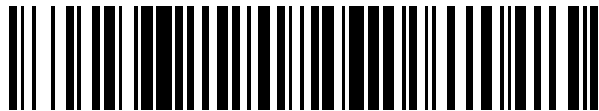


19



OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 728 139**

21 Número de solicitud: 201990028

51 Int. Cl.:

G06F 13/42 (2006.01)

G06F 1/32 (2009.01)

12

SOLICITUD DE PATENTE

A2

22 Fecha de presentación:

27.09.2017

30 Prioridad:

30.09.2016 US 62/402,098

43 Fecha de publicación de la solicitud:

22.10.2019

71 Solicitantes:

**ACLARA TECHNOLOGIES LLC (100.0%)
77 West Port Plaza, Suite 500
63146 St. Louis US**

72 Inventor/es:

**HAYNES, David y
DIERKING, Timothy**

74 Agente/Representante:

ÁLVAREZ LÓPEZ, Sonia

54 Título: **Mejoras a los patrones IEC 61968-9:2013 para las comunicaciones de un sistema de servicios públicos**

57 Resumen:

Mejoras en los patrones de lectura de contadores IEC 61968-9:2013. Estos incluyen la compresión significativa de los mensajes en un solo número entero que, cuando se transmite a través del sistema de comunicación de una empresa de servicios públicos, se utiliza para configurar contadores en las ubicaciones de los usuarios de servicios públicos, obtener datos de contadores, comandos y operaciones de control en una ubicación de servicios públicos, y ampliar las capacidades de mensajería. Se describe un procedimiento para simplificar el patrón IEC 61968-9:2013 para las comunicaciones entre una instalación de comunicaciones de servicios públicos HeadEnd y el contador en un EndPoint para permitir el uso del patrón para operaciones que no pudo funcionar anteriormente. Mientras proporciona esta capacidad mejorada, la capacidad del patrón IEC 61968-9:2013 para funcionar para su propósito previsto; es decir, la lectura y el control del contador, no está comprometida o impedida.

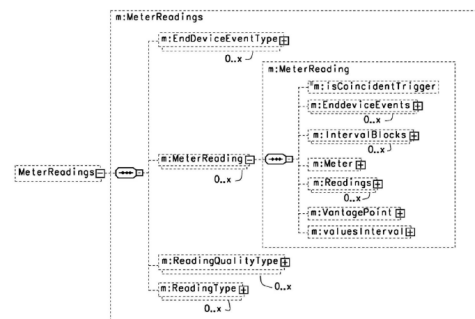


FIG. 1

DESCRIPCIÓN

Mejoras a los patrones IEC 61968-9:2013 para las comunicaciones de un sistema de servicios públicos

5 REFERENCIA A SOLICITUDES RELACIONADAS

Esta solicitud se basa en, y reclama el beneficio de, la solicitud de patente provisional 62/402,098 de los Estados Unidos presentada el 30 de septiembre de 2016 y que se incorpora aquí como referencia.

10 ANTECEDENTES DE LA INVENCION

Esta invención se refiere a comunicaciones para servicios eléctricos y otros servicios públicos; y, más particularmente, a un procedimiento y dispositivo para comunicaciones potenciadas y mejoradas entre una empresa de servicios públicos y sus clientes.

15

Un servicio público suministra un producto particular (electricidad, gas, agua) a través de un sistema de distribución a numerosos usuarios finales. Por lo general, la empresa de servicios públicos instala un contador en cada ubicación del cliente para medir la cantidad del producto consumido por ese cliente en esa instalación o sitio. Con el tiempo, estos contadores se han convertido en lo que ahora se conoce generalmente como "contadores inteligentes" en el sentido de que ahora no solo pueden medir el uso del producto, sino que también reciben instrucciones operativas del servicio público para, por ejemplo, controlar o configurar una carga en el sistema de distribución del servicio público en función del clima actual u otras condiciones, enviar informes o actualizaciones sobre las condiciones en la ubicación de un cliente particular al servicio público, y realizar otras funciones.

20

25 Las comunicaciones entre una empresa de servicios públicos y sus clientes se realizan utilizando ciertas normas como, por ejemplo, la norma 61968-9:2013 de la Comisión Electrotécnica Internacional (IEC). IEC 61968 comprende una serie de normas para el intercambio de información en un sistema de distribución eléctrica, las normas tienen distintos patrones para distintos propósitos comerciales. La norma IEC 61968-9 (en lo sucesivo, a veces denominada "Parte-9") se refiere a los contadores de electricidad, lectura y control de contadores; es decir, las comunicaciones entre la instalación de cabecera de un servicio público (HeadEnd o HE) que recibe, procesa y transmite señales a través de la red de comunicaciones del servicio público a un módulo de comunicaciones instalado en el contador en la ubicación del cliente; es decir, un punto final (EndPoint o EP).

30

Una edición reciente de la norma de la Parte-9 enumera 28 patrones, muchos de los cuales se utilizan para la configuración y el control de los componentes del sistema de comunicación. Se ha encontrado que la compatibilidad con estos patrones puede ser difícil y costosa para un dispositivo con limitaciones de cálculo; y que aunque algunos de los patrones son bastante grandes, no son necesariamente adecuados para satisfacer la necesidad de ciertas aplicaciones de una empresa de servicios públicos. Además, el patrón actual es algo engorroso. Como ilustración, una solicitud de "contador de lectura" en el formato IEC se formatea como una "cadena" de números enteros (enteros) y puntos ("."). Se requiere una técnica complicada (no descrita) para construir un identificador que defina o describa la unidad de medida para la medición de un contador eléctrico.

35

40

Por ejemplo, para solicitar una lectura de marcación típica de la superficie de un contador de electricidad residencial, el identificador utilizado en una Interfaz de programación de aplicaciones (API) es una cadena como

45

Las reglas de construcción para esta cadena en particular especifican 18 campos compuestos de caracteres; es decir, los dígitos y los puntos.

50 La presente invención está dirigida a extensiones a identificadores de tipo de lectura IEC 61968-9:2013 para admitir la configurabilidad de EndPoint; es decir, la configuración del dispositivo (contador de servicio público) al que se envían las instrucciones, los datos y las solicitudes de información.

