IEC 61968-9

Edition 1.0 2009-09

INTERNATIONAL STANDARD

Application integration at electric utilities – System interfaces for distribution management –
Part 9: Interfaces for meter reading and control

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland
Email: inmail@iec.ch
Web: www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

- Catalogue of IEC publications: www.iec.ch/searchpub
The IEC on-line Catalogue enables you to search by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, withdrawn and replaced publications.
- IEC Just Published: www.iec.ch/online_news/justpub
Stay up to date on all new IEC publications. Just Published details twice a month all new publications released. Available on-line and also by email.
- Electropedia: www.electropedia.org
The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.
- Customer Service Centre: www.iec.ch/webstore/custserv
If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:

Email: csc@iec.ch
Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00

IEC®

IEC 61968-9

Edition 1.0 2009-09

INTERNATIONAL STANDARD

colour inside

Application integration at electric utilities – System interfaces for distribution management – Part 9: Interfaces for meter reading and control

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

PRICE CODE XH

ICS 33.200

ISBN 978-2-88910-592-2

Warning! Make sure that you obtained this publication from an authorized distributor.

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

APPLICATION INTEGRATION AT ELECTRIC UTILITIES – SYSTEM INTERFACES FOR DISTRIBUTION MANAGEMENT –

Part 9: Interfaces for meter reading and control

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 61968-9 has been prepared by IEC technical committee 57: Power systems management and associated information exchange.

The text of this standard is based on the following documents:

FDIS	Report on voting
57/1009/FDIS	57/1020/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts in IEC 61968 series, under the general title: *Application integration at electric utilities – System interfaces for distribution management*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the maintenance result date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

A bilingual version of this publication may be issued at a later date.

IMPORTANT – The “colour inside” logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this publication using a colour printer.

INTRODUCTION

The purpose of this document is to define a standard for the integration of metering systems (MS), which would include traditional (one or two-way) automated meter reading (AMR) systems, with other systems and business functions within the scope of IEC 61968. The scope of this standard is the exchange of information between a metering system and other systems within the utility enterprise. The specific details of communication protocols those systems employ are outside the scope of this standard. Instead, this standard will recognize and model the general capabilities that can be potentially provided by advanced and/or legacy meter infrastructures, including two-way communication capabilities such as load control, dynamic pricing, outage detection, distributed energy resource (DER) control signals and on-request read. In this way, this standard will not be impacted by the specification, development and/or deployment of next generation meter infrastructures, either through the use of standards or proprietary means.

The IEC 61968 series of standards is intended to facilitate inter-application integration as opposed to intra-application integration. Intra-application integration is aimed at programs in the same application system, usually communicating with each other using middleware that is embedded in their underlying runtime environment, and tends to be optimized for close, real-time, synchronous connections and interactive request/reply or conversation communication models. Therefore, these interface standards are relevant to loosely coupled applications with more heterogeneity in languages, operating systems, protocols and management tools. This series of standards is intended to support applications that need to exchange data every few seconds, minutes, or hours rather than waiting for a nightly batch run. This series of standards, which are intended to be implemented with middleware services that exchange messages among applications, will complement, not replace utility data warehouses, database gateways, and operational stores.

As used in IEC 61968, a distribution management system (DMS) consists of various distributed application components for the utility to manage electrical distribution networks. These capabilities include monitoring and control of equipment for power delivery, management processes to ensure system reliability, voltage management, demand-side management, outage management, work management, automated mapping and facilities management. Standard interfaces are defined for each class of applications identified in the interface reference model (IRM), which is described in IEC 61968-1: System interfaces for distribution management – Part 1: Interface architecture and general requirements.

.

This part of IEC 61968 contains the clauses listed below.

Document overview for IEC 61968-9

Clause/Annex	Title	Purpose
1	Scope	The scope and purpose of the document are described
2	Normative references	Documents that contain provisions which, through reference in this text, constitute provisions of this International Standard
3	Terms, definitions and abbreviations	Provides terms, definitions and abbreviations
4	Reference and information models	Description of general approach to metering system, reference model, use cases, interface reference model, meter reading and control functions and components, message type terms and static information model
5	Meter reading and control message types	Message types related to the exchange of information for documents related to meter reading and control
Annex A	Description of message type verbs	Description of the verbs that are used for the message types
Annex B	Recommended message structure	CIM extensions to support the recommended message structure for meter reading and control
Annex C	Recommended procedure for the generation of a ReadingType	Recommended technique for constructing, and offers sample enumerations for the ReadingTypeId textual name and mRID
Annex D	Recommended Quality code enumerations	Recommends technique for constructing, and offers sample enumerations for reading quality codes
Annex E	Recommended EndDeviceEvent category enumerations	Recommends EndDeviceEvent alarm codes
Annex F	Recommended EndDeviceControl code enumerations	Describes recommended technique for leveraging EndDeviceEvent codes to create EndDeviceControl codes
Annex G	Recommended procedure for maintaining relationships between objects	To describe the use of the master resource identifier (mRID)
Annex H	XML schemas for message payloads	To provide xsd information for use by developers to create IEC 61968-9 messages
Annex I	Mappings to ebIX	To provide mappings between IEC 61968-9 meter readings and ebIX.
Annex J	Request parameters	Describes the qualification of GET requests using request parameters

APPLICATION INTEGRATION AT ELECTRIC UTILITIES – SYSTEM INTERFACES FOR DISTRIBUTION MANAGEMENT –

Part 9: Interfaces for meter reading and control

1 Scope

This document is Part 9 of the IEC 61968 standard and specifies the information content of a set of message types that can be used to support many of the business functions related to meter reading and control. Typical uses of the message types include meter reading, meter control, meter events, customer data synchronization and customer switching. Although intended primarily for electrical distribution networks, IEC 61968-9 can be used for other metering applications, including non-electrical metered quantities necessary to support gas and water networks.

The purpose of this document is to define a standard for the integration of metering systems (MS), which includes traditional manual systems, and (one or two-way) automated meter reading (AMR) systems, with other systems and business functions within the scope of IEC 61968. The scope of this International Standard is the exchange of information between a metering system and other systems within the utility enterprise. The specific details of communication protocols those systems employ are outside the scope of this standard. Instead, this standard will recognize and model the general capabilities that can be potentially provided by advanced and/or legacy meter infrastructures, including two-way communication capabilities such as load control, dynamic pricing, outage detection, distributed energy resource (DER) control signals and on-request read. In this way, this standard will not be impacted by the specification, development and/or deployment of next generation meter infrastructures either through the use of standards or proprietary means.

The capabilities and information provided by a meter reading system are important for a variety of purposes, including (but not limited to) interval data, time-based demand data, time-based energy data (usage and production), outage management, service interruption, service restoration, quality of service monitoring, distribution network analysis, distribution planning, demand reduction, customer billing and work management. This standard also extends the CIM (Common Information Model) to support the exchange of meter data.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 60050-300, *International Electrotechnical Vocabulary – Electrical and electronic measurements and measuring instruments – Part 311: General terms relating to measurements – Part 312: General terms relating to electrical measurements – Part 313: Types of electrical measuring instruments – Part 314: Specific terms according to the type of instrument*

IEC 61968-1, *Application integration at electric utilities – System interfaces for distribution management – Part 1: Interface architecture and general requirements*

IEC 61968-2, *Application integration at electric utilities – System interfaces for distribution management – Part 2: Glossary*

IEC 61968-3, *Application integration at electric utilities – System interfaces for distribution management – Part 3: Interface for network operations*

IEC 61970-301, *Energy management system application program interface (EMS-API) – Part 301: Common information model (CIM) base*

IEC 62051-1, *Electricity metering – Data exchange for meter reading, tariff and load control – Glossary of terms – Part 1: Terms related to data exchange with metering equipment using DLMS/COSEM*

IEC 62055-31, *Electricity metering – Payment systems – Part 31: Particular requirements – Static payment meters for active energy (classes 1 and 2)*

IEC 62056 (all parts), *Electricity metering – Data exchange for meter reading, tariff and load control*

ISO 4217, *Codes for the representation of currencies and funds*

ISO 8601:2004, *Data elements and interchange formats – Information interchange – Representation of dates and times*

ANSI C12.19, *Utility Industry End Device Data Tables*

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 60050-300, IEC 61968-2, IEC 62051-1, IEC 62055-31 and the following apply.

Where there is a difference between the definitions in this document and those contained in other referenced IEC documents, then those defined in IEC 61968-2 shall take precedence over the others listed, and those defined in IEC 61968-9 shall take precedence over those defined in IEC 61968-2.

3.1.1
customer program
classification scheme for the sale of energy to consumers according to a particular tariff

The program may specify the purpose, conditions on the time of use, the service voltage(s), the volumes consumed, and/or other terms as a condition of the sale.

NOTE Utilities may promote particular programs to their industrial, commercial, agricultural, and residential customers in an effort to encourage a particular behaviour, or to make them aware of their options.

3.1.2
end device
equipment located at the end of the communication system, usually on the customer premises, which may perform functions such as metrology, connect/disconnect, load control, demand response, or other functions, and may have power relay and/or communications capability

This is represented within the CIM using the EndDeviceAsset class.

3.1.3
load control device
type of end device which can receive signals causing it to shed load for the purposes of maintaining network reliability and/or commercial agreements

3.1.4
meter
type of end device which performs metrology and supports the tariffing of the distribution and/or transmission network

This is represented using the CIM MeterAsset class, which is a subclass of EndDeviceAsset.

NOTE A meter could be defined as a 61850 device with logical nodes.

3.1.5
meter changeout
process of replacing an existing meter with a new meter

NOTE The installer will customarily follow a work order which specifies a given location, and usually requires that he or she capture readings from the old and new meters, and record the time and day in which the work was performed.

3.1.6
payment meter
electricity meter with additional functionality that can be operated and controlled to allow the flow of energy according to agreed payment modes

3.1.7
prepayment mode
payment mode in which automatic interruption occurs when available credit is exhausted

3.2 Abbreviations

AM	asset management
AMR	automated meter reading
AMI	advanced metering infrastructure
CIM	common information model
CIS	customer information system
COSEM	companion specification for energy metering
DLMS UA	device language message specification user association
DMS	distribution management system
IDR	interval data recorder
IEC	International Electrotechnical Commission
LC	load control
LMS	load management system
MAM	meter asset management
MDM	meter data management
MM	meter maintenance
MR	meter reading
MS	metering system
NO	network operations
OMS	outage management system
POS	point of sale
RF	radio frequency

VEE	validating, editing, and estimating
WM	work management

4 Reference and information models

4.1 General approach to metering systems

The spinning disk in an electromechanical meter generally serves as a pulse initiator to the meter recorder module. In a similar fashion, solid-state meters may also employ a metrology unit that generates pulses which represent a fraction of a kWh, and if more sophisticated, the solid-state meter may have a meter recorder which is able to accumulate many different kinds of information and store it for presentation to the meter communications module using a message and table-driven protocol such as ANSI C12.19 or IEC 62056.

