. El instante más alejado para el que se devuelven datos es 60 días después del día actual y no se pueden pedir datos anteriores al día actual. Se no se especifica el rango temporal en la petición, se devuelven datos para 5 días desde el día actual. Se sólo se especifica el instante inicial, se devuelven datos de 5 días a partir del instante inicial (este incluido). Se sólo se especifica el instante final, se devuelven datos desde el día actual hasta el instante final (ambos incluidos), siempre que a duración no sea superior al máximo permitido. Véanse as Observaciones posteriores sobre los parámetros *startTime* y *endTime*.

7.3 Parámetros

Los parámetros que admite esta operación son los indicados en la *tabla de parámetros comunes*. En el caso de los parámetros *startTime* y *endTime*, tendrán que ser indicados con su formato completo (yyyy-MM-ddTHH:mm:ss) pero únicamente se tendrá en cuenta o año, mes y día indicados (se ignora a parte THH:mm:ss).

7.4 Resultados

La estructura de los datos devueltos por esta operación es a indicada en el apartado *Estructura de los datos devueltos por las operaciones /getNumericForecastInfo, /getTidesInfo y /getSolarInfo*. A mayores, para cada localización, da la siguientes información sobre el puerto, *port*, y el puerto de referencia, *referencePort*, de los que se están tomando los datos. Para cada puerto da la siguiente información:

Nombre	Comentarios
id	Identificador del puerto.
name	Nombre del lugar donde se encuentra el puerto.
geometry	Coordenadas del puerto.

Dentro de cada día, el contenido de cada variable es el siguiente:

Nombre	Comentarios
name	Nombre de la variable, <i>tides nestecaso</i> ¹ .
units	unidades en las que se muestran los datos.
summary	Información sobre pleamares y bajamares (ver <i>summary</i>).
values	El valor de las mareas cada media hora (ver <i>values</i>).

summary es o conjunto de pleamares y bajamares del día. Cada elemento tiene el siguiente contenido:

Nombre	Comentarios
id	Identificador del dato.
state	Estado da marea. puede tomar los valores <i>High tides</i> , para la pleamar, y <i>Low tides</i> , para la bajamar.
TimeInstant	Instante temporal no que tiene lugar una pleamar o una bajamar, segundo corresponda.
height	Altura de la marea, en metros.

values es un conjunto de datos de altura de marea cada media hora. Cada dato tiene el siguiente contenido:

Nombre	Comentarios
TimeInstant	Instante temporal sobre o que se da el dato.
height	Altura da marea, en metros.

Más en detalle para cada formato de salida:

- **JSON:**

Devuelve un objeto GeoJSON no que el objeto *properties* de cada *feature* tiene la siguiente estructura:

```
"properties":{
  "id":ID,
  "name":NAME,
  "municipality":MUNICIPALITY,
  "province":PROVINCE,
  "type":TYPE,
  "port":{
    "id":PORT_IDENTIFIER,
    "name":PORT_NAME,
    "geometry": {
      "type":"Point",
      "coordinates": [
        PORT_X,
        PORT_Y
      ]
    }
  },
},
"referencePort":{
```

¹En la versión v2 de laAPI esta variable se llamaba *tides_info*.

```

    "id":REFERENCE_PORT_IDENTIFIER,
    "name":REFERENCE_PORT_NAME,
    "geometry": {
      "type":"Point",
      "coordinates":[
        REFERENCE_PORT_X,
        REFERENCE_PORT_Y
      ]
    },
  },
},
"days":[
  ...
  {
    "timePeriod":{
      "begin": {
        "timeInstant":BEGIN_TIME_PERIOD
      },
      "end": {
        "timeInstant":END_TIME_PERIOD
      }
    },
    "variables":[
      {
        "name":"tides",
        "units":"m", "
        summary":[
          ...
          {
            "id":VALUE_IDENTIFIER,
            "state":TIDES_STATE,
            "timeInstant":LOW_OR_HIGH_TIDES_TIME_INSTANT,
            "height":LOW_OR_HIGH_TIDES_TIDES_HEIGHT,
          },
          ...
        ],
        values":[
          ...
          {
            "timeInstant":TIME_INSTANT,
            "height":TIDES_HEIGHT
          },
          ...
        ]
      }
    ]
  }
  ...
]
}

```

Donde:

- *ID*, *NAME*, *MUNICIPALITY*, *PROVINCE* y *TYPE* son atributos de la localización. Sólo se incluyen cuando la operación se invoca con el parámetro *locationIds*.
- *PORT_IDENTIFIER* es el identificador del puerto del que se toman los datos de pleamar y bajamar.
- *PORT_NAME* es el nombre del puerto del que se toman los datos de pleamar y bajamar.
- *PORT_X* y *PORT_Y* son las coordenadas del puerto del que se toman los datos de pleamar y bajamar (lonxi-

tude y latitud respectivamente).