RESUMEN DE LA INVENCION

55

La presente descripción está dirigida a las mejoras en los patrones de lectura de contadores 61968- 9:2013 de la IEC para extender los identificadores de tipo de lectura para admitir la configuración de un dispositivo EndPoint (EP) (es decir, un contador de servicio público) para el cual se envían las instrucciones, los datos y las solicitudes de información. Estas mejoras incluyen la compresión significativa del número de bytes utilizados para redactar los mensajes enviados desde una ubicación HeadEnd (HE) al EP para obtener datos del contador y otra información,

60

configurar un contador y expandir las capacidades de mensajería del patrón más allá de las capacidades actuales del patrón.

5 En una aplicación, ahora se proporciona una capacidad que permite la reconfiguración de dispositivos con capacidades computacionales limitadas, sin requerir el uso de patrones especializados adicionales, y que elimina la necesidad de los patrones utilizados anteriormente.

10 El uso del procedimiento cumple los objetivos de primero, serializar específicamente un patrón de la Parte-9 para comunicar datos y acontecimientos del contador; y, segundo, crear una API basada en la norma de la Parte-9 para que, además de comunicar las lecturas del contador, también se puedan transmitir los parámetros de control para configurar el contador de manera que rija la operación de los equipos ubicados de forma remota.

15 El procedimiento no solo simplifica el patrón para las comunicaciones entre un HeadEnd y un EndPoint (contador), sino que también permite el uso del patrón para operaciones que hasta ahora no se pueden realizar utilizando el patrón. Al lograr esta capacidad mejorada, la capacidad de la norma para funcionar para el propósito previsto; es decir, la lectura y el control del contador, no está comprometida o impedida.

20 Además, el procedimiento permite una formulación distinta para una cadena de mensaje transmitida desde un HeadEnd a un EndPoint para simplificar y facilitar las comunicaciones entre ellos y al mismo tiempo expandir las capacidades de un contador. El procedimiento permite definir parámetros personalizados, que luego se escriben y leen como los tipos de lectura estándar. Además, el flujo de trabajo requerido para la formulación de una cadena se puede reutilizar para simplificar la programación requerida para admitir la configurabilidad de un contador y su módulo de comunicación.

25 De acuerdo con la invención, el procedimiento utiliza el patrón "MeterReadings" de la Parte-9 que se utiliza para las lecturas y alarmas de los contadores. Ahora, el patrón "MeterReadings " se ha mejorado de manera que, además de sus funciones actuales, se extiende para transmitir parámetros de configuración entre componentes. Debido a esta capacidad mejorada, el patrón de la Parte-9 que de otra manera se utiliza para la configuración de componentes no es necesariamente necesario. Además, de acuerdo con la invención, la formulación de un mensaje enviado entre un
30 HeadEnd y un EndPoint da como resultado que se transmita un solo número entero en lugar de una cadena de números enteros y puntos. Esto representa una reducción significativa en los requisitos para dispositivos con limitaciones de cálculo y el costo de su uso.

35 Las mejoras de la invención por las que se necesitan transmitir menos bytes tienen las ventajas de permitir que más dispositivos se comuniquen a través de un canal de comunicaciones restringido, así como permitir que los dispositivos de comunicación operados por batería permanezcan operacionales durante largos períodos de tiempo antes de que la batería del dispositivo se agote.

40 Otros objetos y características serán en parte evidentes y en parte se señalarán a continuación.

DESCRIPCIÓN DE LOS DIBUJOS

45 Las figuras adjuntas, junto con la descripción detallada que sigue, forman parte de la especificación e ilustran las diversas realizaciones descritas en la especificación.

La Fig. 1 ilustra un patrón 61968-9 de la IEC para lecturas de contadores así como acontecimientos de dispositivos finales;

50 La Fig. 2 es una continuación del patrón de la Fig. 1 para acontecimientos de dispositivos finales;

La Fig. 3 es una continuación adicional del patrón de la Fig. 1 para las lecturas del contador; y,

La Fig. 4 ilustra una parte de un patrón de la Parte-9 para configuraciones de módulos de comunicación.

55 Los caracteres de referencia correspondientes indican partes correspondientes en las distintas vistas de los dibujos.

DESCRIPCIÓN DE UNA REALIZACIÓN PREFERIDA

60 La siguiente descripción detallada ilustra la invención a modo de ejemplo y no a modo de limitación. Esta descripción permite claramente a un experto en la materia realizar y utilizar la invención, y describe varias realizaciones,

adaptaciones, variaciones, alternativas y usos de la invención, incluido el que actualmente se cree que es el mejor modo de llevar a cabo la invención. Adicionalmente, debe entenderse que la invención no está limitada en su aplicación a los detalles de construcción y la disposición de los componentes establecidos en la siguiente descripción o ilustrados en los dibujos.

5

La invención es capaz de otras realizaciones y de ser practicada o llevada a cabo de varias maneras. Además, se entenderá que la fraseología y la terminología utilizadas en el presente documento tienen una finalidad descriptiva y no deben considerarse limitativas.

10 Para los fines de esta descripción, se entenderá que HeadEnd significa un conjunto de software que forma parte de un sistema de información de medición avanzada (AMI) que publica datos de equipos de campo (por ejemplo, contadores inteligentes). Se entenderá que EndPoint significa equipo de campo, particularmente el módulo de comunicaciones en un contador inteligente, transpondedor, etc.

15 En referencia a las Fig. 1-3, se representa un formato para formular una cadena utilizando un patrón de lectura de contador IEC 61968-9:2013 convencional para obtener una lectura de un contador eléctrico. En la Fig. 1, se muestran los encabezados principales o identificadores para la transmisión. En la Fig. 2, se muestran subtítulos o identificadores para acontecimientos de dispositivos finales. En la Fig. 3, se muestran subtítulos o identificadores para una lectura.

20

De acuerdo con la presente invención, las lecturas del contador y los parámetros personalizados se enumeran en una lista de HEEP (HeadEndEndPoint), y luego se enumera una enumeración particular como un número entero. El mensaje en la enumeración de tipo de lectura HEEP correspondiente a una lectura de kilovatios-hora netos se transmite como el número entero "20" en lugar de, por ejemplo, la cadena "0.0.0.1.4.1.12.0.0.0.0.0.0.0.3.72.0" .

