



RED ELÉCTRICA DE ESPAÑA

Protocolo de Comunicaciones entre
Concentradores de Medidas.
Solución basada en Web Services

Especificación Funcional

Abril de 2017
Versión 3.0 RC 5



Versión	Fecha	Modificaciones
1	10/04/2003	Combinación del documento elaborado por Indra con el de bit MAILER (creación del documento).
1.1	10/04/2003	Correcciones según registro de cambios
1.2	30/04/2003	Correcciones debidas a la reunión del 29/04/2003
1.3	28/07/2003	Modificaciones después de las primeras pruebas.
2.0	18/12/2003	Modificaciones para cerrar el documento
2.1	25/03/2004	3.5.1: Se cambia el Identificador del Concentrador en la cabecera de los mensajes SOAP de OPCIONAL a OBLIGATORIO 4.1.5: Se añaden nuevos códigos de retorno de la Operación EnviarFichero (códigos 3, 4, 5 y 6) Corrección de erratas
2.2	10/02/2010	Actualización de ejemplos de mensajes SOAP.
3.0 RC 1	14/11/2014	Se renueva el protocolo adaptando los servicios web al estándar publicado en la TS IEC 62325-504
3.0 RC 2	19/10/2015	Se modifica la numeración de las versiones 3.X del documento para reflejar las Release Candidates previas a la puesta en servicio del protocolo en su versión final 3.0. Se incluye un listado de errores y se actualiza la gestión de los mismos. Se actualizan los protocolos de seguridad. Se establece en 50 Mb el tamaño máximo de fichero que se permite transferir en una única invocación del servicio <i>Get Message</i> .
3.0 RC 3	08/03/2016	Se establecen valores definitivos para la parametrización detallada en el <i>Anexo 4: Guía de Buenas Prácticas</i> . Se actualiza la lista de errores recogidos en el <i>Anexo 5: Lista de errores</i> devueltos por el servidor.
3.0 RC 4	24/10/2016	Se modifica la redacción del apartado <i>4.5 Identificación de Concentradores</i> , donde se especifica que los certificados de servidor para la comunicación con el CP serán emitidos sólo por Red Eléctrica de España.
3.0 RC 5	17/04/2017	Se añaden dos nuevas referencias al final del apartado 2. <i>Referencias</i> . Se actualiza el apartado <i>4.3 Localización de esquemas</i> con la URL definitiva de descarga de los esquemas usados por el protocolo. Se corrigen referencias erróneas y se actualiza la lista de errores en el <i>Anexo 5: Lista de errores</i> .



Índice

1. INTRODUCCIÓN.....	3
1.1 Objetivos	3
1.2 Convenciones del documento.....	3
1.3 Visión general.....	3
2. REFERENCIAS.....	5
3. DIAGRAMA DE FRONTERA	6
4. ESPECIFICACIÓN DE LOS MENSAJES.....	7
4.1 SOAP	7
4.2 Definición del Atributo Namespace	7
4.3 Localización de esquemas	7
4.4 Definición de esquemas	7
4.5 Identificación de Concentradores.....	7
4.6 Retorno de Errores.....	8
5. DESCRIPCIÓN DE LOS SERVICIOS WEB	9
5.1 Petición de Fecha y Hora.....	9
5.2 List Messages	10
5.3 Get Message.....	15
ANEXO 1: FORMATO DE LA INFORMACIÓN	19
ANEXO 2: TRATAMIENTO DE FICHEROS.....	20
Introducción	20
División de Ficheros	20
ANEXO 3: DESCRIPCIÓN WDSL DE LOS SERVICIOS WEB	21
ANEXO 4: GUÍA DE BUENAS PRÁCTICAS	22
ANEXO 5: LISTA DE ERRORES DEVUELTOS POR EL SERVIDOR	23



1. INTRODUCCIÓN

1.1 Objetivos

El objetivo de este documento es recoger la especificación funcional del nuevo protocolo de comunicaciones entre concentradores de medidas basado en la TS IEC 62325-504, basada a su vez en el estándar IEC-61968-100. En el documento se detalla la siguiente información:

- Diagrama de frontera del prototipo que representa a los principales actores con los que se relaciona.
- Especificación de funciones, parámetros y tipo de parámetros
- Descripción y ejemplos de los casos de uso.

1.2 Convenciones del documento

Las palabras que vienen descritas en formato Arial en mayúsculas deben ser interpretadas de forma especial. Las palabras clave que se utilizan en el documento son: DEBE, NO DEBE, RECOMENDAR, PUEDE.

- DEBE: esta palabra corresponde con los adjetivos “REQUERIDO” u “OBLIGATORIO”, significa que la definición es imprescindible en la especificación.
- NO DEBE: significa que la definición está prohibida en la especificación.
- RECOMENDAR: o el adjetivo “RECOMENDADO” significa que pueden existir varias razones para ignorar el elemento pero que en algunas ocasiones es bueno que aparezca dicho elemento.
- PUEDE: esta palabra o el adjetivo “OPCIONAL”, significa que el elemento es completamente opcional. Nos da la posibilidad de utilizarlo o no según convenga.

Los argumentos de entrada de los servicios se marcarán como:

- Ob: Obligatorio. Es necesario invocar el servicio especificando un valor para este argumento.
- Op: Opcional. No es necesario el uso del argumento en la invocación del servicio.
- Sn: Selección n. Si varios de los argumentos opcionales están marcados con el mismo número “n” en “Sn” dichos argumentos deben estar presentes de forma indivisible en la invocación del servicio.

1.3 Visión general

Los objetivos que se pretenden cubrir con el protocolo de comunicaciones de Concentradores de Medida son los siguientes:

- Proporcionar una forma de comunicación entre Concentradores de Medida basada en estándares.
- Proporcionar una forma de comunicación que sea aplicable a todas las comunicaciones entre los distintos tipos de Concentradores de Medida del mercado eléctrico.
- Proporcionar una especificación del protocolo que permita su implementación por terceros que deseen comunicar con los Concentradores de Medida del mercado



eléctrico.

El protocolo dispondrá de la siguiente funcionalidad básica:

- Consulta de fecha y hora en servidor: servicio QueryData.
- Consulta de información disponible en concentradores remotos: servicio List Messages.
- Recogida de información disponible en concentradores remotos: servicio Get Message.