The most common metered data element is kWh, but many electricity meters can also capture kW, kVAr, kVArh, and other similar billing quantities. Some meters can also capture pure engineering quantities such as voltage, current, power factor, etc.

Some AMR systems attempt to add value to low-end meters by adding functionality that the meter may lack. For low-end meters (e.g., energy only), it is common for an AMR meter module to add the capability to perform demand metering, interval data recording (IDR), maintain an energization count, or even provide estimates of certain engineering quantities such as voltage.

Commercial and industrial meters often employ current transformers and voltage transformers to meter the actual service. Primary voltages and currents are scaled down using potential transformers (PTs) and current transformers (CTs) so that the meter does not have to be constructed to withstand the high voltages and currents actually delivered to the load. Secondary voltage or current values are those that are often measured directly by the meter. Secondary values are small percentage of the primary values that may actually delivered to or connected to the load. If secondary voltages and currents are measured by the meter, these can be converted back to primary values using the PT and CT ratios, which are just the ratio of primary to secondary values.

The metering system will convey meter data and other value-added data through the metering system network to the destination. Depending on the system, the journey may involve multiple steps through public or private networks, licensed or unlicensed RF spectrums, standardized or proprietary systems, in a one-way or two-way fashion.

Some general operations or services can be defined for a metering system. These general operations will translate to specific actions in the context of a given metering solution.

General operations can be scheduled or called on-demand. Each operation returns an answer with a status. A message encapsulates a general operation.

Examples of meter general operations (reading) are:

- reading phone calls;
- reading load curve;
- reading encryption keys;
- reading equipment configuration;
- reading contract parameters;
- reading functional maintenance data;
- reading technical maintenance data;
- metrological reading;

- reading status parameters;
- reading quality;
- reading date-time;
- meter reading;
- reading historical data;
- reading programming data.

Examples of meter general operations (writing) are:

- writing client parameters;
- writing season parameters;
- writing date-time changing seasons;
- writing encryption keys;
- writing first time in-service;
- writing calling parameters;
- writing public holiday schedule;
- writing self-read parameters;
- writing status parameters;
- writing programming parameters (pricing);
- writing defect parameters;
- writing minimal defect threshold period;
- writing nominal voltage level.

Readers of IEC 61968-9 interested in standards concerning meter reading should refer to IEC 62056, DLMS UA (device language message specification user association) and the COSEM model: Companion specification for energy metering. COSEM specifies 3 steps:

a) the meter model and data identification;
b) the mapping of the model into protocol data units;
c) the transportation of the bits.

4.2 Reference model

4.2.1 General

The following diagrams serve as reference models and provide examples of the logical components and data flows related to this International Standard. Subclause 4.5.2 provides references to terms that are defined by the CIM.

The “meter” is treated as an “end device”. The end device may contain a metrology capability, it may contain a communications capability, it may be a load control unit, and it may contain a mixture of many different types of functionality. Figure 1 attempts to describe the concept of essentially a shopping-list of functionality which may be available in the (logical or physical) end device.

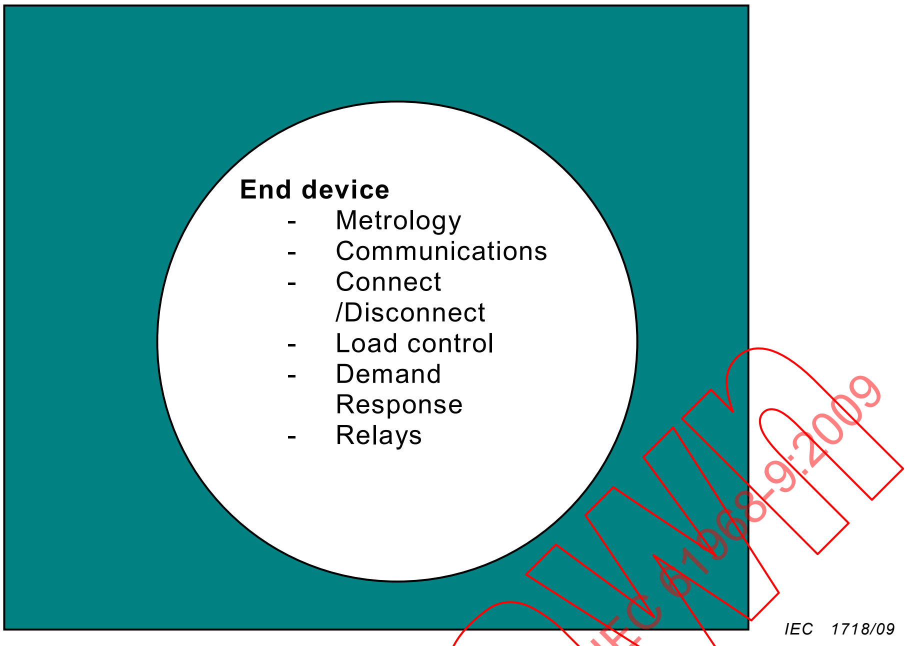


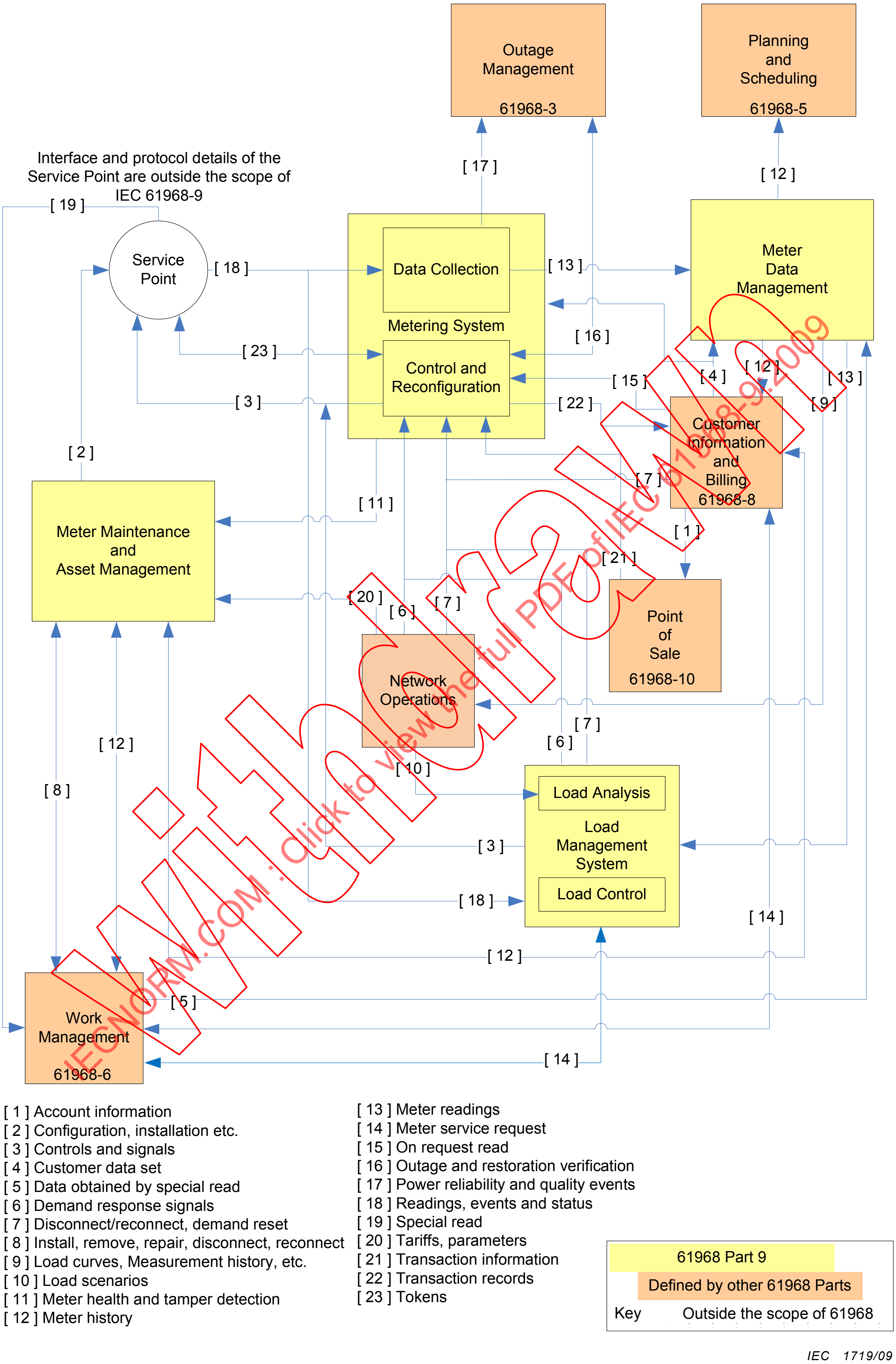
Figure 1 – Example of an end device with functions

The end device itself, while presenting a myriad of functionality, is not the subject of this standard. There are many interactions which ultimately affect the end device but occur as a result of a communication between systems at a much higher level in the control architecture. The interaction between systems is within the scope of this standard, and Figure 2 attempts to describe the exact systems involved, and provide an overview to the stakeholder landscape.

From the perspective of this standard, an end device:

- has a unique identity;
- is managed as a physical asset;
- may issue events;
- may receive control requests;
- may collect and report measured values;
- may participate in utility business processes.

The following diagrams of Figures 2 and 3 describe the flows between the components in the reference model. Two diagrams are used in order to reduce the complexity that would be experienced by a single diagram. The numbers in brackets provide linkages to the flow definitions.