25

La cadena de caracteres anterior se genera utilizando un lenguaje de marcado extensible (XML) que es bien conocido en la técnica.

El siguiente ejemplo ilustra la conversión de un modelo XML a una serialización HEEP de acuerdo con la invención.

30

Cargar datos de perfil con alarmas.

Los datos de intervalo utilizan un MeterReadings.xsd en XML y una estructura de bits de "Lecturas de contador compactas" en el HEEP.

35

La formulación en XML es la siguiente:

```

40 <EventMessage xmlns="htn:}/7ec.ff}/7C57/2011./sche:aa/n:esss^e">
    <Header>
        <Verb>created</Verb>
        <Noun>MeterReadings</Noun>
    </Header>
    <Payload>
45 <MeterReadings xmlns:m="http://iec.ch/TC57/2011/MeterReadings#"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://iec.ch/TC57/2011/MeterReadings#
    MeterReadings.xsd">
        <m:MeterReading>
            <m:EndDeviceEvents>
50 <m:createdDateTime>2013-07-
16T09:28:32Z</m:createdDateTime>
                <m:EndDeviceEventType ref="3.2.22.150"/> <!--
                electricMeter.battery.charge.minLimitReached--> </m:EndDeviceEvents>
            <m:EndDeviceEvents>
55 <m:createdDateTime>2013-07-
16T09:30:47Z</m:createdDateTime>
                <m:EndDeviceEventType ref="3.18.83.79"/> <!--
                electricMeter.memory.program.error--> </m:EndDeviceEvents>
            <m:EndDeviceEvents>
60 <m:createdDateTime>2001-12-

```



```

        <m:name>12345</m:name>
      </m:Names>
    </m:Meter>
  <m:valuesInterval>
5 <m:end>2013-07-16T10:30:00Z</m:end>
    </m:valuesInterval>
  </m:MeterReading>
</m:MeterReadings>
</Payload>
10 </EventMessage>

```

El tamaño resultante del mensaje XML tiene aproximadamente 2920 bytes de longitud.

La versión HEEP se presenta de acuerdo con el siguiente cuadro:

15

Elemento	Binario	Decimal	Hex	Tamaño en bytes
Interface Revision Number		0	00	1
TransactionType=Bubblellp	10	131	83	1
resource=/bu/lp	00 0011			
method =POST		2	02	1
RequestID		2	02	1
valuesInterval.end= 2013-07-16T10:30:00Z		1373970600	51E520A8	4
AlarmQty=3	0011	54	36	1
ReadingQty=6	0110			
EndDeviceEvents.createdDateTime =2013-07-16T09:28:32Z		1373966912	51E51240	4
EndDeviceEvents. EndDeviceEventType= Low Battery		23	0017	2
NameValuePairQty	0000	0	00	1
EndDevEDValueSize	0000			
EndDeviceEvents.createdDateTime =2013-07-16T09:30:47Z		1373967047	51E512C7	4
EndDeviceEvents. EndDeviceEventType= Meter Flash Memory Error		17	0011	2
NameValuePairQty	0000	0	00	1
EndDevEDValueSize	0000			
EndDeviceEvents.createdDateTime =2001-12-17T09:30:47Z		1008581447	3C1DBB47	4
EndDeviceEvents. EndDeviceEventType= reverse rotation detected		11	000B	2
NameValuePairQty	0001	17	11	1

ES 2 728 139 A2

EndDevEDValueSize	0001			
EndDeviceEventDetails.name reverse rotation count.3.12.93.219		1033	0409	2
EndDeviceEventDetails.value		3	03	1
timeStampPresent=FALSE	0	42	2A	1
ReadingQualitiesPresent=FALSE	0			
PendingPowerOfTen=-5	10 1			
ReadingValueSizeInBytes=3B	010			
ReadingType=5min deltaData fwd kWh		73	0049	2
Readings.value		2345678	23CACE	3
timeStampPresent=FALSE	0	40	28	1
ReadingQualitiesPresent=FALSE	0			
PendingPowerOfTen=-5	10 1			
ReadingValueSizeInBytes=1 B	000			
ReadingType=5 minute deltaData rev kWh		75	004B	2
Readings.value		0	00	1
timeStampPresent=FALSE	0	42 72	2A 0048	1 2
ReadingQualitiesPresent=FALSE	0			
PendingPowerOfTen=-5	10 1			
ReadingValueSizeInBytes=3B	010			
ReadingType=5 minute deltaData fwd kVARh				
Readings.value=1.23456		123456	01E240	3
timeStampPresent=FALSE	0	9	09	1
ReadingQualitiesPresent=FALSE	0			
PendingPowerOfTen=-1	00 1			
ReadingValueSizeInBytes=2B	001			
ReadingType=Volts RMS Ph A		34	0022	2
Readings.value=479.9		4799	12BF	2
timeStampPresent=FALSE	0	9	09	1
ReadingQualitiesPresent=FALSE	0			
PendingPowerOfTen=-1	00 1			
ReadingValueSizeInBytes=2B	001			
ReadingType= Volts rms Ph B		36	0024	2

Readings.value=480.0		4800	12C0	2
timeStampPresent=FALSE	0	8	08	1
ReadingQualitiesPresent=FALSE	0			
PendingPowerOfTen=-1	00 1			
ReadingValueSizeInBytes=1 B	000			
ReadingType= Volts Ph C		35	0023	2
Readings.value=10.0		100	64	1
Size				63

Con respecto a lo anterior, se observará que la identificación del contador particular (ID del contador) no está representada porque está representada en una capa distinta.

5 Es importante destacar que el uso de la estructura de bits HEEP ahora reduce el tamaño del mensaje transmitido de 2920 bytes en el formato XML a solo 63 bytes. Esto es una reducción de aproximadamente el 98 % en el número de bytes que deben transmitirse para obtener la misma información.

A modo de otro ejemplo, las alarmas en tiempo real, como las alarmas de "último suspiro" y "energía restaurada",
10 utilizan un "MeterReadings.xsd" en XML, y una estructura de bits de "Lecturas de contador completas" en el HEEP.