- *REFERENCE_PORT_IDENTIFIER* es el identificador del puerto de referencia del que se toman las alturas de mar e a cada media hora.
- *REFERENCE_PORT_NAME* es el nombre del puerto de referencia del que se toman las alturas de marea cada media hora.
- *REFERENCE_PORT_X* y *REFERENCE_PORT_Y* son las coordenadas del puerto de referencia del que se toman las alturas de marea cada media hora (longitud y latitud respectivamente).
- *BEGIN_TIME_PERIOD* es, en este caso, el primer instante del día.
- *END_TIME_PERIOD* es, en este caso, o último instante del día.
- *VALUE_IDENTIFIER* es un identificador del dato. Toma valores consecutivos desde 1 (1, 2, 3...).
- *TIDES_STATE* indica si se trata de una pleamar o una bajamar, *Hightides* y *Lowtides* respectivamente.
- *LOW_OR_HIGH_TIDES_TIME_INSTANT* indica el instante en el que tiene lugar una pleamar o una bajamar.
- *LOW_OR_HIGH_TIDES_TIDES_HEIGHT* indica la altura de una pleamar o una bajamar.
- *TIME_INSTANT* indica el instante para el que se da un dato de altura de marea.
- *TIDES_HEIGHT* indica la altura de marea en un instante dado.

En el caso de que no haya datos para un día concreto, se omite el atributo *units*.

■ GML:

Devuelve un documento GML en el que cada elemento *location* tiene la siguiente estructura:

```
<location id=ID name=NAME municipality=MUNICIPALITY province=PROVINCE type=TYPE>
  <geometry>
    <gml:Point srsName=CRS>
      <gml:pos>Y X</gml:pos>
    </gml:Point>
  </geometry>
  <port>
    <id>PORT_IDENTIFIER</id>
    <name>PORT_NAME</name>
    <gml:Point srsName=CRS>
      <gml:pos>PORT_Y PORT_X</gml:pos>
    </gml:Point>
  </port>
  <referencePort>
    <id>REFERENCE_PORT_IDENTIFIER</id>
    <name>REFERENCE_PORT_NAME</name>
    <gml:Point srsName=CRS>
      <gml:pos>REFERENCE_PORT_Y REFERENCE_PORT_X</gml:pos>
    </gml:Point>
  </referencePort>
  <days>
    ...
    <day>
      <gml:TimePeriod>
        <gml:begin>
          <gml:TimeInstant>
            <gml:timePosition>BEGIN_TIME_PERIOD</gml:timePosition>
          </gml:TimeInstant>
        </gml:begin>
        <gml:end>
          <gml:TimeInstant>
```



```

        <gml:timePosition>END_TIME_PERIOD</gml:timePosition>
    </gml:TimeInstant>
</gml:end>
</gml:TimePeriod>
<variables>
    <variable name="tides"units="m">
        <summary>
            ...
            <tideValue id=VALUE_IDENTIFIER state=TIDES_STATE>
                <gml:TimeInstant>
                    <gml:timePosition>LOW_OR_HIGH_TIDES_TIME_INSTANT</gml:timePosition>
                </gml:TimeInstant>
                <height>LOW_OR_HIGH_TIDES_TIDES_HEIGHT</height>
            </tideValue>
            ...
        </summary>
        <values>
            ...
            <hourValue>
                <gml:TimeInstant>
                    <gml:timePosition>TIME_INSTANT</gml:timePosition>
                </gml:TimeInstant>
                <height>TIDES_HEIGHT</height>
            </hourValue>
            ...
        </values>
    </variable>
</variables>
</day>
...
</days>
</location>

```

Donde:

- *ID*, *NAME*, *MUNICIPALITY*, *PROVINCE* y *TYPE* son atributos de la localización. Sólo se incluyen cuando la operación se invoca con el parámetro *locationIds*.
- *CRS* es o sistema de coordenadas utilizado (actualmente es siempre EPSG:4326).
- *Y* y *X* son las coordenadas de la localización (latitud y longitud respectivamente).
- *PORT_IDENTIFIER* es el identificador del puerto del que se toman los datos de pleamar y bajar.
- *PORT_NAME* es el nombre del puerto del que se toman los datos de pleamar y bajar.
- *PORT_Y* y *PORT_X* son las coordenadas del puerto del que se toman los datos de pleamar y bajar (latitud y longitud respectivamente).
- *REFERENCE_PORT_IDENTIFIER* es el identificador del puerto de referencia del que se toman las alturas de marea cada media hora.
- *REFERENCE_PORT_NAME* es el nombre del puerto de referencia del que se toman las alturas de marea cada media hora.
- *REFERENCE_PORT_Y* y *REFERENCE_PORT_X* son las coordenadas del puerto de referencia del que se toman las alturas de marea cada media hora (latitud y longitud respectivamente).
- *BEGIN_TIME_PERIOD* es, en este caso, el primer instante del día.
- *END_TIME_PERIOD* es, en este caso, o último instante del día.

- *VALUE_IDENTIFIER* es un identificador del dato. Toma valores consecutivos desde 1 (1, 2, 3...).
- *TIDES_STATE* indica si se trata d una pleamar o una bajamar, *Hightides* y *Lowtides* respectivamente.
- *LOW_OR_HIGH_TIDES_TIME_INSTANT* indica el instante no que tiene lugar una pleamar o una bajamar.
- *LOW_OR_HIGH_TIDES_TIDES_HEIGHT* indica la altura de una pleamar o una bajamar.
- *TIME_INSTANT* indica el instante para el que se da un dato de altura de marea.
- *TIDES_HEIGHT* indica la altura da marea en un instante dado.