[1] Account information
[2] Configuration, installation etc.
[3] Controls and signals
[4] Customer data set
[5] Data obtained by special read
[6] Demand response signals
[7] Disconnect/reconnect, demand reset
[8] Install, remove, repair, disconnect, reconnect
[9] Load curves, Measurement history, etc.
[10] Load scenarios
[11] Meter health and tamper detection
[12] Meter history
[13] Meter readings
[14] Meter service request
[15] On request read
[16] Outage and restoration verification
[17] Power reliability and quality events
[18] Readings, events and status
[19] Special read
[20] Tariffs, parameters
[21] Transaction information
[22] Transaction records
[23] Tokens

Figure 2 – IEC 61968-9 reference model

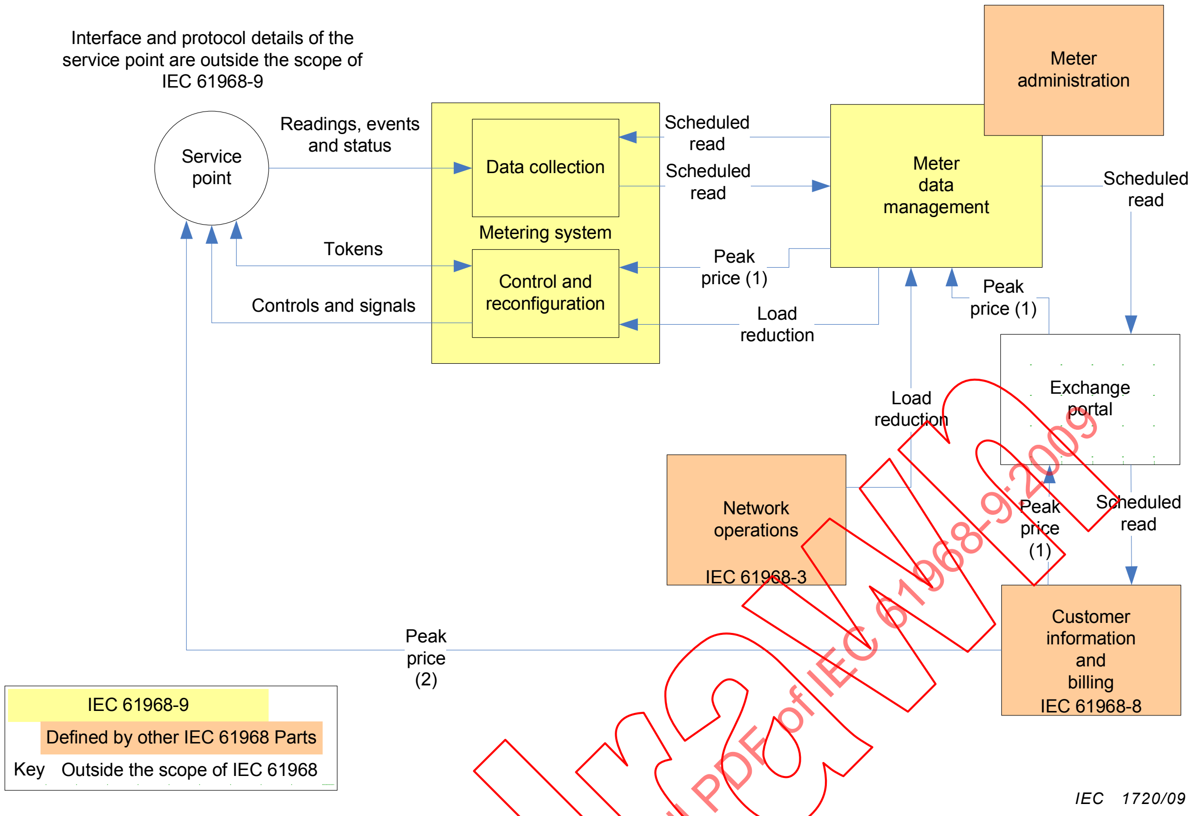


Figure 3 – IEC 61968-9 reference model with customer information and billing system

The reference architecture reflects five main logical components (potentially realized as systems or subsystems) related to metering:

a) metering system, potentially including data collection and control and reconfiguration functionality;

b) meter data management;

c) meter maintenance;

d) load management, potentially including load control and load analysis functionality;

e) meter asset management;

f) meter administration functions, which may be a part of meter data management.

The metering system and meter maintenance system components may be bound to a specific meter and associated communication technology or they may support multiple meters (and more generically end devices) types and associated communication technology. The meter data management system component may support and consolidate meter data from more than one metering system. The metering system may also support consolidation of various measurement and data collection implementations, providing a consistent interface to the meter data management system component.

4.2.2 Metering system (MS) – Data collection

The tasks of the data collection sub-component within the metering system may include:

- readings and status collection, obtained through either manual or automated means, on a scheduled or on-request basis;
- transmission of meter readings and status to a meter data management system;
- transmission of power reliability and quality event data to outage management, network operations, and capacity planning systems;

- transmission of communication network health information to those responsible for maintaining the communications network.

It is important to note that metering systems are significantly diverse with respect to technologies used, protocols used, capabilities and frequency of data collection. The details of the internals of meters, communication transports and protocols are outside the scope of this International Standard.

A handheld device used by a human meter reader could be regarded as a metering system. In a typical meter reading by human, a reader inputs meter data into a handheld device in a field. The handheld device is not connected to a communication network in a field. At the end of day work, the handheld device is connected to a communication network at reader’s home (not in an office), and meter data is uploaded to a meter data management system. In this case, this communication network should be part of a metering system. Data collection is described in IEC 61968-1 as “MR-RMR.”

4.2.3 Metering system (MS) – Control and reconfiguration

The tasks of the control and reconfiguration subcomponent within the metering system are:

a) primary interface in executing meter control commands;

b) communicating payment system information;

c) act as a communication gateway for load control devices;

d) service connect/disconnect;

e) configuration of tariff units of measure and calendar;

f) configuration of power quality measurement;

g) configuration of meter event recording;

h) relay of load control signals;

i) configuration of meter identity and security credentials;

j) fraud detection.

This subcomponent is identified separately within the metering system in order to recognize the existence of metering systems that do not have the ability to send messages to meters. IEC 61968-1 describes this subcomponent as “MR-MOP-Meter Configuration.”

4.2.4 Load control

The metering system infrastructure may often be used as a communication gateway to load control units (IEC 61968-1 describes this as “MR-LDC”). Load control units are end devices with load control (LC) capability. These are wired to control individual, target devices. End devices with LC functionality can take on different forms. Quite often a dedicated LC unit can be located at (or near) the device to be controlled. Another approach is to use meters that have relays which are configured to serve as LC devices. Still another approach is to interface with a customer energy management system (which would be another type of end device).

4.2.5 Load management system (LMS)

A load management system is used to manage and control load by the utility in order to promote system reliability. A load management system may perform load forecasting, contingency analysis, and other simulations prior to issuing a load control command. This function largely falls under the domain of IEC 61968-5 (operational planning and optimization).

4.2.6 Outage management system (OMS)

An outage management system (OMS) is used by distribution operators to detect and track outages, and to assist in the process of verification and/or restoration. An OMS typically combines (or has ties to) functionality such as network operations fault management, operational planning and optimization, network operation simulation, maintenance and construction, maintenance and inspection, and work scheduling and dispatching.

The metering system can be an important source of information for identifying the existence and extent of outages, and can be used to verify the restoration of outages. The MS might have the ability to publish outage and restoration data to the OMS as it finds it. This type of information typically joins customer call-ins in the OMS to allow it to better predict the location of the outage. However, due to the time-sensitive nature of outage detection, there is also the potential need for a request/reply interface with the MS, where the OMS can ask for specific equipment on the distribution network to be tested by the MS, and an energization status returned for analysis. The request/reply interface can be used by the OMS to supply critical data needed to make an outage prediction, or by a dispatcher who would like to verify that power has been restored to all of the metered endpoints downstream of a particular switch prior to sending the lineman to the next location.

4.2.7 Meter asset management (MAM) system

Utilities will employ some form of asset management software in an effort to maintain detailed records regarding their physical assets. Asset management is treated categorically in IEC 61968-4. However, metering has such special requirements, that it is common for a utility to use specialized meter asset management software. The software inventories the asset – providing a record of its physical attributes as well as its location. For sake of discussion, IEC 61968-9 will talk about a MAM system which is closely coupled to the MS and MDM, though some implementations will successfully generalize the asset management application sufficiently so that it can live within a more generic AM system.

4.2.8 Meter data management (MDM) system

From a historical perspective, it was common for a utility to have more than one automated meter reading system. Alternatively, a utility might outsource meter reading services to one or more third-party service providers who operate an AMR system and/or read the meters manually. The meter data management (MDM) system is used to provide a common repository, and point of management and access of meter data that is collected from disparate metering systems. In addition to data aggregation, quite often the MDM will also make an effort to scrutinize the data collected from the various metering systems, and provide a validating, editing, and estimating (VEE) capability. (IEC 61968-1 includes these functions under “MR-RMR-Meter data aggregation,” and “MR-MOP-Meter data management.”)

4.2.9 Customer information system (CIS)

A CIS will typically encompass functionality related to customer care and billing. This is a subject which is external to the IEC 61968 standard (refer to customer account management (EXT-ACT)). The billing function is driven by readings, typically demand or time-of-use, obtained from the meter. The CIS is also often involved with processes related to billing inquires, meter disconnect and meter reconnect, rate program changes.

4.2.10 Network operations (NO)

Network operations (IEC 61968-3) may occasionally need to issue load control and pricing signals to meters. This can be done for both economic and emergency reasons.

4.2.11 Meter maintenance (MM)

Meter maintenance is responsible for functionality related to the configuration and installation of meters. This type of functionality generally falls under meter asset management, or asset management in general. Performing meter maintenance may require exchanges with work management.

4.2.12 Planning

The planning function is described in operational planning and optimization, network operation simulation (OP-SIM).

4.2.13 Work management (WM)

A work management system is responsible for work that is performed by field resources. This subject is covered in maintenance and construction maintenance and inspection (MC-MAI).

With respect to metering, WM includes the installation, maintenance and replacement of meters. This may also involve the process of special reads.

4.2.14 Point of sale (POS)

A point of sale (POS) system is used for the management of payment meters, where a customer either purchases a token or makes a payment for service.

4.2.15 Meter

The meter records the data used for tariffing public networks, and data used for network balancing mechanisms.

Readings captured by the metering system are collected by a system such as the MDM before being presented for billing purposes. Billing entities may correct the data, or, in some regions, the energy supplier may perform validating, editing, and estimating (VEE) according to rules established by the appropriate supervising regulatory agency. In any case, those corrections are made available to the user who requests them.