La formulación en XML es la siguiente:

```

<EventMessage xmlns="http://iec.ch/TC57/2011/schem
15 <Header>
  <Verb>created</Verb>
  <Noun>MeterReadings</Noun>
</Header>
<Payload>
20 <m:MeterReadings xmlns:m="http://iec.ch/TC57/2011/MeterReadings#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://iec.ch/TC57/2011/MeterReadings# MeterReadings.xsd">
  <m:MeterReading> <m:isCoincidentTrigger>>false</m:isCoincidentTrigger> <m:EndDeviceEvents>
    <m:createdDateTime>2015-01 -
25 13T11:15:08Z</m:createdDateTime>
  <m:EndDeviceEventType ref="26.26.0.216"/>
    <!--com Device, power.. restored-->
  </m:EndDeviceEvents>
  <m:Meter>
30 <m:Names>
  <m:name>12345</m:name>
  </m:Names>
  </m:Meter>
  <m:Readings>
35 <m:value>34</m:value>
  <m:ReadingType
ref="0.0.0.1.0.1.137.0.0.0.0.0.0.0.0.0.0.111,0"/>
  <!--bulkQuantity electricitySecondaryMetered
powerQuality (count)-->
40 </m:Readings>
  <m:Readings>
    <m:value>240.1 </m:value> <m:ReadingType
ref="0.0.0.6.0.1.54.0.0.0.0.0.0.0.128.0.29.0"/>
    <!--indicating electricitySecondaryMetered voltage-rms phaseA (V)-->

```


ES 2 728 139 A2

```

5      </m:Readings>
      <m:Readings>
        <m:value>240.1 </m:value> <m:ReadingType
ref="0.0.0.6.0.1.54.0.0.0.0.0.0.0.0.0.64.0.29.0"/>
10     <!--indicating electricitySecondaryMetered voltage-rms phaseB (V)->
      </m:Readings>
      <m:Readings>
        <m:value>240.2</m:value> <m:ReadingType
ref="0.0.0.6.0.1.54.0.0.0.0.0.0.0.0.0.32.0.29.0"/>
15     <!--indicating electricitySecondaryMetered voltage-rms phaseB (V)->
      </m:Readings>
      <m:valuesInterval>
        <m:end>2015-01 -13T11:15:10Z</m:end>
        <!--This is a common timestamp for all readings and events that otherwise do not have a
15     timestamp provided.- ->
      </m:valuesInterval>
    </m:MeterReading>
  </m:MeterReadings>
</Payload>
20 </EventMessage>

```

Esta versión XML requiere 1699 bytes.

La versión HEEP se presenta de acuerdo con el siguiente cuadro:

25

Elemento	Binario	Decimal	Hex	Tamaño en bytes
Interface Revision Number		0	00	1
TransactionType=Bubblellp	10	129	81	1
resource=/bu/am	00 0001			
method =POST		2	02	1
RequestID		26	1A	1
valuesInterval.end= 2015-01-13T11:15:10		1421147710	54B4FE3E	4
AlarmQty=1	0001	20	14	1
ReadingQty=4	0100			
EndDeviceEvents.createdDateTime =2015-01-13T11:15:08		1421147708	54B4FE3C	4
EndDeviceEvents. End DeviceEventT ype = comDevice. power., restored		120	0078	2
NameValuePairQty	0000	0	00	1
EDEventDetailValueSize	0000			
IsCoincidentTrigger=FALSE	0	0	00	1
timeStampPresent=FALSE	0			
ReadingQualitiesPresent=FALSE	0			

ES 2 728 139 A2

RSVD	00			
PendPowerOfTen=None	000			
ReadingsValueSizeInBytes	0000 0000 0001	23	0017	2
RSVD	0			
ReadingsValueType=uint	111			
ReadingType=powerQuality count		128	0080	2
Readings.value		34	22	1
IsCoincidentTrigger=FALSE	0	1	01	1
timeStampPresent=FALSE	0			
ReadingQualitiesPresent=FALSE	0			
RSVD	00			
PendPowerOfTen=deci	001			
ReadingsValueSizeInBytes	0000 0000 0010	39	0027	2
RSVD	0			
ReadingsValueType=uint	111			
ReadingType=voltage phase A		34	0022	2
Readings.value		2401	0961	2
IsCoincidentTrigger=FALSE	0	1	01	1
timeStampPresent=FALSE	0			
ReadingQualitiesPresent=FALSE	0			
RSVD	00			
PendPowerOfTen=deci	001			
ReadingsValueSizeInBytes	0000 0000 0010	39	0027	2
RSVD	0			
ReadingsValueType=uint	111			

ReadingType=voltage phase B		36	0024	2
Readings.value=240.1		2401	0961	2
IsCoincidentTrigger=FALSE	0	1	01	1
timeStampPresent=FALSE	0			
ReadingQualitiesPresent=FALSE	0			
RSVD	00			
PendPowerOfTen=deci	001			
ReadingsValueSizeInBytes	0000 0000 0010	39	0027	2
RSVD	0			
ReadingsValueTypeAndCast	111			
ReadingType=voltage phase C		35	0023	2
Readings.value=240.2		2402	0962	2
Size				43

De acuerdo con el procedimiento de la invención, la versión HEEP requiere solo 43 bytes, lo que es ~ 2½ % de lo requerido en la versión XML.

- 5 Los expertos en la técnica entenderán que los resultados de ambos ejemplos incluyen ahorros estadísticamente significativos en la operación del sistema para adquirir la misma información que se obtuvo previamente utilizando los protocolos XML y representan reducciones estadísticamente significativas en los requisitos para dispositivos de memoria restringidos o restringidos computacionalmente y el coste de usarlos.
- 10 A modo de ejemplo, ciertos canales de comunicación están restringidos de modo que solo se pueden transmitir cantidades limitadas de información sobre ellos. En consecuencia, la capacidad de transmitir menos bytes a través del canal para comunicar instrucciones o adquirir datos ahora permite que más dispositivos se comuniquen a través de dichos canales. Además, algunos dispositivos de comunicación funcionan con baterías y el tiempo que pueden comunicarse está limitado por la cantidad de bytes que deben transmitir para comunicar instrucciones o adquirir
- 15 datos antes de que sus baterías se agoten. Tener que transmitir menos bytes, como resultado de las mejoras de la presente invención, ahora permite que estos dispositivos permanezcan operativos durante más tiempo antes de que la batería de un dispositivo se agote.