En el caso de que no haya datos para un día concreto, se omite o atributo *units*.

■ **HTML:**

La respuesta es un documento HTML, con el correspondiente CSS, con el siguiente aspecto:

Predicción para latitude=44, lonxitude=-8

Luns abril 29, 2013

	Id	Nome	Latitude	Longitude
Porto	5	Ferrol	43,4667	-8,2333

Estado	Altura	Hora
Low tides	0.9 m	01:13
High tides	4.3 m	07:29
Low tides	1.1 m	13:31
High tides	4.4 m	19:51

	Id	Nome	Latitude	Longitude
Porto de referencia	1	A Coruña	43,35	-8,3833

	00:00	00:30	01:00	01:30	02:00	02:30	03:00	04:30	04:00	04:
Altura	1.266 m	1.049 m	0.944 m	0.953 m	1.07 m	1.284 m	1.579 m	1.935 m	2.331 m	2.74

Martes abril 30, 2013

	Id	Nome	Latitude	Longitude
Porto	5	Ferrol	43,4667	-8,2333

Estado	Altura	Hora
--------	--------	------

■ **KML:**

La respuesta es un documento KML con la Estructura indicada en el apartado *Estructura de los datos devueltos por las operaciones /getNumericForecastInfo, /getTidesInfo y /getSolarInfo*.

7.5 Ejemplos

Obtener los datos de mareas para el puerto más próximo a un punto dado:

https://servizos.meteogalicia.gal/apiv4/getTidesInfo?coords=-8.637,43.45&API_KEY=***

Operación /getSolarInfo

8.1 Introducción

Esta operación ofrece información, para un conjunto de días, sobre las horas de salida y puesta del sol (orto y ocaso respectivamente), así como del mediodía (punto más alto) y de las horas totales de luz. Todo esto para cualquier punto del planeta.

8.2 Rango temporal

El número máximo de días que se puede pedir en una petición es 365. El instante más alejado para el que se devuelven datos es 365 días después del día actual y no se pueden pedir datos anteriores al día actual. Se no se especifica el rango temporal en la petición, se devuelven datos para cinco días desde el día actual. Se sólo se especifica el instante inicial, se devuelven datos de 5 días a partir del día del instante inicial (este incluido). Se sólo se especifica el instante final, se devuelven datos desde el día actual hasta el instante final (ambos incluidos), siempre que a duración no sea superior al máximo permitido. Véanse as Observaciones posteriores sobre los parámetros *startTime* y *endTime*.

8.3 Parámetros

Los parámetros que admite esta operación son los comentados en la *tabla de parámetro comunes*. En el caso de los parámetros *startTime* y *endTime*, tendrán que ser indicados con su formato completo (yyyy-MM-ddTHH:mm:ss) pero únicamente se tendrá en cuenta o año, mes y día.

8.4 Resultados

La estructura de los resultados devueltos es la indicada en el apartado *Estructura de los datos devueltos por las operaciones /getNumericForecastInfo, /getTidesInfo y /getSolarInfo*. Dentro de cada día, el contenido de cada variable es el siguiente:

Nombre	Comentarios
name	Nombre de la variable, <i>solar</i> en este caso.
sunrise	Instante de tiempo en el que se producirá a salida del sol.
midday	Instante de tiempo en el que se producirá el mediodía.
sunset	Instante de tiempo en el que se producirá la puesta del sol.

duration	Tempo de luz solar.
----------	---------------------

Más en detalle para cada formato de salida:

■ JSON:

Devuelve un objeto GeoJSON en el que el objeto *properties* de cada *feature* tiene la siguiente estructura:

```
"properties":{
  "id":ID,
  "name":NAME,
  "municipality":MUNICIPALITY,
  "province":PROVINCE,
  "type":TYPE,
  "days":[
    ...
    {
      "timePeriod":{
        "begin": {
          "timeInstant":BEGIN_TIME_PERIOD
        },
        "end": {
          "timeInstant":END_TIME_PERIOD
        }
      },
      "variables": [
        {
          "name":"solar",
          "sunrise":SUNRISE_TIME_INSTANT,
          "midday":MIDDAY_TIME_INSTANT,
          "sunset":SUNSET_TIME_INSTANT,
          "duration":DURATION
        }
      ]
    }
    ...
  ]
}
```

Donde:

- *ID*, *NAME*, *MUNICIPALITY*, *PROVINCE* y *TYPE* son atributos de la localización. Sólo se incluyen cuando la operación se invoca con el parámetro *locationIds*.
- *BEGIN_TIME_PERIOD* es, en este caso, el primer instante del día.
- *END_TIME_PERIOD* es, en este caso, o último instante del día.
- *SUNRISE_TIME_INSTANT* es el instante en el que se producirá a salida del sol ese día.
- *MIDDAY_TIME_INSTANT* es el instante en el que se producirá o mediodía ese día.
- *SUNSET_TIME_INSTANT* es el instante en el que se producirá la puesta del sol ese día.
- *DURATION* es a duración del día (horas de sol), con formato *Xh Xm*, por ejemplo *9h 12m*.