2. REFERENCIAS

- [1] TS IEC 62325-504: Electronic data interchanges on the Internal Electricity Market
- [2] WS-I Basic Profile 1.1, <http://www.ws-i.org/Profiles/BasicProfile-1.1-2006-04-10.html>
- [3] WSDL, *Web Services Description Language (WSDL) 1.1*
- [4] IEC-40210, *W3C SOAP Version 1.2 Part I: Messaging Framework (2nd Edition)*
- [5] XML Schema 1.0: XML Schema Language Part 1: Structure, W3C Recommendation 28 October 2004; XML Schema Language Part 2: Data Types, W3C Recommendation 28 October 2004
- [6] XML Signature Syntax and Processing Version 2.0 (<http://www.w3.org/TR/xmlsig-core2/>)
- [7] IEC-62325 Part 451-1: *Framework for energy market communications - Acknowledgement business process and contextual model for CIM European market.*
- [8] IEC-61968-100: Implementation Profiles for IEC 61968
- [9] Ficheros para el intercambio de información de medida (Versión 20. Junio 2014)
- [10] RFC-7231: Hypertext Transfer Protocol (HTTP/1.1) (<https://tools.ietf.org/html/rfc7231>)
- [11] Guía de acceso a los sistemas e-sios (<https://api.esios.ree.es/documents/244/download?locale=en>)
- [12] IEC TS 62325-504:2015 Utilization of web services for electronic data interchanges on the European energy Market for electricity.
- [13] Additional HTTP Status Codes. (<https://tools.ietf.org/html/rfc6585>)



3. DIAGRAMA DE FRONTERA

A continuación se muestra el diagrama de frontera del protocolo de comunicaciones entre Concentradores de Medida.



En este diagrama se identifican los siguientes actores:

- **Cliente.** Son aplicaciones que utilizan la funcionalidad aportada por el protocolo para establecer comunicaciones con otros Concentradores de Medidas servidores. Estos actores serán aplicaciones que utilizarán las librerías de invocación de servicios.
- **Autoridad Certificadora.** Será la encargada de proporcionar los certificados necesarios para establecer la comunicación https entre los Concentradores de Medidas y firmar los mensajes intercambiados.
- **Servidor.** Son los sistemas externos con los que el protocolo debe soportar la comunicación.



4. ESPECIFICACIÓN DE LOS MENSAJES

En este apartado describiremos la estructura de los mensajes del protocolo para el intercambio de información entre concentradores.

4.1 SOAP

Las llamadas a los distintos servicios SOAP que comprenden el protocolo se realizarán utilizando exclusivamente como capa de transporte HTTP. Para asegurar la seguridad y la autenticidad de las partes previamente se habrá establecido una sesión segura (HTTPS) mediante alguno de los protocolos de acceso especificados en [11]. Para el establecimiento de esta sesión el servidor y el cliente dispondrán de un certificado digital.

La codificación de los mensajes SOAP será con estilo “document-style” con las diferentes partes de los mensajes definidas por los esquemas XML del mensaje (<body use=”literal”>).

Los ficheros de información intercambiados mediante el servicio *Get Message* irán codificados en Base64 embebidos en el propio mensaje XML.

4.2 Definición del Atributo Namespace

Los espacios de nombres que se utilizarán en el intercambio de información se definen en [1].

4.3 Localización de esquemas

Los esquemas utilizados en este intercambio de información estarán disponibles en:

<https://api.esios.ree.es/documents/326/download>

4.4 Definición de esquemas

Los esquemas de los servicios web disponibles se encuentran especificados en la documentación referenciada [1] y [8].

4.5 Identificación de Concentradores

Se define un mecanismo OBLIGATORIO de identificación de Concentrador, basado en la utilización del certificado digital usado para el establecimiento de la conexión segura entre el cliente y el servidor.

Para cada concentrador de medidas, Red Eléctrica de España emitirá un certificado para la parte servidora y un certificado para la parte cliente. Para los agentes que dispongan sólo de parte cliente se emitirá sólo un certificado de cliente.



El certificado digital será usado también para la elaboración de la firma de los mensajes XML intercambiados entre los distintos actores cliente y servidor, cuando aplique, de acuerdo con lo definido en [1].

Los certificados tendrán incluido el punto de distribución de las CRL's.

Se recomienda y es responsabilidad del Servidor de la Aplicación que utiliza este protocolo, el implementar un mecanismo de gestión de los clientes a los que permitirá conectarse.

4.6 Retorno de Errores

Para notificar los posibles errores que se produzcan en los servicios web a los clientes que los invoquen se utilizará el esquema *FaultMessage* especificado en el apartado 6.3.6 de [8].

El elemento *FaultMessage* define los siguientes subelementos:

- *Error.code*: Este elemento se utiliza para suministrar una forma de identificar el fallo por medios algorítmicos. Este elemento DEBE estar presente en el elemento *FaultMessage*.
- *Error.details*: Este elemento proporciona una descripción legible por el ser humano del fallo. DEBE estar presente en el elemento *FaultMessage* y deberá proporcionar, al menos, información explicativa de la naturaleza del fallo.

Los servicios descritos en esta especificación generarán los errores indicados en el anexo 5.

En caso de ocurrir un error en la recepción del fichero el concentrador solicitante publicará al concentrador emisor un fichero de tipo "NOOK". El nombre de este fichero será el mismo que el que tenga el fichero recibido, añadiendo la extensión ".NOOK". El fichero contendrá un string indicando el tipo de error encontrado:

- Firma del mensaje incorrecta.
- Tipo de fichero no identificado.

Dada la naturaleza web de los servicios, todos los errores http son de aplicación. Destacamos entre ellos:

- HTTP-401 (Unauthorized): El Usuario no está autorizado para acceder al servidor o enviar información a este. Este error se produce cuando el usuario no tiene los roles necesarios para acceder a los recursos. También se producirá si está utilizando un certificado caducado.
- HTTP-403 (Forbidden): El Usuario no tiene acceso al servidor. Este error se produce típicamente cuando el usuario no hace uso de su certificado o este no es reconocido como válido por el servidor.

Consúltese a este efecto [10] "Response Status Codes".



5. DESCRIPCIÓN DE LOS SERVICIOS WEB

Los argumentos y retornos de los servicios web que conforman el protocolo consisten en estructuras XML. Estos argumentos se validarán contra los esquemas XSD indicados en el apartado 4.4.