Además de estas ventajas, una ventaja adicional de las mejoras de la presente invención es que proporcionan la capacidad de reconfigurar dispositivos como los contadores inteligentes, por ejemplo, sin requerir soporte para patrones adicionales especializados. Actualmente, el patrón de la Parte-9 define que MeterReadings.xsd tiene un tamaño de 21 kB. Sin embargo, de acuerdo con la invención, MeterReadings.xsd ahora también se puede utilizar para realizar cambios de configuración y esta reutilización permite que se usen otros patrones. Estos incluyen, por ejemplo, MeterConfig.xsd que se define con un tamaño de 42kB, y ComModuleConfig.xsd que se define con un

25 tamaño de 17kB. Además, otros patrones que se utilizaban anteriormente ahora se vuelven innecesarios. Todo esto ahorra un valioso espacio de memoria en dispositivos con capacidades computacionales limitadas. Un ejemplo de esta característica se describe a continuación.

Con referencia a la Fig. 4, se representa una parte de un patrón de la Parte-9 para las configuraciones de los módulos de comunicación. Además de la recopilación de datos y las instrucciones de control de los dos ejemplos anteriores, el procedimiento de la invención permite además cambios o ajustes en el módulo de comunicaciones de un contador inteligente para afectar las operaciones del sistema. Estos ajustes se pueden realizar "por ondas" y la norma de la Parte-9, cuya parte se muestra en la Fig. 4, emplea un patrón XML para este propósito. El siguiente es

30

un ejemplo de una configuración de un contador para un desplazamiento de zona horaria.

El patrón empleado es ComModuleConfig.xsd., Y el proceso XML es el siguiente:

```

5 <EventMessage xmlns="hzn:TC57/2011/ComModuleConfig#">
  <Header>
    <Verb>changed</Verb>
    <Noun>ComModuleConfig</Noun>
  </Header>
10 <Payload>
  <m:ComModuleConfig
    xmlns:m="http://iec.ch/TC57/2011/ComModuleConfig#" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://iec.ch/TC57/2011/ComModuleConfig# comModuleConfig.xsd">
    <m:ComModule>
15      <m:timeZoneOffset>-6.0</m:timeZoneOffset>
      <m:electronicAddress>
        <m:mac>12345</m:mac>
      </m:electronicAddress>
    </m:ComModule>
20 </m:ComModuleConfig>
  </Payload>
</EventMessage>

```

Según lo formulado, el mensaje requiere 546 bytes.

25 En la versión HEEP de acuerdo con el procedimiento de la invención, este parámetro de zona horaria se trata como cualquier otra lectura y lleva una escritura de parámetro dentro de una estructura de bits "ExchangeWithID" y la versión HEEP se representa de acuerdo con el siguiente cuadro:

Elemento	Binario	Decimal	Hex	Tamaño en Bytes
Interface Revision Number		0	00	1
TransactionType=Request	00	30	1E	1
resource=/or/pm	01			
	1110			
MethodStatus=PUT		3	03	1
RequestID		6	06	1
EDEvent	0000	2	02	1
ReadingQty	0001			
isCoincidentTrigger=FALSE	0	0	00	1
timeStampPresent=FALSE	0			
ReadingQualitiesPresent=FALSE	0			
RSVD	00			
PendingPowerOfTenAdjustment	000			

ReadingsValueSizeInBytes	0000 0000 0001	21	0015	2
RSVD	0			
ReadingsValueTypecast=INT	101			
ReadingType= timeZoneOffset		1224	04C8	2
Readings.value =-6.0 hr = -21,600 seconds		-21,600	FFABA0	3
			Size	12

Nuevamente de acuerdo con el procedimiento de la invención, la versión HEEP requiere solo 12 bytes, lo que es aproximadamente el 2 % de lo requerido en la versión XML.

- 5 En vista de lo anterior, se verá que se han alcanzado los diversos objetos y ventajas de la presente divulgación y se han obtenido otros resultados ventajosos.

REIVINDICACIONES

- 5 1. En un patrón IEC 61968-9 para obtener lecturas de un dispositivo EndPoint que incluye un contador del servicio público, así como ordenar y controlar los acontecimientos que ocurren en un dispositivo EndPoint, la mejora incluye reformatear un mensaje transmitido desde un HeadEnd a un EndPoint desde un formato de protocolo XML en un formato HeadEnd-EndPoint (HEEP) para reducir el número de bytes que conforman el mensaje que se transmite, para mejorar las comunicaciones entre el HeadEnd y el EndPoint y aumentar la cantidad de tipos de mensajes que pueden transmitirse entre el HeadEnd y el EndPoint.
10
2. La mejora de la reivindicación 1, en la que la transmisión de mensajes compuesta por el número reducido de bytes facilita la comunicación de mensajes a través de un canal de comunicaciones restringido a dispositivos adicionales que, de otro modo, podrían comunicarse con los que si los mensajes se generaran utilizando el formato de protocolo XML.
15
3. La mejora de la reivindicación 1, en la que la transmisión de mensajes generados con el número reducido de bytes permite que los dispositivos de comunicación operados por batería permanezcan operativos durante más tiempo antes de que la batería de un dispositivo se agote debido a que la reducción en el número de bytes de mensajes reduce la descarga de la batería de un dispositivo que funciona con batería al recibir y procesar los mensajes.
20
4. La mejora de la reivindicación 1 para proporcionar una capacidad de reconfigurar dispositivos que tienen capacidades computacionales limitadas, sin requerir el uso de patrones de mensajes especializados adicionales, y eliminando la necesidad de patrones utilizados si los mensajes se generaron en un formato de protocolo XML.
25
5. La mejora de la reivindicación 1, en la que los mensajes transmitidos entre HeadEnd y EndPoint solo comprenden un número entero único en lugar de una cadena de enteros y puntos, de acuerdo con que se requiere utilizar el formato de protocolo XML.
30
6. La mejora de la reivindicación 1, en la que el reformateo de mensajes permite definir parámetros personalizados para un contador, siendo dichos parámetros escritos y leídos como otros tipos de lectura estándar.
- 35 7. La mejora de la reivindicación 6, cuyos patrones de lectura de contadores se extienden para permitir la transmisión de parámetros de configuración entre dispositivos, incluyendo otros contadores de servicios públicos.
8. La mejora de la reivindicación 7, que incluye además la reconfiguración de dispositivos con capacidades computacionales limitadas sin requerir el uso de patrones especializados o patrones utilizados previamente para reconfigurar los dispositivos.
40
9. Un procedimiento para mejorar las comunicaciones en un sistema de comunicación de servicios públicos que comprende:
45 reformatear un mensaje generado en un patrón IEC 61968-9 de un formato de protocolo XML a un formato de HeadEnd-EndPoint (HEEP) en el que el mensaje se transmite desde una ubicación HeadEnd (HE) a una ubicación EndPoint (EP) en la que se encuentra un contador del servicio público, el mensaje generado de esta manera requiere menos cantidad de bytes para producir y transmitir en el formato HEEP que en el formato del protocolo XML;
50 transmitir el mensaje desde la ubicación HeadEnd a la ubicación EndPoint a través de la comunicación del servicio público en el formato HEEP para obtener lecturas del contador del servicio público en el EndPoint y para ordenar y controlar los acontecimientos que ocurren en el EndPoint utilizando el contador del servicio público en la ubicación EndPoint, lo que reduce el número de bytes que comprende el mensaje transmitido, lo que aumenta el número de tipos de mensajes que se pueden transmitir entre HeadEnd y EndPoint.
55
10. El procedimiento de la reivindicación 9, que facilita aún más la comunicación de mensajes a través de un canal de comunicaciones restringido para que los mensajes puedan transmitirse a dispositivos adicionales de los que podrían comunicarse de otra manera que si los mensajes se generaran utilizando el formato de protocolo XML.
60