■ GML:

Devuelve un documento GML en el que cada elemento *location* tiene la siguiente estructura:

```
<locationid=ID name=NAME municipality=MUNICIPALITY province=PROVINCE type=TYPE>
  <geometry>
    <gml:Point srsName=CRS>
      <gml:pos>Y X</gml:pos>
    </gml:Point>
  </geometry>
  <days>
```

```

...
<day>
  <gml:TimePeriod>
    <gml:begin>
      <gml:TimeInstant>
        <gml:timePosition>BEGIN_TIME_PERIOD</gml:timePosition>
      </gml:TimeInstant>
    </gml:begin>
    <gml:end>
      <gml:TimeInstant>
        <gml:timePosition>END_TIME_PERIOD</gml:timePosition>
      </gml:TimeInstant>
    </gml:end>
  </gml:TimePeriod>
  <variables>
    <variablename="solar">
      <sunrise>
        <gml:TimeInstant>
          <gml:timePosition>SUNRISE_TIME_INSTANT</gml:timePosition>
        </gml:TimeInstant>
      </sunrise>
      <midday>
        <gml:TimeInstant>
          <gml:timePosition>MIDDAY_TIME_INSTANT</gml:timePosition>
        </gml:TimeInstant>
      </midday>
      <sunset>
        <gml:TimeInstant>
          <gml:timePosition>SUNSET_TIME_INSTANT</gml:timePosition>
        </gml:TimeInstant>
      </sunset>
      <duration>DURATION</duration>
    </variable>
  </variables>
</day>
...
</days>
</location>

```

Donde:

- *ID*, *NAME*, *MUNICIPALITY*, *PROVINCE* y *TYPE* son atributos de la localización. Sólo se incluyen cuando la operación se invoca con el parámetro *locationIds*.
- *CRS* es o sistema de coordenadas usado (actualmente es siempre EPSG:4326).
- *Y* y *X* son las coordenadas de la localización (latitud y longitud respectivamente).
- *BEGIN_TIME_PERIOD* es en este caso, el primer instante del día.
- *END_TIME_PERIOD* es, en este caso, o último instante del día.
- *SUNRISE_TIME_INSTANT* es el instante en el que se producirá a salida del sol ese día.
- *MIDDAY_TIME_INSTANT* es el instante en el que se producirá o mediodía ese día.
- *SUNSET_TIME_INSTANT* es el instante en el que se producirá la puesta del sol ese día.
- *DURATION* es a duración del día (horas de sol), con formato *Xh Xm*, por ejemplo *9h 12m*.

■ **HTML:**

La respuesta es un documento HTML, con el correspondiente CSS, con el siguiente aspecto:

Predicción para latitude=44, lonxitude=-8

Luns abril 29, 2013

Orto	Mediodía	Ocaso	Duración
07:28:18	14:29:32	21:30:46	14h 2m

Martes abril 30, 2013

Orto	Mediodía	Ocaso	Duración
07:26:48	14:29:23	21:31:58	14h 5m

Mércores maio 01, 2013

Orto	Mediodía	Ocaso	Duración
07:25:20	14:29:15	21:33:10	14h 7m

Xoves maio 02, 2013

Orto	Mediodía	Ocaso	Duración
07:23:54	14:29:08	21:34:22	14h 10m

Venres maio 03, 2013

Orto	Mediodía	Ocaso	Duración
07:22:28	14:29:01	21:35:33	14h 13m

Todas as horas en hora local de Galicia

- **KML:**

La respuesta es un documento KML con la Estructura indicada en el apartado *Estructura de los datos devueltos por las operaciones /getNumericForecastInfo, /getTidesInfo y /getSolarInfo*.

8.5 Observaciones

Esta petición devuelve valores para o punto exacto que se solicita en la petición. Por esta razón, los elementos *variable* no incluyen a mayores o subelemento *geometry* en ninguno de los formatos de salida.

8.6 Ejemplos

Obtener los datos por defecto para un punto determinado:

https://servizos.meteogalicia.gal/apiv4/getSolarInfo?coords=-8.350573861318628,43.3697102138535&API_KEY=***

Obtener los datos para tres días concretos y en formato HTML:

```
https://servizos.meteogalicia.gal/apiv4/getSolarInfo?coords=-  
8.350573861318628,43.3697102138535&format=text/html&startTime=2014-03-07T00:00:00&  
endTime=2014-03-09T00:00:00&API_KEY=***
```

Nota: Para ejecutar esta última petición es necesario ajustar los parámetros *startTime* y *endTime* a valores actuales.

Excepciones

A continuación se indican los tipos de Excepciones contempladas en la API según las operaciones en las que se puede dar cada una de ellas.