Where this International Standard refers to a 'meter', it should be realized that a 'meter' is an end device that has metrology capability, it may or may not have communications capability, it may or may not have connect/disconnect capability, or a host of other capabilities. Given that a meter will have metrology capability, it will in all likelihood meter kWh, but possibly also demand, reactive energy and demand, time of use quantities, interval data, engineering quantities and more.

4.2.16 Load control devices

Load control devices are used to control loads at a ServiceDeliveryPoint. The metering system may often have a communication network which can be used for transmitting load control signals to various CommunicationsAsset(s) in order to control the load presented by the EndDeviceAsset(s). Alternatively, the communication network could be used to communicate demand response (price) signals to the CommunicationsAsset(s) in order to affect the load presented by the EndDeviceAsset(s).

4.3 Interface reference model

It is not the intention of this standard to define the applications and systems that vendors should produce. It is expected that a concrete (physical) application will provide the functionality of one or more abstract (logical) components as listed in this standard. These abstract components are grouped by the business functions of the interface reference model.

In this standard, the term abstract component is used to refer to that portion of a software system that supports one or more of the interfaces defined in IEC 61968-3 to IEC 61968-9. It does not necessarily mean that compliant software is delivered neither as separate modules nor as a single system.

IEC 61968-1 describes infrastructure services common to all abstract components while IEC 61968-3 to IEC 61968-9 define the details of the information exchanged for specific types of abstract component.

IEC 61968 series defines the following.

a) An inter-application infrastructure is compliant if it supplies services defined in IEC 61968-1 to support at least two applications with interfaces compliant to clauses of IEC 61968-3 to IEC 61968-9.

b) An application interface is compliant if it supports the interface standards defined in IEC 61968-3 to IEC 61968-9 for the relevant abstract components defined in the interface reference model.

c) An application is only required to support interface standards of the applicable components listed under abstract components. An application is not required to support interfaces required by other abstract components of the same business sub-function or within the same business function. While this standard primarily defines information exchanged among components in different business functions, it will occasionally also define information exchanged among components within a single business function when a strong market need for this capability has been realised.

4.4 Meter reading and control functions and components

It should be noted that the message types defined in this document: 'IEC 61968-9: Interfaces for meter reading and control', may be sent or received by any type of component within a distribution management system (DMS) system.

Table 1 shows these functions and typical abstract components that are expected to be producers of information for these message types. Typical consumers of the information include, but are not restricted to, the other components as listed in IEC 61968-1.

Table 1 – Business functions and abstract components

Business functions	Business sub-functions	Abstract components
Meter reading and control (MR)	Metering system (MS)	Data collection
		End point controls
		End point reconfiguration
		Disconnect/reconnect
		Demand reset
		On request read
		Point of sale
		Outage detection and restoration verification
		Power reliability and quality events
		Metering system events
	Meter maintenance and asset management	End point install, configure, remove, repair, disconnect, reconnect
		End point asset history
		End point reconfiguration
		Special read
		Meter service request
		Tariffs
	Meter data management (MDM)	Meter data repository
		Usage history
		Validation, estimation and editing
		Customer billing data
	Load management (LM)	Load analysis
		Load control
		Demand response
		Performance measurements
		Risk management

4.5 Static information model

4.5.1 General

The information model relevant to meter reading and control consists of classes that provide a template for the attributes for each message.

The classes are defined in detail in IEC 61970-301.

4.5.2 Classes for meter reading and control

Table 2 lists classes used within message types. Usually, all the attributes of these classes are contained within a message type. The descriptions provided describe usage within this part.

Classes described as type "Asset" are defined in the 61968/Assets package of the CIM.

Classes described as type "Customer" are defined in the 61968/Customers package of the CIM.

Classes described as type "Metering" are defined in the 61968/Metering package of the CIM.

Classes described as type “Prepayment” are described in the 61968/PaymentMetering package of the CIM.

Classes described as type "Profile" are contextual profiles defined for IEC 61968-9 that describe message definitions defined using CIM objects.

Table 2 – Classes for meter reading and control

Noun	Type	Description
AuxiliaryAccount	Prepayment	Variable and dynamic part of AuxiliaryAgreement, generally representing the current state of the account related to the outstanding balance defined in AuxiliaryAgreement.
AuxiliaryAgreement	Prepayment	An ad-hoc auxiliary account agreement associated with a customer agreement, not part of the customer's account, but typically subject to formal agreement between customer and supplier (utility). Typically this is used to collect revenue owing by the customer for other services or arrears accrued with the utility for other services. It is typically linked to a prepaid token purchase transaction, thus forcing the customer to make a payment towards settlement of the auxiliary account balance whenever he needs to purchase a prepaid token for electricity. The present status of AuxiliaryAgreement can be defined in the context of the utility's business rules, for example: enabled, disabled, pending, over recovered, under recovered, written off, etc.
AuxiliaryAgreementConfig	Priofile	Message profile for AuxiliaryAgreements.
Card	Prepayment	Documentation of the tender when it is a type of card (credit, debit, etc).
Cashier	Prepayment	The operator of the point of sale for the duration of CashierShift. Cashier is under the exclusive management control of Vendor.
CashierShift	Prepayment	The operating shift for a cashier, during which he may transact against the CashierShift, subject to VendorShift being open.
Charge	Prepayment	A charge element associated with other entities such as tariff structures, auxiliary agreements or other charge elements. The total charge amount applicable to this instance of charge is the sum of fixedPortion plus percentagePortion.
Cheque	Prepayment	The actual tender when it is a type of cheque. Cheque reference number (as printed on the cheque) is specified in 'name.'
ComFunction	Metering	Communication function of communication equipment or a device such as a meter.
ComMediaAsset	Assets	Communication media such as fibre optic cable, power-line, telephone, etc.
ConsumptionTariffInterval	Prepayment	One of a sequence of intervals defined in terms of consumption quantity of a service such as electricity, water, gas, etc. It is typically used in association with TariffProfile to define the steps or blocks in a step tariff structure, where startValue simultaneously defines the entry value of this step and the closing value of the previous step. Where consumption is &gt;= startValue, it falls within this interval and where consumption is &lt; startValue, it falls within the previous interval.
Customer	Customers	Organisation receiving services from ServiceSupplier.
CustomerAccount	Customers	Assignment of a group of products and services purchased by the customer through a CustomerAgreement, used as a mechanism for customer billing and payment. It contains common information from the various types of CustomerAgreements to create billings (invoices) for a customer and receive payment.
CustomerAccountConfig	Profile	Message profile for CuustomerAccounts.
CustomerAgreement	Customers	Agreement between the customer and the ServiceSupplier to pay for service at a specific ServiceLocation. It records certain billing information about the type of service provided at the ServiceLocation and is used during charge creation to determine the type of service.
CustomerAgreementConfig	Profile	Message profile for CustomerAgreements.
CustomerMeterDataSet	Profile	The CustomerMeterDataSet includes one or more CustomerAccounts for one or more ServiceLocations for one or more ServiceDeliveryPoints. TheCustomerMeterDataSet may include one or more EndDeviceGroups.

Noun	Type	Description
DemandResponseProgram	Metering	Demand response program.
DeviceFunction	Metering	Function performed by a device such as a meter, communication equipment, controllers, etc.
ElectricmeteringFunction	Metering	Functionality performed by an electric meter.
EndDeviceAsset	Metering	The EndDeviceAsset is equipment that performs the role of an end device. It may contain functionality such as metrology, communications, load control, connect/disconnect, or other capabilities. It is known as "the meter", "a smart meter", an "advanced meter", "air conditioner", "refrigerator", "pool pump", etc. that a CommModule and/or MeterAsset may monitor and/or control. The asset may be owned by a consumer, a service provider, utility or other party.
EndDeviceAssets	Profile	Message used to convey descriptions of one or more EndDeviceAssets.
EndDeviceControl	Metering	Used to issue commands to EndDeviceAssets such as meters. May be addressed by EndDeviceAsset or by EndDeviceGroup. EndDeviceControls may have control types and parameters.
EndDeviceControls	Profile	Message used to convey one or more EndDeviceControls
EndDeviceEvent	Metering	Used to report events detected by end devices such as meters. Each EndDeviceEvent has an event type and a timestamp.
EndDeviceEvents	Profile	Message used to convey one or more EndDeviceEvents.
EndDeviceFirmware	Profile	Profile for EndDeviceAsset firmware configuration messages.
EndDeviceGroup	Metering	An EndDeviceGroup is used for grouping end devices for a variety of purposes including, but not limited to, load control and other types of demand response. An EndDeviceGroup may belong to another EndDeviceGroup, and an end device may belong to zero or more EndDeviceGroups. In some cases, the group ID is maintained within the end device, but in other cases it can be managed by a metering system.
IntervalBlock	Metering	Time sequence of readings of the same ReadingType. Contained IntervalReadings may need conversion through the application of an offset and a scalar defined in associated pending.
IntervalReading	Metering	Data captured at regular intervals of time. Interval data could be captured as incremental data, absolute data, or relative data. The source for the data is usually a tariff quantity or an engineering quantity. Data is typically captured in time-tagged, uniform, fixed-length intervals of 5 min, 10 min, 15 min, 30 min, or 60 min. NOTE Interval data is sometimes also called "interval data readings" (IDR).
MerchantAccount	Prepayment	The operating account controlled by MerchantAgreement, against which vendor may vend tokens or receipt payments. Transactions via VendorShift debit the account and bank deposits via BankStatement credit the account.
MerchantAgreement	Prepayment	A formal controlling contractual agreement between supplier and merchant, in terms of which merchant is authorised to vend tokens and receipt payments on behalf of supplier. Merchant is accountable to supplier for revenue collected at PointOfSale.
MeterAsset	Metering	The MeterAsset class is used to describe meters. A MeterAsset is a type of EndDevice typically used to measure and potentially monitor a customer load. The EndDeviceAsset class definition should be used as the basis for MeterAsset class.
MeterAssetConfig	Profile	Message profile for MeterAssets configuration messages.
MeterAssetReading	Profile	Profile for manually obtaining MeterReadings for MeterAssets.
MeterReading	Metering	A set of values obtained from the meter. Each MeterReading may have multiple ReadingTypes, and each ReadingType may contain multiple values.
MeterReadings	Profile	Profile for conveying MeterReadings.