11. El procedimiento de la reivindicación 9, en el que la transmisión de mensajes generados con el número reducido de bytes permite que los dispositivos de comunicación operados por batería permanezcan operativos durante más tiempo antes de que la batería de un dispositivo se agote debido a que la reducción en el número de bytes de mensajes reduce la descarga de la batería por el dispositivo que funciona con batería al recibir y procesar los mensajes.
12. El procedimiento de la reivindicación 9, que incluye además proporcionar una capacidad para reconfigurar dispositivos que tienen capacidades computacionales limitadas sin requerir el uso de patrones de mensajes especializados adicionales, y que elimina la necesidad de patrones utilizados si los mensajes se generaron en el formato de protocolo XML.
13. El procedimiento de la reivindicación 9, que incluye además la transmisión de los mensajes entre HeadEnd y EndPoint como un solo número entero en lugar de una cadena de enteros y puntos, como se requiere si los mensajes se generaron utilizando el formato de protocolo XML.
14. El procedimiento de la reivindicación 9, que incluye además el reformateo de mensajes en el formato HEEP que permite definir parámetros personalizados para un contador, dichos parámetros se escriben y leen como otros tipos de lectura estándar.
15. El procedimiento de la reivindicación 14, que incluye además extender los patrones de lectura de contadores para permitir la transmisión de parámetros de configuración entre dispositivos dentro del sistema de comunicación de servicios públicos incluyendo otros contadores de servicios públicos.