Excepciones comunes a todas las operaciones

Código	Descripción
000	Error interno da aplicación o de alguno servidor de datos
001	No se obtuvo la respuesta dentro del tiempo máximo de respuesta
002	Algún de los parámetros indicados no existe o está mal escrito
003	Algún de los parámetros indicados está duplicado
004	Algún de los parámetros indicados está vacío
005	No se encuentra el parámetro <i>API_KEY</i>
006	la <i>API_KEY</i> no es válida
007	El idioma indicado (parámetro <i>lang</i>) no existe, está mal escrito o no está soportado
008	El parámetro <i>format</i> indica un formato no soportado
009	El parámetro <i>exceptionsFormat</i> indica un formato no soportado
010	El sistema de coordenadas indicado (parámetro <i>CRS</i>) no está soportado o es inválido
011	El estilo indicado (parámetro <i>style</i>) no está soportado o es inválido

Excepciones que pueden darse en la operación */findPlaces*

Código	Descripción
100	No se especificó el valor del parámetro <i>location</i>
101	El formato del parámetro <i>types</i> es inválido
102	El parámetro <i>types</i> contiene algún valor inválido

Excepciones que pueden darse en las operaciones */getNumericForecastInfo*, */getTidesInfo* y */getSolarInfo*

Código	Descripción
200	No se especificó nin el parámetro <i>locationIds</i> nin el parámetro <i>coords</i>
201	Se especificaron los parámetros <i>locationIds</i> y <i>coords</i> al mismo tiempo
202	Se indicaron más puntos de los permitidos(actualmente o límite son 20 puntos)
203	Algún de los valores indicados para el parámetro <i>coords</i> está vacío
204	Algún de los valores indicados para el parámetro <i>locationIds</i> está vacío
205	El formato del parámetro <i>coords</i> es inválido
206	Algún de los valores indicados para el parámetro <i>coords</i> es inválido o está mal escrito
207	El formato del parámetro <i>locationIds</i> es inválido
208	Algún de los valores indicados para el parámetro <i>locationIds</i> es inválido o está mal escrito
209	El formato de alguna de las fechas indicadas es inválido (debe ser <i>yyyy-MM-ddTHH:mm:ss</i>)
210	El formato del parámetro <i>tz</i> es inválido
211	No se encontró ningún lugar con el identificador de lugar indicado
212	El instante inicial indicado es posterior al instante final
213	El instante inicial indicado es anterior al día actual
214	El instante final indicado es anterior al día actual
215	El intervalo de tiempo especificado es demasiado grande
216	No hay datos para o intervalo temporal indicado
217	El punto o puntos indicados caen fuera de los límites geográficos para los que hay datos

Excepciones que pueden darse en la operación */getNumericForecastInfo*

Código	Descripción
300	El valor del parámetro <i>variables</i> no es válido
301	El valor del parámetro <i>models</i> no es válido
302	El valor del parámetro <i>grids</i> no es válido
303	El valor del parámetro <i>units</i> no es válido
304	El valor del parámetro <i>autoAdjustPosition</i> no es válido
305	El valor de las unidades indicadas para la variable <i>wind</i> no es válido
306	El número de variables no es igual al número de modelos
307	El número de variables no es igual al número de mallas
308	El número de variables no es igual al número de unidades
309	alguna de las variables indicadas no existen
310	Algún de los modelos indicados no existe
311	Algunas de las mallas indicadas no existe
312	Alguna de las unidades indicadas no existe
313	Hay variables repetidas. En el caso de indicar variables repetidas, se debe indicar un modelo distinto para cada una. Si un par variable-modelo está repetido, para cada un se debe indicar una malla distinta.
314	Se indicaron valores para el parámetro <i>grids</i> sin indicar el modelo correspondiente
315	Un de los modelos indicados no es aplicable para esa variable
316	Una de las mallas indicadas no es aplicable para esa variable
317	Una de las unidades indicadas no es aplicable para esa variable
318	El parámetro <i>endTime</i> indica un instante anterior a la hora actual. En este caso debe incluirse el parámetro <i>startTime</i>
319	El intervalo de tiempo especificado es demasiado pequeño

Excepciones que pueden darse en la operación */getTidesInfo*

Código	Descripción
400	La información sobre los puertos para recuperar datos de mareas no está disponible

Apéndices:

A1. Novedades de la versión v4

Nota: La versión v4 de la API es compatible con versiones anteriores, pero es necesario solicitar una nueva API_KEY.

Estas son algunas de las principales novedades de la versión v4 de la API:

- **Modelo USWAN: Malla no Estructurada del modelo SWAN.**

Las peticiones al modelo SWAN, que en la versión v3 de la API sólo tenía disponibles las mallas Ártabro y RiasBaixas, recomendamos sustituirlas por peticiones al modelo USWAN, malla Galicia, ya que es un modelo con una mejor resolución y que, además, abarca todo el litoral gallego.

Exemplo de petición:

https://servizos.meteogalicia.gal/apiv4/getNumericForecastInfo?coords=-8.393145,43.4372239&models=USWAN&variables=significant_wave_height&grids=Galicia&lang=gl&format=text/html&API_KEY=****

A2. Acerca de este documento

Cambios realizados

- **20/05/2014**
Versión inicial de la guía de la API v3.
- **30/04/2021**
Versión inicial de la guía de la API v4.