La validación consistirá en una comprobación de que el argumento o retorno está bien formado conforme a las reglas de XML, y además se realizará la comprobación de que cumple con las restricciones expresadas en el correspondiente esquema XSD.

Procedemos a detallar cada uno de los servicios Web que componen el protocolo.

5.1 *Petición de Fecha y Hora*

Las peticiones de fecha y hora las atenderá el servicio web *QueryData* definido en [1].

Argumentos: El valor del argumento *DataType* será igual a la cadena "serverTimestamp".

Retorno: La operación de consulta retornará la fecha en el elemento *Timestamp* de la cabecera del mensaje de respuesta.

Ejemplo

La llamada a la función debe responder al siguiente esquema:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
...
  <soap:Body>
    <msg:RequestMessage xmlns:msg="http://iec.ch/TC57/2011/schema/message">
      <msg:Header>
        <msg:Verb>get</msg:Verb>
        <msg:Noun>QueryData</msg:Noun>
        <msg:Context>PRODUCTION</msg:Context>
      </msg:Header>
      <msg:Request>
        <msg:Option>
          <msg:name>DataType</msg:name>
          <msg:value>serverTimestamp</msg:value>
        </msg:Option>
      </msg:Request>
    </msg:RequestMessage>
  </soap:Body>
</soap:Envelope>
```

Este es el retorno que se obtendría al hacer una llamada al servicio de petición de Fecha y Hora:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <msg:ResponseMessage xmlns:msg="http://iec.ch/TC57/2011/schema/message">
      <msg:Header>
        <msg:Verb>reply</msg:Verb>
        <msg:Noun>QueryData</msg:Noun>
        <msg:Context>PRODUCTION</msg:Context>
        <msg:Timestamp>2014-07-01T09:30:47Z</msg:Timestamp>
        <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
          ...
        </Signature>
      </msg:Header>
    </msg:ResponseMessage>
  </soap:Body>
</soap:Envelope>
```



```

        </Signature>
    </msg:Header>
    <msg:Reply>
        <msg:Result>OK</msg:Result>
    </msg:Reply>
    <msg:Payload>
        <QueryData xmlns="urn:iec62325.504:messages:1:0">
            <RequestParameters>
                <Parameter>
                    <name>DataType</name>
                    <value>serverTimestamp</value>
                </Parameter>
            </RequestParameters>
        </QueryData>
    </msg:Payload>
</msg:ResponseMessage>
</soap:Body>
</soap:Envelope>
    
```

Secuencia de eventos

Acciones sobre el Protocolo	Acciones del Protocolo
<ol style="list-style-type: none"> 1. Un Concentrador de Medidas remoto inicia una conexión https con el servidor del protocolo. 4. El Concentrador de Medidas remoto invoca al servicio peticiónFechaHora 7. El Concentrador de Medidas cierra la conexión https. 	<ol style="list-style-type: none"> 2. El sistema verifica la identidad del Concentrador. 3. El sistema acepta la conexión. 5. El sistema obtiene la fecha y hora de la máquina en la que se ejecuta. 6. El sistema envía la fecha y la hora obtenidas al Concentrador de Medidas que la solicitó.
<p>Secuencia alternativa de eventos: No se puede verificar identidad del Concentrador.</p>	
Acciones sobre el Protocolo	Acciones del Protocolo
<ol style="list-style-type: none"> 4. El Concentrador de Medidas cierra la conexión https 	<ol style="list-style-type: none"> 3. El sistema rechaza la conexión con un error http-401 / http-403.

5.2 List Messages

Las peticiones de listado de ficheros publicados las atenderá el servicio web *List Messages*.

El esquema de este servicio está especificado en el apartado 7.2 de [8].



Argumentos:

Argumento	Tipo	Ob/Op/S	Descripción
<i>StartTime</i>	dateTime	S1	Inicio del periodo de tiempo definido en el <i>IntervalType</i> , en hora UTC.
<i>EndTime</i>	dateTime	S1	Fin del periodo de tiempo definido en el <i>IntervalType</i> , en hora UTC.
<i>IntervalType</i>	string	S1	Indica si el intervalo de tiempo definido por el <i>StartTime</i> y el <i>EndTime</i> hace referencia al periodo de aplicación de los datos (Application, valor por defecto) o al periodo de publicación de los ficheros listados (Server).
<i>Code</i>	number	S2	Código numérico asignado internamente a cada fichero publicado.
<i>MessageIdentification</i>	string	Op	Nombre completo del fichero. Admite el carácter comodín "*".
<i>MsgType</i>	string	Op	Tipo de fichero. Los tipos de ficheros disponibles son los indicados en [9], en el campo "Tipo de fichero" para cada fichero especificado
<i>Owner</i>	string	Op	Código de participante al que pertenecen los datos contenidos en el fichero.

En caso de invocar el servicio usando los argumentos del grupo S1, y teniendo en cuenta la existencia de ficheros para los que el servidor desconoce el periodo concreto de aplicación de sus datos, se recomienda utilizar el valor *IntervalType* = *Server*.

La siguiente tabla muestra los tipos de fichero (MsgType) posibles:

Tipo	Descripción
AGR	<ul style="list-style-type: none"> Definición de Agregaciones (3.2.1) ¹ Baja de Agregaciones (3.2.2)
CUR	<ul style="list-style-type: none"> Curva de Carga Horaria de Punto de Medida (3.1.1) Curva de Carga Horaria de Punto Frontera (3.1.2) Curva de Carga Cuarto Horaria de Punto de Medida (3.1.3)
FIR	<ul style="list-style-type: none"> Firmas de Curvas de Carga Horaria de Punto de Medida (3.1.13)
INV	<ul style="list-style-type: none"> Inventario de Puntos de Medidas de Clientes (3.3.2) Solicitud de Modificación de Inventario de Puntos de Medidas de Clientes (3.3.3) Inventario de Equipos de Medida de Clientes (3.3.5) Solicitud de Modificación de Inventario de Equipos de Clientes (3.3.6) Notificación Fecha de Lectura (3.3.9)
INC	<ul style="list-style-type: none"> Incidencia en punto de medida (3.1.7) Eventos de Registrador (3.1.8) Anulación de medidas en Punto Frontera (3.1.18)

¹ Los capítulos entre paréntesis se refieren al documento *Ficheros para el intercambio de información de medida* [9]