Noun	Type	Description
MeterReadSchedule	Profile	A MeterReadSchedule message is used to schedule meter readings.
MeterServiceRequests	Profile	A meter service request is a type of work that can be used for a variety of meter service related activities. These activities would include meter installation, meter change out, customer disconnect/reconnect, , etc.
MeterServiceWork	Metering	Work involving meters.
MeterSystemEvents	Profile	Profile for meter system event messages.
Pending	Metering	When present, a scalar conversion that is associated with IntervalBlock and which needs to be applied to every contained IntervalReading value. This conversion results in a new associated ReadingType, reflecting the true dimensions of interval reading values after the conversion.
PointOfSale	Prepayment	Logical point where transactions take place with operational interaction between cashier and the payment system; in certain cases PointOfSale interacts directly with the end customer, in which case cashier might not be a real person: for example a self-service kiosk or over the internet.
ReadingQuality	Metering	Quality of a specific reading value or interval reading value. Note that more than one quality may be applicable to a given reading. Typically not unsed unless problems or unusual conditions occur (i.e., quality for each reading is assumed to be 'Good' unless stated otherwise in associated ReadingQuality).
ReadingType	Metering	Type of data conveyed by a specific reading.
Receipt	Prepayment	Record of total receipted payment from customer.
ReceiptRecord	Profile	Profile for receipt messages.
ReceiptSummary	Prepayment	Record of detail of receipts pertaining to one shift of operation (one record per 'tenderKind').
Register	Metering	Display for quantity that is metered on an end device such as a meter.
SDPLocation	Metering	Location of an individual service delivery point. For residential or most businesses, it is typically the location of a meter on the customer's premises. For transmission, it is the point(s) of interconnection on the transmission provider's transmission system where capacity and/or energy transmitted by the transmission provider is made available to the receiving party. The point(s) of delivery is specified in the service agreement.
SDPLocationConfig	Profile	Profile for SDPLocation configuration messages.
ServiceCategory	Customers	Category of service provided to the customer.
ServiceCategoryConfig	Profile	Profile for ServiceCategory configuration messages.
ServiceDeliveryPoint	Metering	Logical point on the network where the ownership of the service changes hands. It is one of potentially many service points within a ServiceLocation, delivering service in accordance with a CustomerAgreement. Used at the place where a meter may be installed.
ServiceDeliveryPointConfig	Profile	Profile for ServiceDeliveryPoint configuration messages.
ServiceLocationConfig	Profile	Profile for ServiceLocation configuration messages.
ServiceSupplier	Prepayment	Organisation that provides services to customers.
Shift	Prepayment	Generally referring to a period of operation or work performed. Whether shift is open/closed can be derived from attributes 'activiryInterval.start' and 'activityInterval.end'. The grand total for receipts (i.e., cumulative total of all actual receipted amounts during this shift; bankable + non-bankable; excludes rounding error totals) can be derived from Receipt attributes: =sum(Receipt.receiptAmount) ; includes bankable and non-bankable receipts.
SupplierConfig	Profile	Profile for supplier configuration messages.

Noun	Type	Description
Tariff	Prepayment	Document, approved by the responsible regulatory agency, listing the terms and conditions, including a schedule of prices, under which utility services will be provided. It has a unique number within the state or province. For rate schedules it is frequently allocated by the affiliated public utilities commission.
TariffConfig	Profile	Profile for tariff configuration messages.
TariffProfile	Prepayment	A schedule of charges; structure associated with tariff that allows the definition of complex tariff structures such as step and time of use when used in conjunction with TimeTariffInterval and Charge. Inherited 'status.value' is defined in the context of the utility's business rules, for example: active, inactive, etc.
Tender	Prepayment	Tender is what is "offered" by the customer towards making a payment and is often more than the required payment (hence the need for 'change'). The payment is thus that part of the Tender that goes towards settlement of a particular transaction. Tender is modelled as an aggregation of cheque and card. Both these tender types can exist in a single tender bid thus 'accountHolderName' must exist separately in each of cheque and card as each could have a different account holder name.
TimeTariffinterval	Prepayment	One of a sequence of time intervals defined in terms of real time. It is typically used in association with TariffProfile to define the intervals in a time of use tariff structure, where startDateTime simultaneously determines the starting point of this interval and the ending point of the previous interval.
Transaction	Prepayment	The record of details of payment for service or token sale.
TransactionRecord	Profile	Profile for Transactions messages.
TransactionSummary	Prepayment	The record of detail of payment transactions pertaining to one shift of operation (one record per 'transactionKind').
Transactor	Prepayment	The entity that ultimately executes the transaction and who is in control of the process; typically this is embodied in secure software running on a server that may employ secure hardware encryption devices for secure transaction processing.
Vendor	Prepayment	The entity that owns PointOfSale and contracts with Cashier to receipt payments and vend tokens using the payment system. Vendor has a private contract with and is managed by merchant who is a type of organisation. Vendor is accountable to Merchant for revenue collected, who is in turn accountable to supplier.
VendorShift	Prepayment	The operating shift for a vendor during which he may transact against the merchant's account. It aggregates transactions and receipts during the shift and periodically debits a merchant account. The totals in VendorShift should always = sum of totals aggregated in all cashier shifts that were open under the particular vendor shift.
NOTE The class definitions provided here are for convenience purposes only. The normative definitions are provided by the CIM.		

4.5.3 Classes related to meter reading and control

Table 3 lists those classes that are associated with meter reading and control classes but only the name of an instance is given within messages defined in this standard. The detailed attributes of these classes are used in message types defined in other parts of IEC 61968.

Table 3 – Classes related to meter reading and control

Related class	Reference	Description
Circuit	Collection Package	Static collection of conducting equipment originating at a main distribution center and supplying one or more secondary distribution centers, one or more branch-circuit distribution centers, or any combination of these two types of equipment. It is the source to the next normally open point.
LoadDataSet	IEC 61968-7	Customer, supply point, or area load data. May include meter records for customers (where regulations allow).
Organisation	IEC 61968-4	This class is used to identify companies or divisions within companies. Organisations might have roles as utilities, contractors, suppliers, manufacturers, etc.
PowerSystemResource	IEC 61970-301	An entity that describes the logical view of a component part of the utility business. PowerSystemResources are further classified as EquipmentContainers e.g. Substations, ConductingEquipment, ProtectionEquipment etc. Instances of type PowerSystemResource may be related to instances of type asset.
PowerTransformer	IEC 61970-301	An electrical device consisting of two or more coupled windings, with or without a magnetic core, for introducing mutual coupling between electric circuits. Transformers can be used to control voltage and phase shift (MW flow).
ServiceLocation	Core2/Location Package	A Customer ServiceLocation has one or more ServiceDeliveryPoint(s). Meters are related to a ServiceDeliveryPoint. The location may be a point or a polygon depending on the specific circumstances. For distribution, the ServiceLocation is typically the location of the utility customer's premise. Because a customer's premise may have one or more meters, the ServiceDeliveryPoint is used to define the actual conducting equipment that the EndDeviceAsset attaches to at the utility customer's ServiceLocation. For Transmission, it is the point(s) of interconnection on the transmission provider's transmission system where capacity and/or energy transmitted by the transmission provider is made available to the receiving party.
NOTE The class definitions provided here are for convenience purposes only. The normative definitions are provided by the CIM.		

5 Meter reading and control message types

5.1 General

The purpose of this clause is to describe the message types related to IEC 61968-9. It is important to note that some of these message types may also be used by other parts of IEC 61968. The general approach to the realization of message structures and XML schemas for IEC 61968 messages is described in 61968-1.

Although they may be represented in sequence diagrams for context and completeness, this document does not describe message formats that are defined by other parts of IEC 61968. The key message payload structures defined by this part of IEC 61968 include the following:

- CustomerMeterDataSet;
- MeterAsset;
- MeterAssetReading;
- EndDeviceControls;
- EndDeviceEvents;
- MeterReadings;

- MeterReadSchedule;
- MeterServiceRequest;
- MeterSystemEvents;
- EndDeviceFirmware.

The normative XML schemas for messages defined by this document are provided in Annex H, providing more detailed, annotated descriptions of the message structures. Message structures are also diagrammed within this clause. The notation convention shows required elements with a solid outline, and option elements with dashed outlines.

It is also important to note that the use cases and sequence diagrams provided in this clause are informative in nature, and are intended to provide examples of usage for the normative messages definitions. There is no intent by this standard to standardize specific business processes.

5.2 End device event messages

5.2.1 General

End device events are viewed as a potential data stream which is not necessarily synchronized to the collection of meter reading for billing purposes. However, it may still be relevant to the billing process. A meter health alarm might be so severe that all readings from the meter have to be considered invalid. The data consumer (e.g. the MDM system) becomes responsible to subscribe to all of the available relevant data, and merge the findings appropriately.

5.2.2 Applications

5.2.2.1 Outage detection

When an outage is not due to a SCADA trip, electric utilities typically depend on the calls from the customers to identify the location of the fault. However, the use of a MS can provide another means to identify the location of trouble. It is often possible for a MS to determine that it has lost contact with a meter, in which case this can be reported for potential use by an outage management system. It is important to note that loss of communications with a meter may not be sufficient to identify the existence of an outage, as it is common for some MS infrastructures to occasionally lose contact with meters for brief periods of time.

Some MS systems may be prone to false alarms as a result of the particular technology used to communicate with the meters. Many vendors are working to improve their technology and the accuracy of the data. The MDM, much like the role it sometimes plays in cleaning up metered data for billing applications, in many instances, can also play a role in cleaning up outage data supplied by the MS before it is relayed to the OMS. Such a decision, to route outage data through the MDM, depends on the capability of the MS to supply accurate data, the ability of the MDM to clean up data without introducing excessive delays, and the ability of the OMS to tolerate false alarms and delays. To support outage analysis and filtering of bad data, the MS may supply audit-trail data and quality of measurement data for the outage event, much like it supplies audit-trail data for billing reads. Figure 4 describes a deployment in which the MDM is employed to broker outage information.