A3. Usos horarios

Los usos horarios admitidos (parámetro tz) son los siguientes:

- Europe/Madrid
- UTC
- Europe/Lisbon
- Europe/London
- Europe/Paris
- Etc/GMT+12
- Etc/GMT+11
- Pacific/Midway
- Pacific/Niue
- Pacific/Pago_Pago
- Pacific/Samoa
- US/Samoa
- America/Adak
- America/Atka
- Etc/GMT+10
- HST
- Pacific/Fakaofu
- Pacific/Honolulu
- Pacific/Johnston
- Pacific/Rarotonga
- Pacific/Tahiti
- SystemV/HST10

- US/Aleutian
- US/Hawaii
- Pacific/Marquesas
- AST
- America/Anchorage
- America/Juneau
- America/Nombre
- America/Sitka
- America/Yakutat
- Etc/GMT+9
- Pacific/Gambier
- SystemV/YST9
- SystemV/YST9YDT
- US/Alaska
- America/Dawson
- America/Ensenada
- America/Los_Angeles
- America/Metlakatla
- America/Santa_Isabel
- America/Tijuana
- America/Vancouver
- America/Whitehorse
- Canada/Pacific
- Canada/Yukon
- Etc/GMT+8
- Mexico/BajaNorte
- PST
- PST8PDT

- Pacific/Pitcairn
- SystemV/PST8
- SystemV/PST8PDT
- US/Pacific
- US/Pacific-New
- America/Boise
- America/Cambridge_Bay
- America/Chihuahua
- America/Dawson_Creek
- America/Denver
- America/Edmonton
- America/Hermosillo
- America/Inuvik
- America/Mazatlan
- America/Ojinaga
- America/Phoenix
- America/Shiprock
- America/Yellowknife
- Canada/Mountain
- Etc/GMT+7
- MST
- MST7MDT
- Mexico/BajaSur
- Navajo
- PNT
- SystemV/MST7
- SystemV/MST7MDT
- US/Arizona
- US/Mountain

- America/Bahia_Banderas
- America/Belize
- America/Cancun
- America/Chicago
- America/Costa_Rica
- America/El_Salvador
- America/Guatemala
- America/Indiana/Knox
- America/Indiana/Tell_City
- America/Knox_IN
- America/Managua
- America/Matamoros
- America/Menominee
- America/Merida
- America/Mexico_City
- America/Monterrey
- America/North_Dakota/Beulah
- America/North_Dakota/Center
- America/North_Dakota/New_Salem
- America/Rainy_River
- America/Rankin_Inlet
- America/Regina
- America/Resolute
- America/Swift_Current
- America/Tegucigalpa
- America/Winnipeg
- CST
- CST6CDT

- Canada/Central
- Canada/East-Saskatchewan
- Canada/Saskatchewan
- Chile/EasterIsland
- Etc/GMT+6
- Mexico/General
- Pacific/Easter
- Pacific/Galapagos
- SystemV/CST6
- SystemV/CST6CDT
- US/Central
- US/Indiana-Starke
- America/Atikokan
- America/Bogota
- America/Cayman
- America/Coral_Harbour
- America/Detroit
- America/Fort_Wayne
- America/Grand_Turk
- America/Guayaquil
- America/Havana
- America/Indiana/Indianapolis
- America/Indiana/Marengo
- America/Indiana/Petersburg
- America/Indiana/Vevay
- America/Indiana/Vincennes
- America/Indiana/Winamac
- America/Indianapolis
- America/Iqaluit

- America/Jamaica
- America/Kentucky/Louisville
- America/Kentucky/Monticello
- America/Lima
- America/Louisville
- America/Montreal
- America/Nassau
- America/New_York
- America/Nipigon
- America/Panama
- America/Pangnirtung
- America/Port-au-Prince
- America/Thunder_Bay
- America/Toronto
- Canada/Eastern
- Cuba
- EST
- EST5EDT
- Etc/GMT+5
- IET
- Jamaica
- SystemV/EST5
- SystemV/EST5EDT
- US/East-Indiana
- US/Eastern
- US/Michigan
- America/Caracas
- America/Anguilla
- America/Antigua

- America/Argentina/San_Luis
- America/Aruba
- America/Asuncion
- America/Barbados
- America/Blanc-Sablon
- America/Boa_Vista
- America/Campo_Grande
- America/Cuiaba
- America/Curacao
- America/Dominica
- America/Eirunepe
- America/Glace_Bay
- America/Goose_Bay
- America/Grenada
- America/Guadeloupe
- America/Guyana
- America/Halifax
- America/Kralendijk
- America/La_Paz
- America/Lower_Princes
- America/Manaus
- America/Marigot
- America/Martinique
- America/Moncton
- America/Montserrat
- America/Port_of_Spain
- America/puerto_Acre
- America/puerto_Velho
- America/Puerto_Rico

- America/Rio_Branco
- America/Santiago
- America/Santo_Domingo
- America/St_Barthelemy
- America/St_Kitts
- America/St_Lucia
- America/St_Thomas
- America/St_Vincent
- America/Thule
- America/Tortola
- America/Virgin
- Antarctica/Palmer
- Atlantic/Bermuda
- Atlantic/Stanley
- Brazil/Acre
- Brazil/West
- Canada/Atlantic
- Chile/Continental
- Etc/GMT+4
- PRT
- SystemV/AST4
- SystemV/AST4ADT
- America/St_Johns
- CNT
- Canada/Newfoundland
- AGT
- America/Araguaina
- America/Argentina/Buenos_Aires
- America/Argentina/Catamarca