MAG	<ul style="list-style-type: none"> • Medidas Agregadas (3.2.3) • Medidas agregadas formato de longitud variable (3.2.4)
NOK	<ul style="list-style-type: none"> • Ficheros de error en la recepción
OBJ	<ul style="list-style-type: none"> • Objeciones de Medidas Horarias(3.1.9) • Objeciones a Medidas no Horarias(3.1.11)
OSA	<ul style="list-style-type: none"> • Acumulados Mensuales de Energía por Punto Frontera y Magnitud del Participante 1 (3.4.3.1) • Acumulados Mensuales de Energía por Punto Frontera y Magnitud del Participante 2 (3.4.3.2) • Acumulados Mensuales de Energía por Punto Frontera y Magnitud del Participante 2 Correspondientes a Puntos Frontera de un Participante 1 (3.4.3.3)
OSD	<ul style="list-style-type: none"> • Demanda del Sistema (3.5.1.) • Perfiles Finales (3.5.2).
OSE	<ul style="list-style-type: none"> • Ficheros de error de datos enviados al OS
OSG	<ul style="list-style-type: none"> • Inventarios de agregaciones tipo 3 y siguientes (3.4.4.1) • Acumulados mensuales de energía por agregación del participante 1 (distribuidora) (3.4.4.2) • Acumulados mensuales de energía por agregación del participante 2 (comercializadora) (3.4.4.3) • Acumulados mensuales de energía por agregación del participante 1 (distribuidor) correspondientes a agregaciones de un participante 2 (comercializadores) (3.4.4.4) • Datos horarios diarios de energía por agregación del participante 1 (distribuidor) (3.4.4.5) • Datos horarios diarios de energía por agregación del participante 2 (comercializadora) (3.4.4.6) • Datos horarios diarios de energía por agregación del participante 1 (distribuidor) correspondientes a agregaciones de un participante 2 (comercializadora) (3.4.4.7)
OSI	<ul style="list-style-type: none"> • Datos de Inventario OS. Varios ficheros de acuerdo 3.4.1
OSP	<ul style="list-style-type: none"> • Datos Horarios de Energía por Punto Frontera y Magnitud del Participante 1(3.4.2.1) • Datos Horarios de Energía por Punto Frontera y Magnitud del Participante 2 (3.4.2.2) • Datos Horarios de Energía por Punto Frontera y Magnitud del Participante 2 Correspondientes a Puntos Frontera de un Participante 1 (3.4.2.3.) • Datos Horarios de las Pérdidas de Transporte (3.4.2.4).
PTE	<ul style="list-style-type: none"> • Medidas de Punto de Horarias Pendientes(3.1.14) • Saldos de Contador Pendientes(3.1.17) • Firmas de Puntos de Medida Pendientes(3.1.15)



ROB	<ul style="list-style-type: none"> • Respuesta de Objeciones Horarias (3.1.10) • Respuestas a Objeciones de Medidas no Horarias (3.1.12).
TAR	<ul style="list-style-type: none"> • Cierres de Contrato 1, 2 y 3 de Punto de Medida (3.1.4) • Lecturas Instantáneas de Contrato 1, 2 y 3 de Punto de Medida(3.1.5) • Saldo de Lectura de Facturación (3.1.6)

Retorno: El servicio devolverá una lista de ficheros disponibles para el cliente que lo invoca conforme a los filtros especificados en la petición. Cada fichero vendrá definido por la siguiente información:

1. *Code*: Código numérico asignado internamente al fichero en el servidor.
2. *MessageIdentification*: Nombre completo del fichero (número de versión y extensión incluidos).
3. *Status*: "OK" para ficheros correctos.
4. *ApplicationTimeInterval Start*: Fecha y hora de inicio del periodo al que aplica el contenido del fichero.
5. *ApplicationTimeInterval End*: Fecha y hora de fin del periodo al que aplica el contenido del fichero.
6. *ServerTimeStamp*: Fecha y hora de publicación del fichero en el servidor.
7. *Type*: Tipo al que pertenece el fichero.
8. *Owner*: Código de participante al que pertenecen los datos contenidos en el fichero.

Si no se utiliza el filtro *MsgType*, se responderá con un listado de todos los ficheros de los que disponga el servidor para el cliente que invoca el servicio.

Ejemplo

Se muestra un ejemplo de llamada:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
...
  <soap:Body>
    <msg:RequestMessage xmlns:msg="http://iec.ch/TC57/2011/schema/message">
      <msg:Header>
        <msg:Verb>get</msg:Verb>
        <msg:Noun>MessageList</msg:Noun>
        <msg:Context>PRODUCTION</msg:Context>
        <msg:Timestamp>2014-09-30T09:30:47.581Z</msg:Timestamp>
        <msg:AckRequired>>true</msg:AckRequired>
      </msg:Header>
      <msg:Request>
        <msg:StartTime>2014-05-19T22:00:00Z</msg:StartTime>
        <msg:EndTime>2014-05-20T22:00:00Z</msg:EndTime>
        <msg:Option>
          <msg:name>IntervalType</msg:name>
          <msg:value>Server</msg:value>
        </msg:Option>
        <msg:Option>
          <msg:name>MsgType</msg:name>
          <msg:value>OSP</msg:value>
        </msg:Option>
      </msg:Request>
    </msg:RequestMessage>
  </soap:Body>
```



```
</soap:Envelope>
```

Este es el retorno que se obtendría al hacer una llamada al servicio cuando no hay ningún fichero del tipo solicitado:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <msg:ResponseMessage xmlns:msg="http://iec.ch/TC57/2011/schema/message">
      <msg:Header>
        <msg:Verb>reply</msg:Verb>
        <msg:Noun>MessageList</msg:Noun>
        <msg:Context>PRODUCTION</msg:Context>
      </msg:Header>
      <msg:Reply>
        <msg:Result>OK</msg:Result>
      </msg:Reply>
      <msg:Payload>
        <urn:MessageList xmlns:urn="urn:iec62325.504:messages:1:0">
          </urn:MessageList>
        </msg:Payload>
      </msg:ResponseMessage>
    </soap:Body>
```