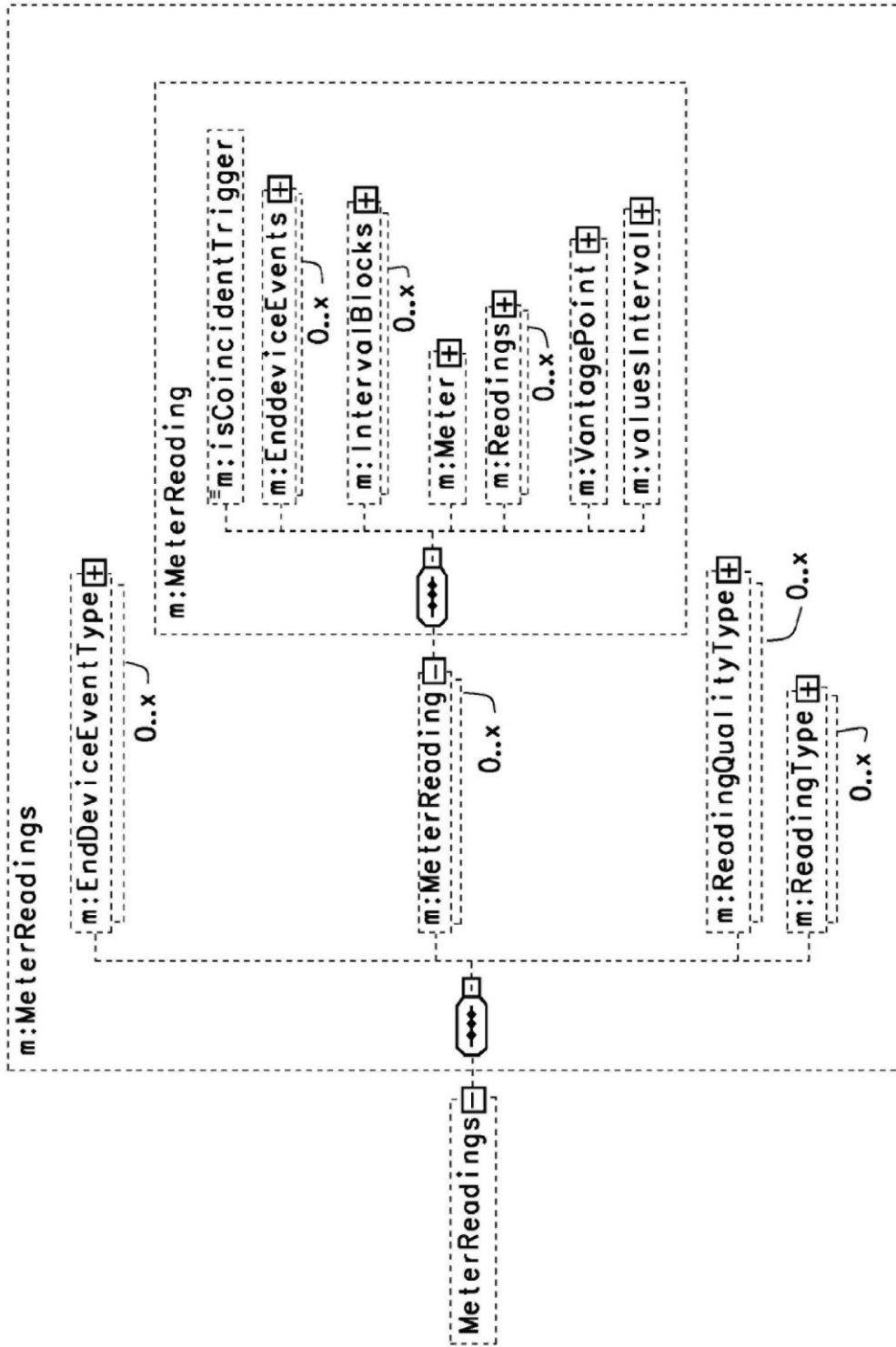


FIG. 1

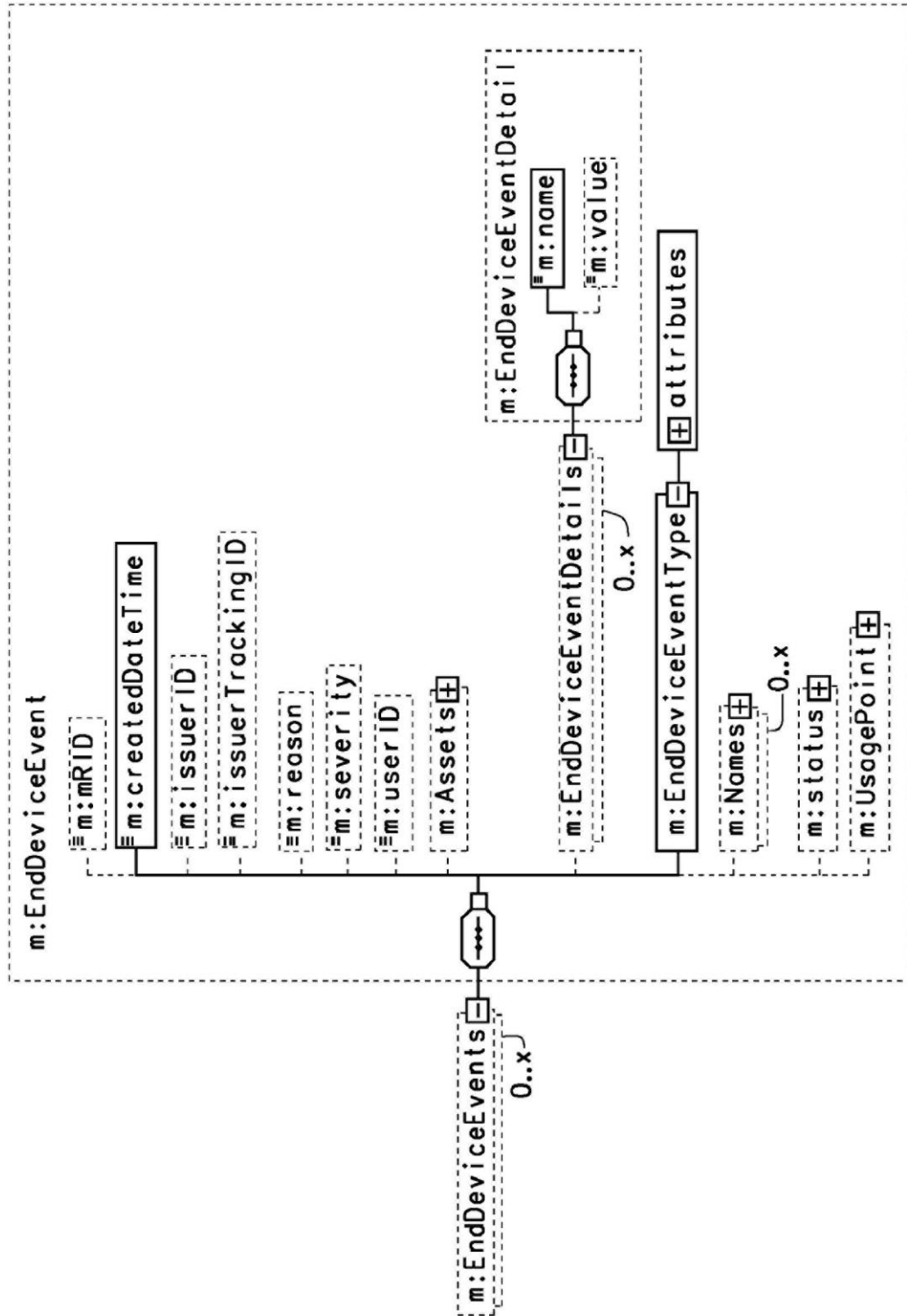


FIG. 2