- America/Argentina/ComodRivadavia
- America/Argentina/Cordoba
- America/Argentina/Jujuy
- America/Argentina/La_Rioja
- America/Argentina/Mendoza
- America/Argentina/Rio_Gallegos
- America/Argentina/Salta
- America/Argentina/San_Juan
- America/Argentina/Tucuman
- America/Argentina/Ushuaia
- America/Bahia
- America/Belem
- America/Buenos_Aires
- America/Catamarca
- America/Cayenne
- America/Cordoba
- America/Fortaleza
- America/Godthab
- America/Jujuy
- America/Maceio
- America/Mendoza
- America/Miquelon
- America/Montevideo
- America/Paramaribo
- America/Recife
- America/Rosario
- America/Santarem
- America/Sao_Paulo
- Antarctica/Rothera

- BET
- Brazil/East
- Etc/GMT+3
- America/Noronha
- Atlantic/South_Georgia
- Brazil/DeNoronha
- Etc/GMT+2
- America/Scoresbysund
- Atlantic/Azores
- Atlantic/Cape_Verde
- Etc/GMT+1
- Africa/Abidjan
- Africa/Accra
- Africa/Bamako
- Africa/Banjul
- Africa/Bissau
- Africa/Casablanca
- Africa/Conakry
- Africa/Dakar
- Africa/El_Aaiun
- Africa/Freetown
- Africa/Lome
- Africa/Monrovia
- Africa/Nouakchott
- Africa/Ouagadougou
- Africa/Sao_Tome
- Africa/Timbuktu
- America/Danmarkshavn
- Atlantic/Canary

- Atlantic/Faeroe
- Atlantic/Faroe
- Atlantic/Madeira
- Atlantic/Reykjavik
- Atlantic/St_Helena
- Eire
- Etc/GMT
- Etc/GMT+0
- Etc/GMT-0
- Etc/GMT0
- Etc/Greenwich
- Etc/UCT
- Etc/UTC
- Etc/Universal
- Etc/Zulu
- Europe/Belfast
- Europe/Dublin
- Europe/Guernsey
- Europe/Isle_of_Man
- Europe/Jersey
- GB
- GB-Eire
- GMT
- GMT0
- Greenwich
- Iceland
- Portugal
- UCT
- Universal

- WET
- Zulu
- Africa/Algiers
- Africa/Bangui
- Africa/Brazzaville
- Africa/Ceuta
- Africa/Douala
- Africa/Kinshasa
- Africa/Lagos
- Africa/Libreville
- Africa/Luanda
- Africa/Malabo
- Africa/Ndjamena
- Africa/Niamey
- Africa/puerto-Novo
- Africa/Tunis
- Africa/Windhoek
- Arctic/Longyearbyen
- Atlantic/Jan_Mayen
- CET
- ECT
- Etc/GMT-1
- Europe/Amsterdam
- Europe/Andorra
- Europe/Belgrade
- Europe/Berlin
- Europe/Bratislava
- Europe/Brussels
- Europe/Budapest

- Europe/Copenhagen
- Europe/Gibraltar
- Europe/Ljubljana
- Europe/Luxembourg
- Europe/Malta
- Europe/Monaco
- Europe/Oslo
- Europe/Podgorica
- Europe/Prague
- Europe/Rome
- Europe/San_Marino
- Europe/Sarajevo
- Europe/Skopje
- Europe/Stockholm
- Europe/Tirane
- Europe/Vaduz
- Europe/Vatican
- Europe/Vienna
- Europe/Warsaw
- Europe/Zagreb
- Europe/Zurich
- MET
- Poland
- ART
- Africa/Blantyre
- Africa/Bujumbura
- Africa/Cairo
- Africa/Gaborone
- Africa/Harare

- Africa/Johannesburg
- Africa/Kigali
- Africa/Lubumbashi
- Africa/Lusaka
- Africa/Maputo
- Africa/Maseru
- Africa/Mbabane
- Africa/Tripoli
- Asia/Amman
- Asia/Beirut
- Asia/Damascus
- Asia/Gaza
- Asia/Hebron
- Asia/Istanbul
- Asia/Jerusalem
- Asia/Nicosia
- Asia/Tel_Aviv
- CAT
- EET
- Egypt
- Etc/GMT-2
- Europe/Athens
- Europe/Bucharest
- Europe/Chisinau
- Europe/Helsinki
- Europe/Istanbul
- Europe/Kiev
- Europe/Mariehamn
- Europe/Nicosia

- Europe/Riga
- Europe/Simferopol
- Europe/Sofia
- Europe/Tallinn
- Europe/Tiraspol
- Europe/Uzhgorod
- Europe/Vilnius
- Europe/Zaporozhye
- Israel
- Libya
- Turkey
- Africa/Addis_Ababa
- Africa/Asmara
- Africa/Asmera
- Africa/Dar_es_Salaam
- Africa/Djibouti
- Africa/Juba
- Africa/Kampala
- Africa/Khartoum
- Africa/Mogadishu
- Africa/Nairobi
- Antarctica/Syowa
- Asia/Aden
- Asia/Baghdad
- Asia/Bahrain
- Asia/Kuwait
- Asia/Qatar
- Asia/Riyadh
- EAT