Este es el retorno de la función cuando hay ficheros del tipo solicitado:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <msg:ResponseMessage xmlns:msg="http://iec.ch/TC57/2011/schema/message">
      <msg:Header>
        <msg:Verb>reply</msg:Verb>
        <msg:Noun>MessageList</msg:Noun>
        <msg:Context>PRODUCTION</msg:Context>
      </msg:Header>
      <msg:Reply>
        <msg:Result>OK</msg:Result>
      </msg:Reply>
      <msg:Payload>
        <urn:MessageList xmlns:urn="urn:iec62325.504:messages:1:0">
          <urn:Message>
            <urn:Code>879021</urn:Code>
            <urn:MessageIdentification>
ACUM_HC_CLE_1111_P1_201212.1</urn:MessageIdentification>
              <urn:Status>OK</urn:Status>
              <urn:ApplicationTimeInterval>
                <urn:start>2014-05-19T22:00:00Z</urn:start>
                <urn:end>2014-05-20T22:00:00Z</urn:end>
              </urn:ApplicationTimeInterval>
              <urn:ServerTimestamp> 2014-05-
21T06:33:40Z</urn:ServerTimestamp>
              <urn:Type>OSP</urn:Type>
              <urn:Owner>1111</urn:Owner>
            </urn:Message>
            ...
          </urn:MessageList>
        </msg:Payload>
      </msg:ResponseMessage>
    </soap:Body>
```

Secuencia de eventos

Acciones del cliente	Acciones del servidor
1. Un concentrador de medidas remoto inicia una conexión https como cliente con el servidor del protocolo.	2. El servidor verifica la identidad del concentrador mediante el certificado digital usado para el establecimiento de la conexión.



<p>4. El concentrador de medidas remoto invoca el servicio List Messages indicando los parámetros con los que desea restringir la búsqueda de ficheros publicados en el servidor.</p> <p>8. El concentrador de medidas recibe el mensaje de respuesta y cierra la conexión https con el servidor.</p>	<p>3. El servidor acepta la conexión.</p> <p>5. El servidor consulta el listado de ficheros disponibles para el concentrador de medidas que satisfacen los filtros especificados en los argumentos de la petición.</p> <p>6. El servidor elabora un listado con los ficheros devueltos en la búsqueda y lo encapsula en el mensaje de respuesta a la petición del concentrador de medidas.</p> <p>7. El servidor envía el mensaje de respuesta al concentrador de medidas.</p>
<p>Secuencia alternativa de eventos: No se puede verificar la identidad del concentrador secundario.</p>	
<p>4. El concentrador de medidas cierra la conexión https con el servidor.</p>	<p>3. El servidor rechaza la conexión con un error http-401 / http-403.</p>

5.3 Get Message

Las peticiones de recoger fichero las atenderá el servicio web *get Message*.

Argumentos:

El argumento *Queue* definido en [1] no será implementado. Dado que la versión del mensaje está incluida en el identificador del mismo, se desaconseja el uso del argumento *MessageVersion*².

Argumento	Tipo	Ob/Op/S	Descripción
<i>MessageIdentification</i>	String	S1	Es un string que contiene el nombre completo del fichero que se desea solicitar, con el número de versión y la extensión ok/bad/bad2/N_T en caso de tenerla.
<i>Code</i>	number	S2	Código numérico asignado internamente a cada fichero publicado.

Retorno: El servidor devolverá la siguiente información en el XML de respuesta:

1. *BinaryName*: Nombre del fichero devuelto.
2. *BinaryContent*: El fichero solicitado encapsulado en el elemento *Payload* del

² En las solicitudes de mensajes que incluyan valor de versión, se comprobará la correspondencia de éste con el nombre del fichero, retornando un error de "mensaje inexistente" en caso de discordancia.



propio mensaje XML, codificado en Base64.

El fichero estará comprimido en formato bzip2.

En el caso de que se solicite recoger un fichero que no existe, se retornará un mensaje de fault indicando esta circunstancia.

El mensaje de respuesta contendrá un solo elemento Payload, en el cual estará el fichero solicitado.

Ejemplo

La llamada a la función usando el argumento Code debe responder al siguiente esquema:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <msg:RequestMessage xmlns:msg="http://iec.ch/TC57/2011/schema/message">
      <msg:Header>
        <msg:Verb>get</msg:Verb>
        <msg:Noun>Any</msg:Noun>
        <msg:Context>PRODUCTION</msg:Context>
        <msg:Timestamp>2014-09-30T09:30:47.581Z</msg:Timestamp>
        <msg:AckRequired>>true</msg:AckRequired>
      </msg:Header>
      <msg:Request>
        <msg:Option>
          <msg:name>Code</msg:name>
          <msg:value>879021</msg:value>
        </msg:Option>
      </msg:Request>
    </msg:RequestMessage>
  </soap:Body></soap:Envelope>
```

La llamada a la función usando los argumentos de nombre y versión del mensaje debe responder al siguiente esquema:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <msg:RequestMessage xmlns:msg="http://iec.ch/TC57/2011/schema/message">
      <msg:Header>
        <msg:Verb>get</msg:Verb>
        <msg:Noun>Any</msg:Noun>
        <msg:Context>PRODUCTION</msg:Context>
        <msg:Timestamp>2014-09-30T09:30:47.581Z</msg:Timestamp>
        <msg:AckRequired>>true</msg:AckRequired>
      </msg:Header>
      <msg:Request>
        <msg:Option>
          <msg:name>MessageIdentification</msg:name>
          <msg:value>ACUMPL_CLE_HC.1</msg:value>
        </msg:Option>
      </msg:Request>
    </msg:RequestMessage>
  </soap:Body></soap:Envelope>
```

Este es el retorno que se obtendría al hacer una llamada al servicio:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  </soap:Header>
  <soap:Body>
```




<p>4. El concentrador de medidas remoto invoca el servicio Get Message indicando los parámetros con los que desea seleccionar un único fichero publicado en el servidor.</p> <p>8. El concentrador de medidas recibe el mensaje de respuesta y cierra la conexión https con el servidor.</p>	<p>5. El servidor consulta que el fichero existe y puede ser consultado por ese concentrador.</p> <p>6. El recupera el fichero (comprimido), lo incluye en el mensaje de respuesta como base64 y firma el mensaje de respuesta.</p> <p>7. El servidor envía el mensaje de respuesta al concentrador de medidas.</p>
<p>Secuencia alternativa de eventos: No se puede recuperar el fichero.</p>	
<p>7. El concentrador de medidas recibe el mensaje Fault y cierra la conexión https con el servidor.</p>	<p>6. El servidor envía un mensaje de respuesta tipo Fault.</p>



ANEXO 1: FORMATO DE LA INFORMACIÓN

La información que se transfiera entre Concentradores de Medida se enviará en forma de ficheros con el formato definido en el documento [9].