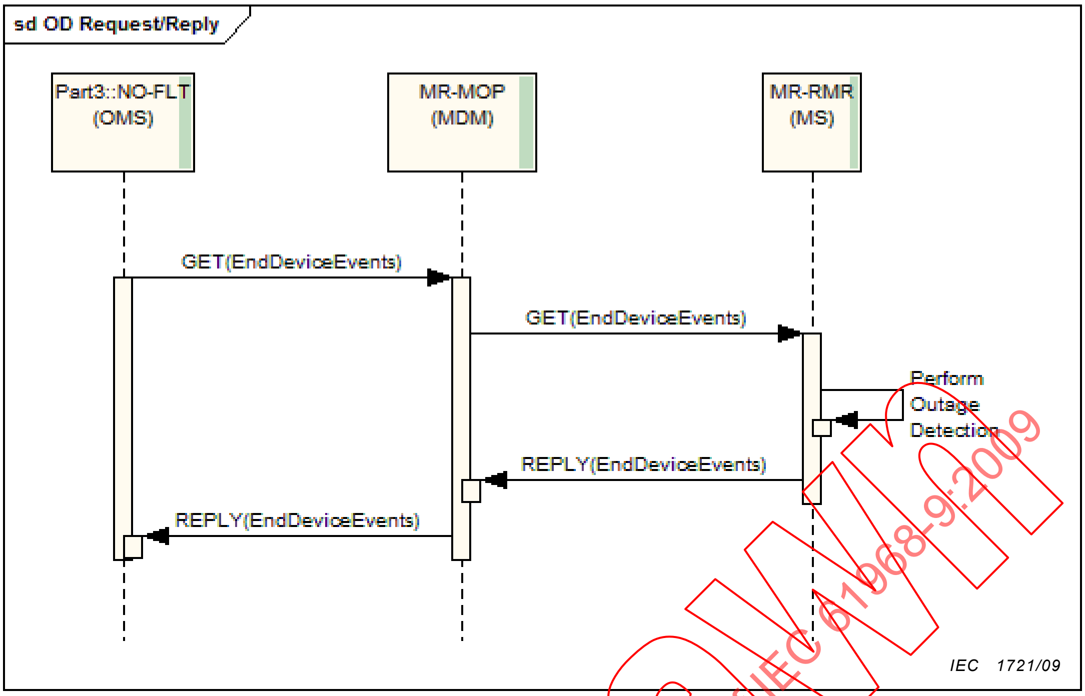


Figure 4 – Outage detection, request/reply message exchange, example 1

Outage management systems analyze the circuit in terms of network topology. The EndDeviceEvent.category can indicate an outage event as detected by an end device for consideration in the analysis, as well as other information such as trouble tickets.

The use of a MDM to broker outage data is at the discretion of the utility. In some deployments, the outage detection request from the OMS may be issued directly to the MS as in the example of Figure 5.

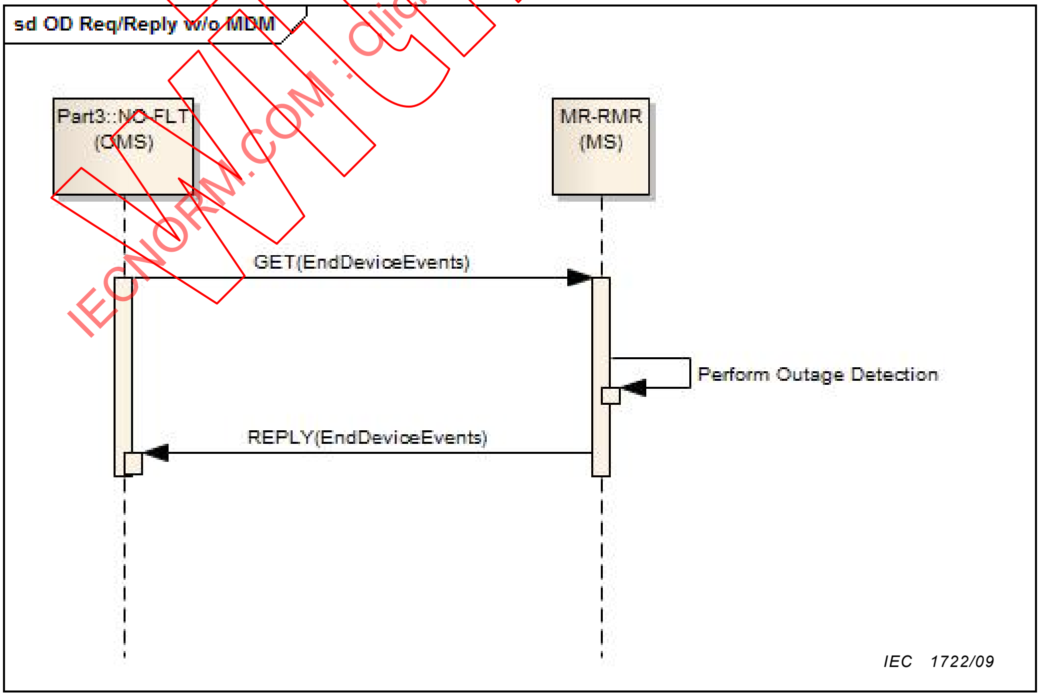


Figure 5 – Outage detection, request/reply message exchange, example 2

A reply (synchronous or asynchronous) from the MS will likely be in the form of zero or more EndDeviceEvents. The mRID will identify the end devices (e.g. meters) affected by the outage or restoration. Some data consumers that interact with the MS might only know how to deal with meter ID's. Depending on the deployment, it may be required to limit the scope of exchanged mRID's to that of meters ID's and not to that of power system resources in general. EndDeviceEvent.status can indicate "live" or "dead". EndDeviceEvent.category can be used to indicate that the status applies to a meter or power transformer. The reason for the event can be described (e.g. EndDeviceEvent.reason = "consecutiveFailCounter") and quantified (e.g. EndDeviceEvent.severity = "3").

While the request/reply exchange is useful whenever the OMS needs supplementary information, some MS are able to self-detect outages. A pub/sub exchange is ideal in these situations. Figure 6 shows such an exchange at a deployment where the MDM serves as an information broker. Figure 7 shows a deployment with information going directly from the MS to the OMS.

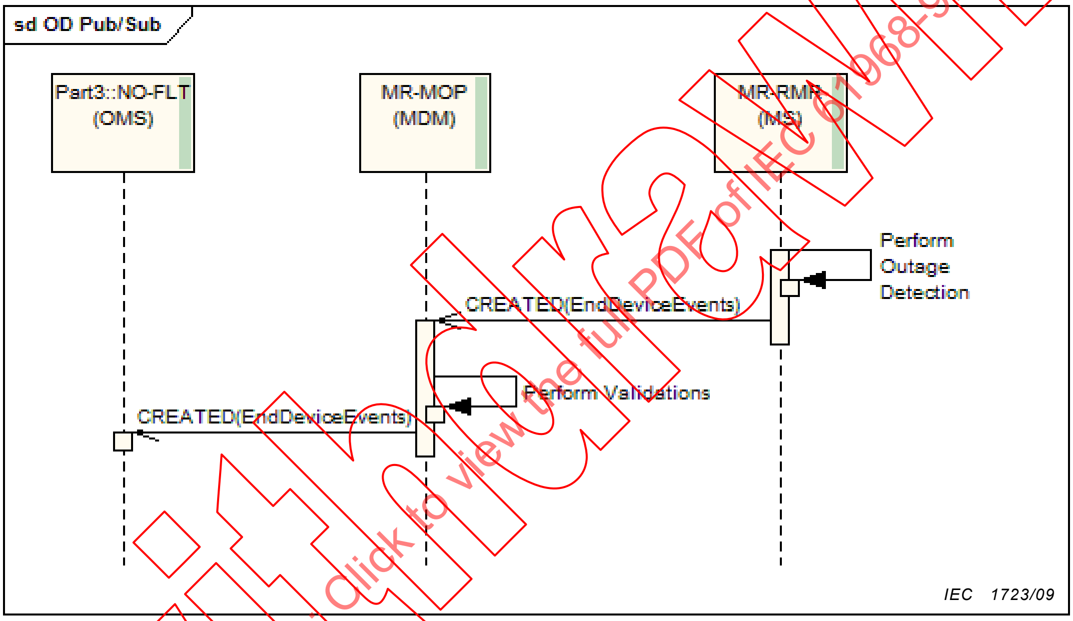


Figure 6 – Outage detection, publish/subscribe exchange, example 1

Not shown is the "SUBSCRIBE" that occurs typically only once upon system installation, and publication of events occurs repeatedly for the duration of the subscription.

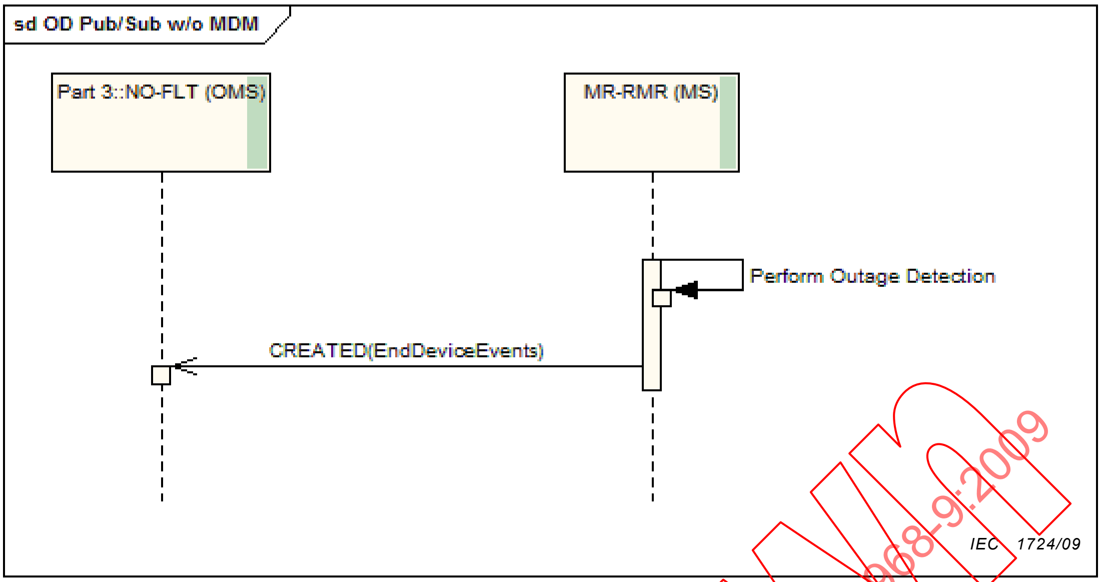


Figure 7 – Outage detection, publish/subscribe exchange, example 2

Some deployments may wish to limit the number of interfaces supported. The MeterReadings message structure also provides the means to convey EndDeviceEvents.