- Etc/GMT-3
- Europe/Kaliningrad
- Europe/Minsk
- Indian/Antananarivo
- Indian/Comoro
- Indian/Mayotte
- Asia/Riyadh87
- Asia/Riyadh88
- Asia/Riyadh89
- Mideast/Riyadh87
- Mideast/Riyadh88
- Mideast/Riyadh89
- Asia/Tehran
- Iran
- Asia/Baku
- Asia/Dubai
- Asia/Muscat
- Asia/Tbilisi
- Asia/Yerevan
- Etc/GMT-4
- Europe/Moscow
- Europe/Samara
- Europe/Volgograd
- Indian/Mahe
- Indian/Mauritius
- Indian/Reunion
- NET
- W-SU
- Asia/Kabul

- Antarctica/Mawson
- Asia/Aqtau
- Asia/Aqtobe
- Asia/Ashgabat
- Asia/Ashkhabad
- Asia/Dushanbe
- Asia/Karachi
- Asia/Oral
- Asia/Samarkand
- Asia/Tashkent
- Etc/GMT-5
- Indian/Kerguelen
- Indian/Maldives
- PLT
- Asia/Calcutta
- Asia/Colombo
- Asia/Kolkata
- IST
- Asia/Kathmandu
- Asia/Katmandu
- Antarctica/Vostok
- Asia/Almaty
- Asia/Bishkek
- Asia/Dacca
- Asia/Dhaka
- Asia/Qyzylorda
- Asia/Thimbu
- Asia/Thimphu
- Asia/Yekaterinburg

- BST
- Etc/GMT-6
- Indian/Chagos
- Asia/Rangoon
- Indian/Cocos
- Antarctica/Davis
- Asia/Bangkok
- Asia/Ho_Chi_Minh
- Asia/Hovd
- Asia/Jakarta
- Asia/Novokuznetsk
- Asia/Novosibirsk
- Asia/Omsk
- Asia/Phnom_Penh
- Asia/Pontianak
- Asia/Saigon
- Asia/Vientiane
- Etc/GMT-7
- Indian/Christmas
- VST
- Antarctica/Casey
- Asia/Brunei
- Asia/Choibalsan
- Asia/Chongqing
- Asia/Chungking
- Asia/Harbin
- Asia/Hong_Kong
- Asia/Kashgar
- Asia/Krasnoyarsk

- Asia/Kuala_Lumpur
- Asia/Kuching
- Asia/Macao
- Asia/Macau
- Asia/Makassar
- Asia/Manila
- Asia/ShangHay
- Asia/Singapore
- Asia/Taipei
- Asia/Ujung_Pandang
- Asia/Ulaanbaatar
- Asia/Ulan_Bator
- Asia/Urumqi
- Australia/Perth
- Australia/West
- CTT
- Etc/GMT-8
- Hongkong
- PRC
- Singapore
- Australia/Eucla
- Asia/Dili
- Asia/Irkutsk
- Asia/Jayapura
- Asia/Pyongyang
- Asia/Seoul
- Asia/Tokyo
- Etc/GMT-9
- JST

- Japan
- Pacific/Palau
- ROK
- ACT
- Australia/Adelaide
- Australia/Broken_Hill
- Australia/Darwin
- Australia/North
- Australia/South
- Australia/Yancowinna
- AET
- Antarctica/DumontDUrville
- Asia/Yakutsk
- Australia/ACT
- Australia/Brisbane
- Australia/Canberra
- Australia/Currie
- Australia/Hobart
- Australia/Lindeman
- Australia/Melbourne
- Australia/NSW
- Australia/Queensland
- Australia/Sydney
- Australia/Victoria
- Etc/GMT-10
- Pacific/Chuuk
- Pacific/Guam
- Pacific/Port_Moresby
- Pacific/Saipan

- Pacific/Truk
- Pacific/Yap
- Australia/LHI
- Australia/Lord_Howe
- Antarctica/Macquarie
- Asia/Sakhalin
- Asia/Vladivostok
- Etc/GMT-11
- Pacific/Efate
- Pacific/Guadalcanal
- Pacific/Kosrae
- Pacific/Noumea
- Pacific/Pohnpei
- Pacific/Ponape
- SST
- Pacific/Norfolk
- Antarctica/McMurdo
- Antarctica/South_Pole
- Asia/Anadyr
- Asia/Kamchatka
- Asia/Magadan
- Etc/GMT-12
- Kwajalein
- NST
- NZ
- Pacific/Auckland
- Pacific/Fiji
- Pacific/Funafuti
- Pacific/Kwajalein

- Pacific/Majuro
- Pacific/Nauru
- Pacific/Tarawa
- Pacific/Wake
- Pacific/Wallis
- NZ-CHAT
- Pacific/Chatham
- Etc/GMT-13
- MIT
- Pacific/Apia
- Pacific/Enderbury
- Pacific/Tongatapu
- Etc/GMT-14
- Pacific/Kiritimati