Estos ficheros se enviarán comprimidos con el fin de minimizar los tiempos de transmisión. Para comprimirlos se utilizará un compresor **bzip2**.



ANEXO 2: TRATAMIENTO DE FICHEROS

Introducción

El tamaño máximo de los ficheros que se pueden transferir en cada invocación del servicio *getMessage* es de 50 MBytes (tamaño del fichero ya comprimido).

Para la transferencia de ficheros cuyo tamaño una vez comprimidos superen este límite, se dividirán en bloques. Los bloques se tratarán como ficheros normales recogándose con el servicio *getMessage*. Una vez que un concentrador tiene todos los bloques que forman un determinado fichero reconstruirá el fichero original, para poder procesarlo normalmente.

División de Ficheros

Los ficheros que deban fraccionarse por ser su tamaño excesivo, se dividirán en bloques con las siguientes características:

- El tamaño máximo de un bloque será de 50 MBytes.
- Los ficheros se dividirán una vez comprimidos.

Los bloques de un fichero se nombrarán añadiendo al nombre original del fichero la siguiente extensión:

<nombre del fichero>.<bloque>_<num bloques>

Donde:

- <nombre del fichero>: Es el nombre del fichero original.
- <bloque>: Es el número de orden de ese bloque.
- <num bloques>: Es el número total de bloques en que se divide el fichero original.

Por ejemplo, un fichero cuyo nombre sea P1_20031120.1, cuyo tamaño una vez comprimido sea de 110 MBytes, se dividiría en los siguientes tres bloques de 50 MBytes:

- P1_20031120.1.1_3: Primer bloque de 50 MBytes.
- P1_20031120.1.2_3: Segundo bloque de 50 MBytes.
- P1_20031120.1.3_3: Tercer y último bloque de 10 MBytes.

Cada bloque será tratado como un fichero independiente por los servicios *List Messages* y *get Message*.



ANEXO 3: DESCRIPCIÓN WDSL DE LOS SERVICIOS WEB

```
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:cmmsg="urn:iec62325.504:messages:1:0" xmlns:wss="urn:iec62325.504:wss:1:0"
xmlns:msg="http://iec.ch/TC57/2011/schema/message" targetNamespace="urn:iec62325.504:wss:1:0">
  <types>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:import namespace="urn:iec62325.504:messages:1:0"
schemaLocation="/iec62325.504.messages.1.0.xsd"/>
      <xs:import namespace="http://iec.ch/TC57/2011/schema/message"
schemaLocation="/ch.iec.tc57.2011.schema.message.xsd"/>
    </xs:schema>
  </types>
  <message name="msgRequestMessage">
    <part name="parameter" element="msg:RequestMessage"/>
  </message>
  <message name="msgResponseMessage">
    <part name="parameter" element="msg:ResponseMessage"/>
  </message>
  <message name="msgFaultMsg">
    <part name="msgFaultMessage" element="msg:FaultMessage"/>
  </message>
  <portType name="port_TFEDI_type">
    <operation name="request">
      <input message="wss:msgRequestMessage"/>
      <output message="wss:msgResponseMessage"/>
      <fault name="msgFaultMessage" message="wss:msgFaultMsg"/>
    </operation>
  </portType>
  <binding name="binding_TFEDI" type="wss:port_TFEDI_type">
    <soap12:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="request">
      <soap12:operation soapActionRequired="false" style="document"/>
      <input>
        <soap12:body use="literal"/>
      </input>
      <output>
        <soap12:body use="literal"/>
      </output>
      <fault name="msgFaultMessage">
        <soap12:fault name="msgFaultMessage" use="literal"/>
      </fault>
    </operation>
  </binding>
  <service name="ServiceEME">
    <port name="Service_EME_Port" binding="wss:binding_TFEDI">
      <soap12:address
location="https://ws.simel.ree.es/WebService_EME/Service_EME"/>
    </port>
  </service>
</definitions>
```



ANEXO 4: GUÍA DE BUENAS PRÁCTICAS

- Cada Concentrador de Medidas podrá realizar un filtrado a nivel de IP del acceso a su servidor. El acceso se permitirá sólo a las IPs de aquellos concentradores clientes que requieran recuperar información de dicho servidor y así lo comuniquen al responsable del mismo.
- Un concentrador pondrá sus ficheros a disposición de sus clientes con una antelación mínima necesaria de una hora antes del plazo límite establecido para que estos últimos puedan recoger y procesar la información.
- Cada Concentrador de Medidas publicará información en ficheros durante un periodo de tiempo no inferior a 7 días de manera general. Las excepciones para determinados tipos de ficheros, en caso de haberlas, se indicarán en [9]. Transcurrido el tiempo mínimo de permanencia desde la publicación del fichero el Concentrador podrá sacarlo del listado de ficheros disponibles para su recogida
- De forma general, un Concentrador de Medidas podrá limitar sus operaciones tal como se indica en el apartado 5.4.4 de [1] (parameterLimits). De forma concreta, se definen los siguientes parámetros de funcionamiento:
 - Un Concentrador de Medidas podrá configurar un número X máximo de días que procesará en una operación de listado (MaxApplicationTimeIntervallnDaysInListRequest). Este número máximo de días no podrá ser inferior a 3.
 - Un Concentrador de Medidas podrá configurar un número X máximo de mensajes que una operación de listado puede devolver (MaxNumMessagesInListResponse). Este número máximo de mensajes no podrá ser inferior a 2000.
 - Un Concentrador de Medidas podrá configurar un número máximo de conexiones por minuto por cada cliente que invoque sus servicios.
 - Un Concentrador de Medidas podrá configurar un número máximo de descargas de ficheros por minuto (MaxGetRequestPerMinute).