5.2.2.2 Meter health events

Some types of meters can sometimes generate meter health events, which can be used to identify issues with meter hardware, configuration or connection that should be resolved. The alarms include such things as diagnostic alarms, tamper alarms, or other unusual conditions. The severity of the alarm might range from a simple notification to “fatal”. Often times, resolution of meter health events require a site visit, so typically they result in the creation of a MeterServiceRequest. These events are communicated using EndDeviceEvent messages.

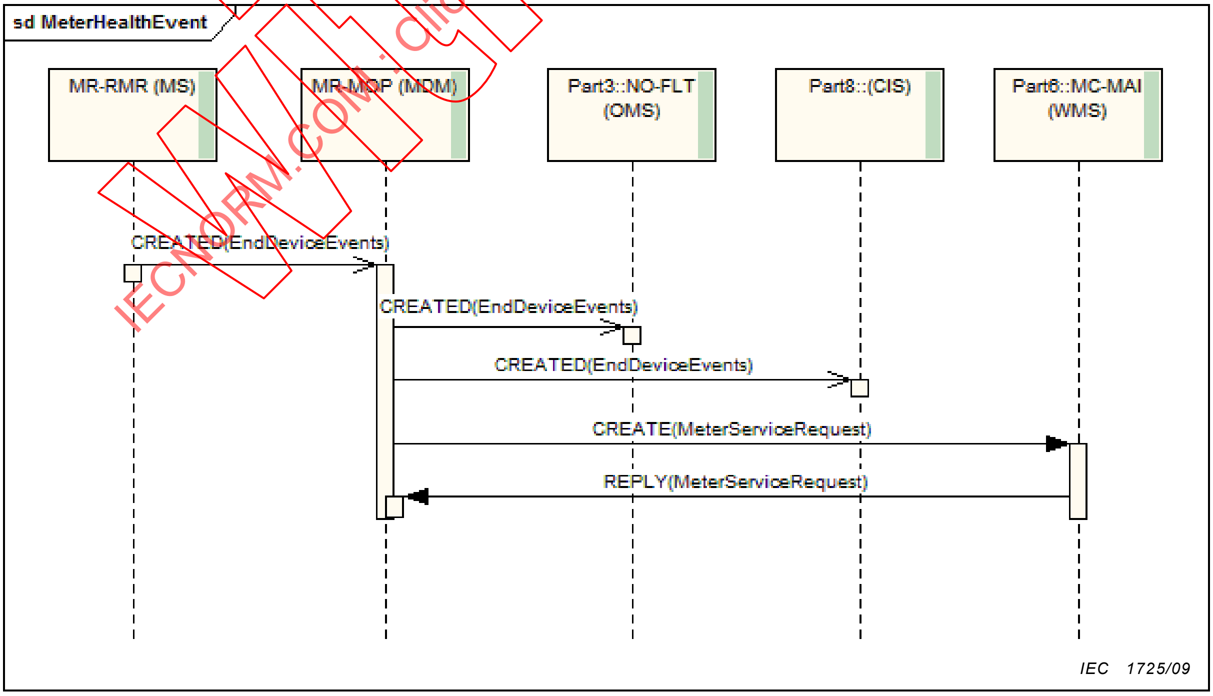


Figure 8 – Meter health event exchange, example 1

In some deployments, the MDM will be present to broker meter health data for other stakeholders as described in the scenario of Figure 8. Other installations however might not have a MDM or use it in this way. It is possible for the MS to publish data directly to the stakeholders that are equipped to receive it. Such an exchange is depicted in Figure 9.

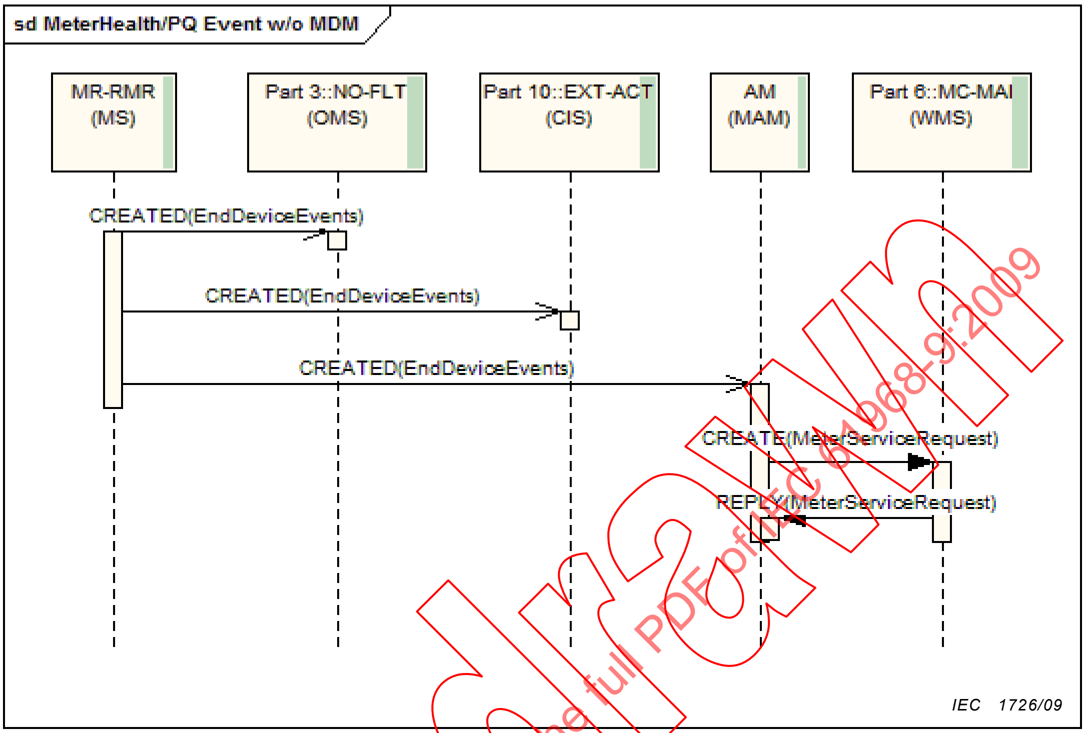


Figure 9 – Meter health event exchange, example 2

5.2.2.3 Power quality event

Meters may collect information related to power quality, including but not limited to, momentary outage events, sustained outage events, low or high voltage events, and high distortion events. This information could be used for outage analysis, maintenance scheduling or capacity planning. Power quality events are a subtype of EndDeviceEvent.

Power quality events may be brokered (i.e. publications managed) by a MDM (as described in Figure 10), or sent directly to the various stakeholders (as described in Figure 11).