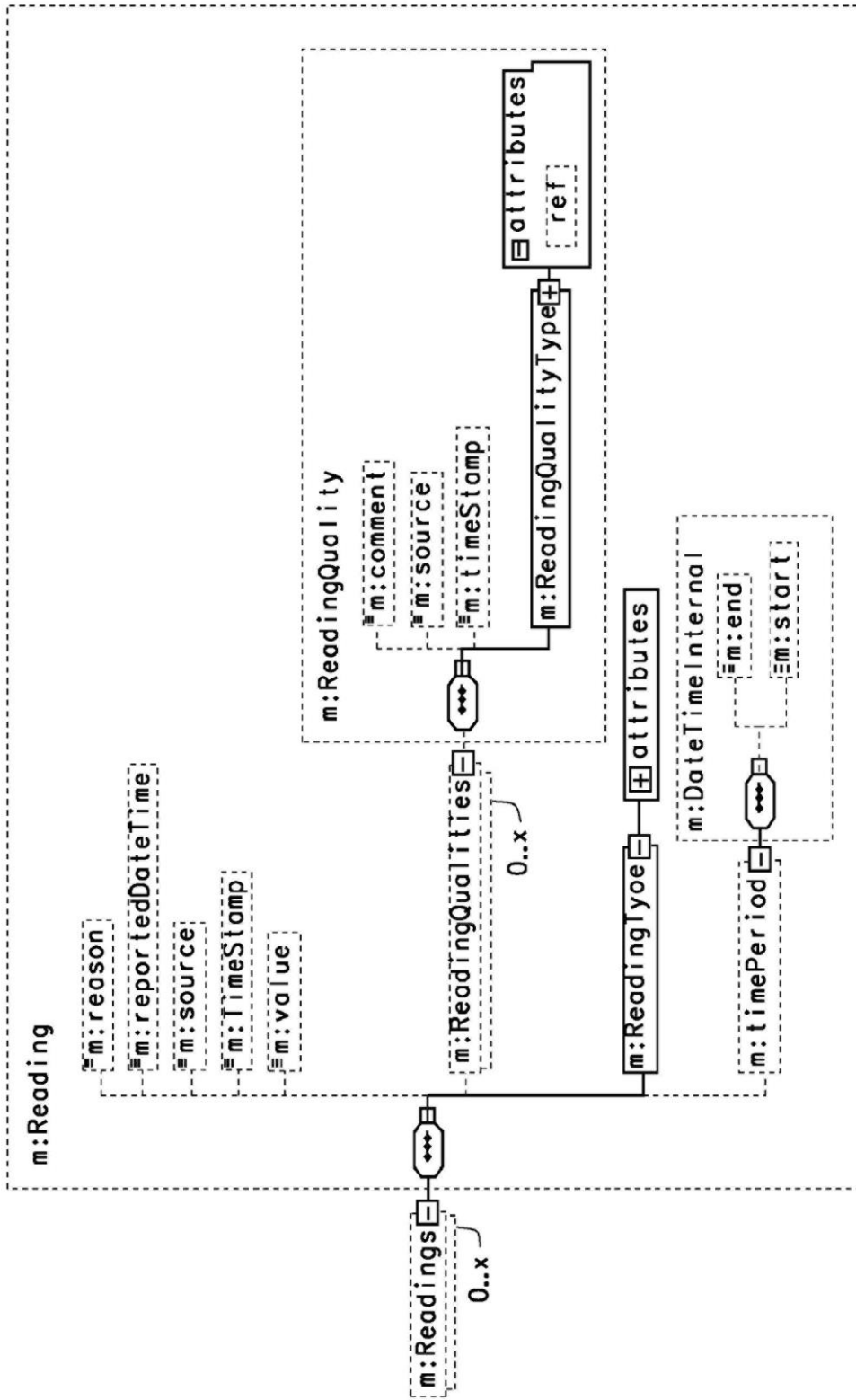


FIG. 3

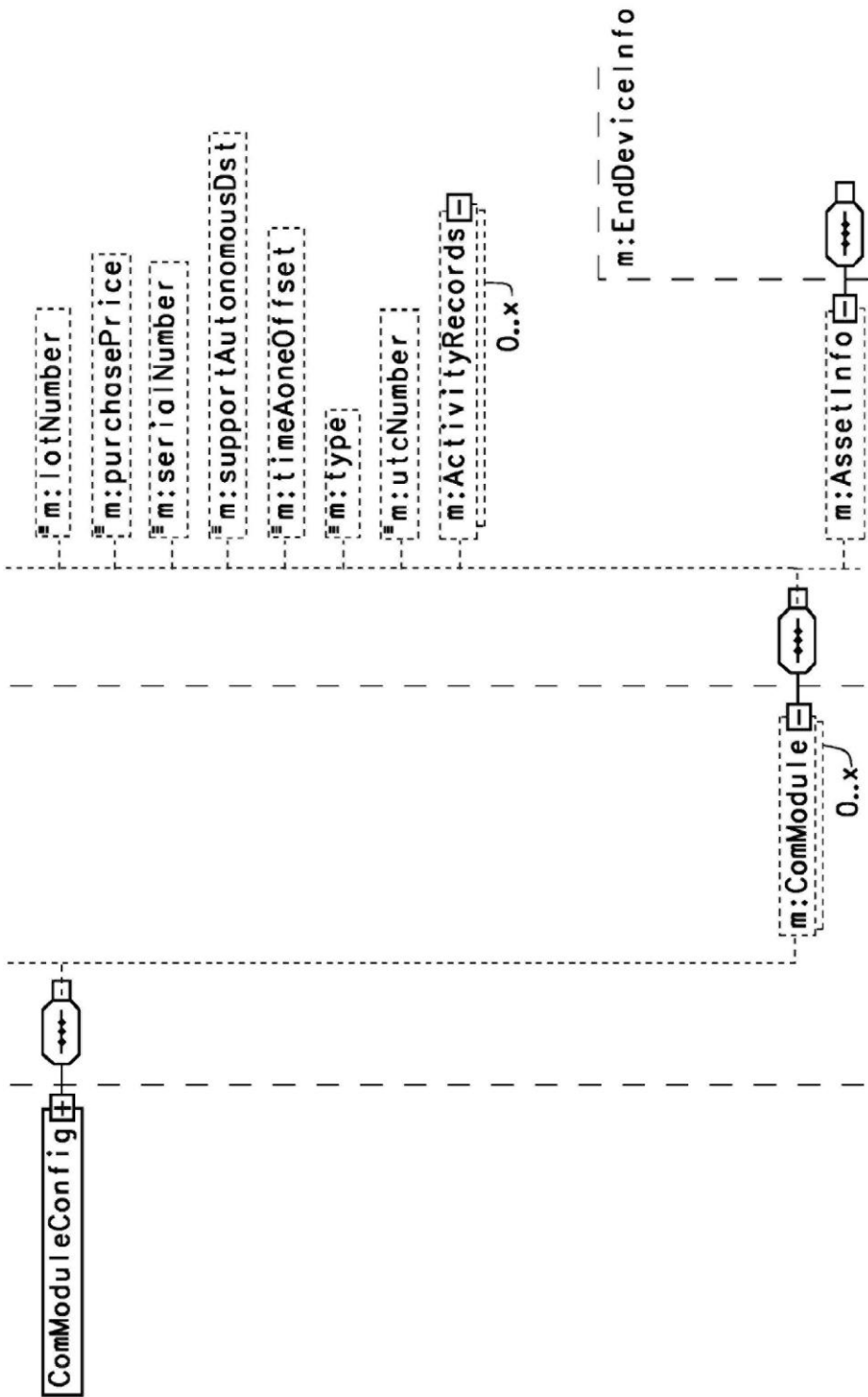


FIG. 4