ANEXO 5: LISTA DE ERRORES DEVUELTOS POR EL SERVIDOR

- **Generales**

- **HAND-001:** *Unable to retrieve remote user from the https context [IP=?].*
Internal code used to send a forbidden error code.
- **HAND-002:** *Request message is not valid against schema. Details: ?.*
Internal code used to send a bad request error code.
- **HAND-003:** *User has no proper role for current message type.*
Internal code used to send an unauthorized error code.
- **HAND-004:** *Unable to read soap body [?]*
Internal code used to send a bad request error code.
- **HAND-005:** *Unsupported combination: [verb=?][noun=?]*
The verb and noun combination used in the request is not supported (do not correspond to any operation defined in [2]). Check the request and send the message.
- **HAND-006:** *Invalid message format*
Internal code used to send a bad request error code. The provided message format is invalid (i.e. a Binary format has been specified for a system that only admits XML)
- **HAND-007:** *Invalid signature*
Received message has invalid signature (invalid certificate, invalid reference, etc.). Check additional error message for details
- **HAND-008:** *Signature syntax error.*
The provided signature document is not valid against schema or there is an element with invalid value (invalid certificate chain, invalid certificate serial number, etc.) Check additional error message for details.
- **HAND-009:** *Unable to sign message.*
System cannot sign the provided message. Check if the message is a valid xml.
- **HAND-010:** *Server returns FAULT message.*
Server has returned a FAULT message instead of a Response Message. Check error details.
- **HAND-011:** *Response message has no payload or payload is empty.*
The response message has no payload or the given one is empty.
- **HAND-012:** *Response has invalid header elements: [verb=?][noun=?] expected: [verb=?][noun=?]*
The server response has invalid header elements that do not correspond with the expected ones. Thus, response is rejected.
- **HAND-013:** *Client do not trust server identity. Check client's trust store, check URL (do not use IP).*



The credentials of the certification authorization used by the server do not match any of the existent entries available in the trust store. Check that the URL used is the one you want. Add to your trust store the server's credentials.

- **HAND-014:** *The identity of the server certificate does not match the configured URL.*
Server has presented an identity "www.examplez.com" that do not match the configured (used) URL "www.example.com". Thus, client does not trust server and communication is aborted.
 - **HAND-015:** *The configured URL is unknown and cannot connected to it. Check URL, check DNS.*
The client cannot reach the configured URL. Check that there is no mistakes in the configured URL. Check your system's DNS configuration.
 - **HAND-016:** *The requested service does not exist on the server. Check configured URL. (HTTP=404, Not found).*
You have requested a web service URI that does not exists in the server. Check the URL.
 - **HAND-017:** *You do not have permission to access the web services on the server. (HTTP=403, Forbidden).*
Your system can reach the server, but your request has been rejected because your credentials (certificate). Check that you are using the correct certificate. Check that you are invoking the correct service.
 - **HAND-018:** *You are unauthorized to perform such request. (HTTP=401, Unauthorized).*
You are not authorized to send the request. Check your certificate.
 - **HAND-019:** *The server has received your request, but it has not sent a response in return (HTTP=200, Accepted).*
Your message was received by the server, but the server has returned an empty (or no [12]) response with status accepted. Ask server administrator for further details.
 - **HAND-020:** *The server has refused the connection. Check connectivity settings, check firewall settings.*
The server exists (see HAND-016) but has refuse your connection. There must be a connectivity problem that prevents the server accepts the connection. Check firewalls settings, check internet settings. It also be possible that the server is shut down.
 - **HAND-021:** *The server has rejected the request. (HTTP=400, Bad request)*
The server has rejected the request because it does not follow the rules. Check that you are not sending a binary message to a server that do not allows the use of binary files (or vice versa)
 - **HAND-022:** *Runtime exception. Check application logs.*
The server has rejected the request. Check application log for more information.
 - **HAND-023:** *Server has returned an unrecognized name. Disable SNI extension and retry.*
The server uses SNI (Server Name Indication) extension but returns an unrecognized name during SSL handshake. Try to disable the SNI extension in the client and retry the connection.
-
- **LIST**
 - **LST-001:** *Invalid parameters. Code must be a positive integer value.*



The provided code is a negative value. Code must be a positive integer number.

- **LST-002:** *Invalid operation parameters. Code must be an integer value.*
The provided code is not a number. Code must be a positive integer number.
- **LST-003:** *Invalid operation parameters. EndTime cannot precede StartTime.*
The provided time interval is incorrect, EndTime precede StartTime.
- **LST-004:** *Invalid operation parameters. Time interval cannot span more than ? days.*
Server has a limit for the number of days that the provided time interval can span. Reduce the number of days and repeat the request.
- **LST-005:** *Invalid operation parameters. You must provide either Code or StartTime and EndTime time interval values*
Check that you have provided a code or time interval filter but not both. If you have provided a time interval filter check that both values StartTime and EndTime were provided.
- **LST-006:** *Database read failed.*
Server is unable to read the database. Contact the system administrator.
- **LST-007:** *The operation returns more than ? messages. Please, use a smaller time interval value or add more filters.*
Server has a limit for the number messages that a response can have and the number of available messages that fulfill the provided filter is greater than such limit. Reduce the number of days that the provided time interval has or add additional filters in order to reduce the number of messages.
- **LST-008:** *Unable to create list response.*
Server is unable to create a list response because an unexpected condition. Contact the system administrator.
- **LST-009:** *Invalid operation parameters. IntervalType must be one of Application, Server.*
The IntervalType value provided is invalid. Check the possible values in the error message.
- **LST-010:** *Invalid operation parameters. ?*
The provided set of parameters is invalid (repeated parameters, invalid date formats, etc.) Check error message for details.
- **LST-011:** *Unknown parameter for list operation: ?*
The provided parameter is not valid (unspecified according to [12]) for list operation. Remove the parameter and send the request.
- **LST-012:** *Received message list has an invalid entry with no message code.*
The client has received an invalid list response where at least one of the entries has no message code. The client could also reject the whole response because it is not valid against schema (message code is mandatory)
- **LST-013:** *Received message list has an invalid list entry with no message identification. Message info: [code=?].*
The client has received an invalid list response where at least one of the entries has no message identification. The list entry's code is shown in the error. The client could also reject the whole response because it is not valid against schema (message's identification is mandatory)



- **LST-014:** *Received message list has an invalid list entry with no message type. Message info: [code=?][id=?].*
The client has received an invalid list response where at least one of the entries has no message type. The list entry's code and identification are shown in the error. The client could also reject the whole response because it is not valid against schema (message type is mandatory)
 - **LST-015:** *Received message list has an invalid list entry with no start application time interval. Message info: [code=?][id=?][type=?]*
The client has received an invalid list response where at least one of the entries has no application time interval. The list entry's code, identification and message type are shown in the error. The client could also reject the whole response because it is not valid against schema (application time interval is mandatory)
 - **LST-016:** *Received message list has an invalid list entry with no server timestamp. Message info: [code=?][id=?][type=?].*
The client has received an invalid list response where at least one of the entries has no server timestamp. The list entry's code, identification and message type are shown in the error. The client could also reject the whole response because it is not valid against schema (server timestamp is mandatory)
 - **LST-017:** *Received message list has an invalid list entry with no owner. Message info: [code=?][id=?][type=?].*
The client has received an invalid list response where at least one of the entries has no owner element. The list entry's code, identification and message type are shown in the error. The client could also reject the whole response because it is not valid against schema (message's owner is mandatory)
 - **LST-018:** *Received message list is invalid. ?*
The client has received an invalid list response so that it is unable to process the elements. Further information about the error nature is included. Client can use this error instead errors LST-012 to LST-017.
 - **LST-019:** *Unable to get message list from the received payload.*
The client has received an invalid response and is unable to create a proper message list (there is no payload or the payload contains a message that is not a list). Check received message. Contact server's administrator.
 - **LST-020:** *User has exceeded list operation limits. User is temporarily blocked.*
User has exceeded the allowed number of list operation requests per minute (`MaxListRequestPerMinute`) further list request will be rejected. Note that the implementation can also use the standard HTTP error code 429 (See [13])
-
- **GET**
 - **GET-001:** *Invalid parameters. Code must be a positive integer value.*
The provided code is a negative value. Code must be a positive integer number.
 - **GET-002:** *Invalid operation parameters. Code must be an integer value.*
The provided code is not a number. Code must be a positive integer number.



- **GET-003:** *Invalid invocation parameters. You must provide either Code or MessageIdentification and MessageVersion values.*
Your request contains both code and identification values. Use only one kind of filter.
- **GET-004:** *Invalid invocation parameters. You must provide Code or MessageIdentification and MessageVersion values.*
Your request has no filter (code or identification).
- **GET-005:** *QUEUE filter is not supported.*
The server does not implement the queue get operation.
- **GET-006:** *The requested message doesn't exist.*
There is no message that fulfils the provided set of filters or, if it exists, the user is not entitled to retrieve such message.
- **GET-007:** *Database read failed.*
The system is unable to read from its database because an unexpected condition. Contact system administrator.
- **GET-008:** *Unable to create get response.*
Server is unable to create a get response because an unexpected condition. Contact the system administrator.
- **GET-009:** *Unable to filter the message payload.*
Server is unable to provide a proper message according to confidentiality rules. Contact the system administrator.
- **GET-010:** *User has exceeded get operation limits. User is temporarily blocked.*
User has exceeded the allowed number of get operation requests per minute (`MaxGetRequestPerMinute`) further get request will be rejected.
- **GET-011:** *Invalid operation parameter. ?*
The provided set of parameters is invalid (repeated parameters, etc.) Check error message for details.
- **GET-012:** *Unknown parameter for get operation: ?*
The provided parameter is not valid (unspecified according to [12]) for get operation. Remove the parameter and send the request.
- **GET-013:** *File read failed*
The system is unable to read from its file system because an unexpected condition. Contact system administrator.
- **GET-014:** *The received message format is not supported [?]. Valid formats are [?]*
Server has returned a message which message format is not supported by [12]. Contact server's system administrator.
- **GET-015:** *Unable to unzip received binary XML.*
Server has returned a compressed xml payload but cannot be decompressed. Contact server's system administrator.
- **GET-016:** *Unable to read payload element.*
Client is unable to read the payload element from the received response. Check response is valid and has a payload element.



- **GET-017:** *Queue value must be "?" not "?"*
Given value for queue is not supported by [12].
- **GET-018:** *Server returns a binary message but did not provide file name.*
Server has return a binary message but it did not provide a file name which is mandatory according to [12].
- **GET-019:** *MessageVersion must be a positive integer.*
The provided MessageVersion value is not a positive integer.
- **GET-020:** *Server requires MessageVersion when MessageIdentification is provided.*
In the context of the business, the server requires a MessageVersion when MessageIdentification is provided.
- **GET-021:** *The received message's size ? is greater that the maximun allowed ?*
There is an agreement regarding the maximum message's size (see MaxPayloadSizeInMBInGetResponse) but the server has send a message which its size is greater than the one agreed.

- **QUERY**
 - **QRY-001:** *Invalid parameters. DataType value must be provided.*
The provided set of parameters does not contain the mandatory parameter DataType .
 - **QRY-002:** *Invalid parameters. Provided DataType value is not recognized.*
The system has no query with the provided identification.
 - **QRY-003:** *Invalid parameters. EndTime cannot precede StartTime.*
The provided time interval is incorrect, EndTime precede StartTime.
 - **QRY-004:** *Unable to create QueryData response.*
Server is unable to create a query data response because an unexpected condition. Contact the system administrator.
 - **QRY-005:** *Invalid parameters. Provided value ? for parameter ? is not recognized.*
For the given parameter the provided value is not recognized (is not valid).
 - **QRY-006:** *Database read failed.*
Server is unable to read the database. Contact the system administrator.
 - **QRY-007:** *Invalid parameters. Provided value ? for parameter ? must be a positive number.*
For the given parameter the provided value must be a positive number.
 - **QRY-008:** *The given parameters (? and ?) are mutually exclusive.*
The given parameters are mutually exclusive. Repeat the request using either one of them.
 - **QRY-009:** *Provided date parameter ? has invalid format: ?.*
The provided date value for the given parameter has an invalid format.
 - **QRY-010:** *Invalid operation parameters. ?*
The provided set of parameters is invalid (repeated parameters, invalid date formats, etc.)



Check error message for details.

- **QRY-011:** *Unknown parameter for query DataType ? : ?*
The provided parameter is not valid for the given data type. Remove the parameter and send the request.
- **QRY-012:** *Cannot process response: ?*
Client is unable to process the received payload. Check error detail.
- **QRY-013:** *User has exceeded Query operation limits. User is temporarily blocked.*
User has exceeded the allowed number of query operation requests per minute (`MaxQueryRequestPerMinute`) further query requests will be rejected. Note that the implementation can also use the standard HTTP error code 429 (See [13])