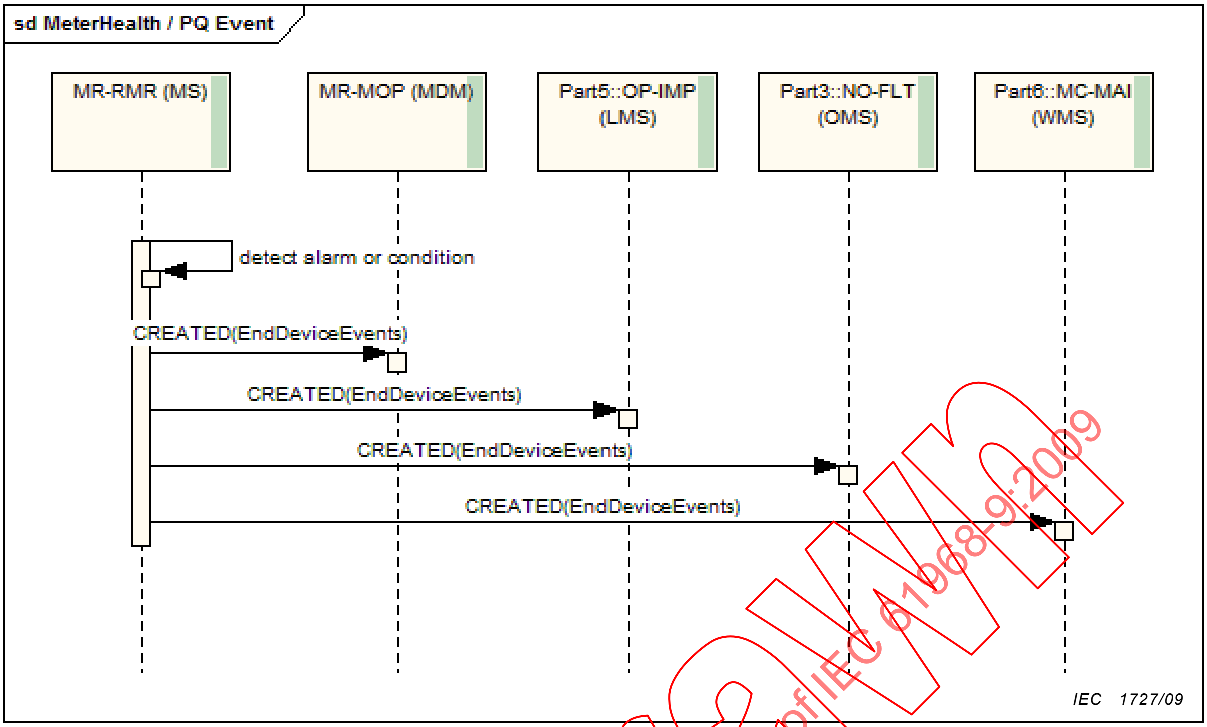


Figure 10 – Power quality event exchange, example 1

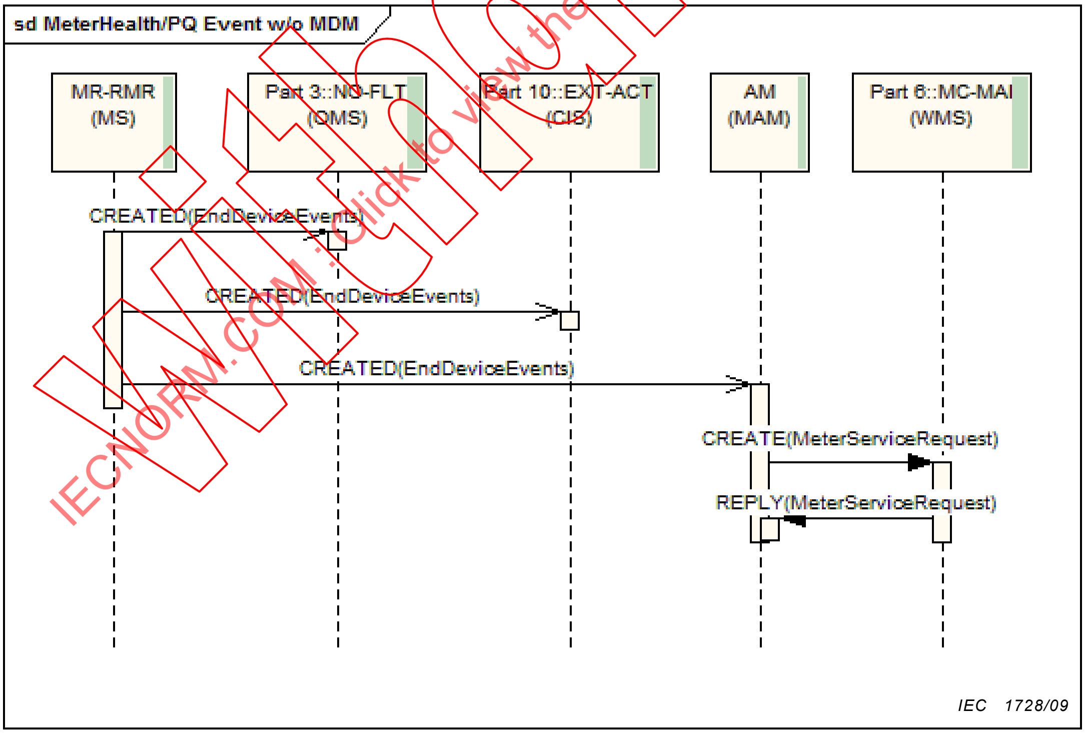
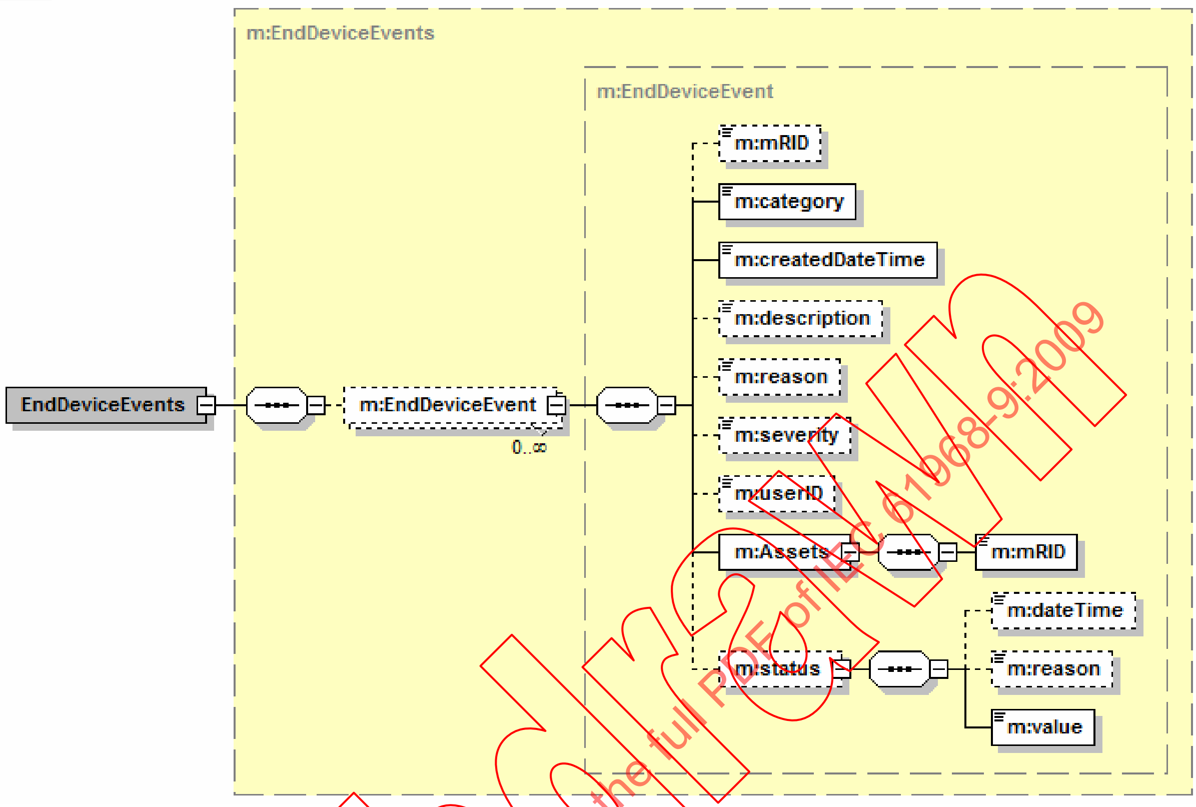


Figure 11 – Power quality event exchange, example 2

5.2.3 Message format

Meter event messages are implemented using EndDeviceEvent structures in order to support a wider variety of event sources than just meters. The EndDeviceEvent.category is an enumeration which indicates the type of event, such as outage detection, meter health, or power quality. The timestamp and mRID of the end device are also required. The message format is described in Figure 12.



IEC 1729/09

Figure 12 – End device event message format

As noted by Figure 12, only the mRID, timestamp and category elements are required. The category element will allow different event types (e.g. meter health, outage detection, etc.) to be differentiated. Different event types would allow for an EndDeviceEvent message to convey events related (but not limited) to:

- sustained outage detection;
- momentary outage detection;
- low voltage threshold detection;
- high voltage threshold detection;
- distortion;
- meter health;
- tamper detection;
- revenue event.

The detailed XML schema is provided in Clause H.4.

5.3 Synchronization between systems

5.3.1 General

In most cases, data may freely flow from the data producer to all of the data consumers (in parallel). In some cases however, systems add value to the data when they handle it. In this case, subscriptions must be established to route the data in sequence between systems.

5.3.2 Applications

5.3.2.1 Meter asset management

Most assets are assigned an ID by their system of record. (Alternatively, they may be assigned an mRID by an ID broker.) The mRID serves as a permanent key. When a data producer publishes information regarding changes to an asset, it can be understood that the mRID has not changed, but other information in the publication has. Information on the use of the SUBSCRIBE, CREATED, UPDATED, and DELETED verbs may be found in Annex A.

Subscriptions are typically established one time initially between systems (see Figure 13), using mechanisms provided by the messaging infrastructure as opposed to application-level messages. Publications flow as long as the subscription is maintained (see example in Figure 14).

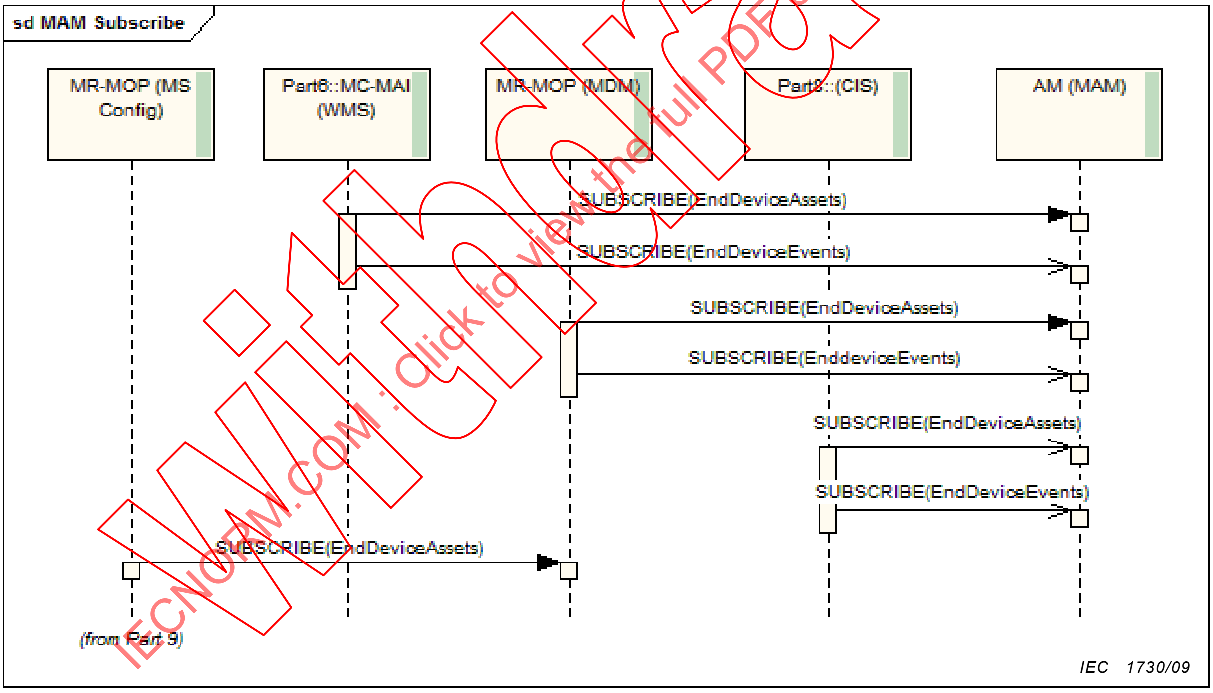

sd MAM Subscribe
MR-MOP (MS Config)
Part6::MC-MAI (WMS)
MR-MOP (MDM)
Part8::(CIS)
AM (MAM)
SUBSCRIBE(EndDeviceAssets)
SUBSCRIBE(EndDeviceEvents)
SUBSCRIBE(EndDeviceAssets)
SUBSCRIBE(EnddeviceEvents)
SUBSCRIBE(EndDeviceAssets)
SUBSCRIBE(EndDeviceEvents)
SUBSCRIBE(EndDeviceAssets)
(from Part 9)
IEC 1730/09

Figure 13 – Example of meter asset management subscriptions

5.3.2.2 End device acceptance into inventory

When an end device is purchased (a meter, a load control unit, whole-house connect/disconnect, etc.), the news of the purchase might propagate to the other systems. In this case, the event may be reported using either the CREATED (where the asset is new) or UPDATED (where the asset has been updated) verb. The EndDeviceAsset message is structured so that it can be used to present a bundle of DeviceFunctionality which describes the capability of the device identified by the Asset name. The EndDeviceAsset message can also be used to describe the relationship between components. For example, it can describe the relationship between a meter communication module, and the meter that hosts it.

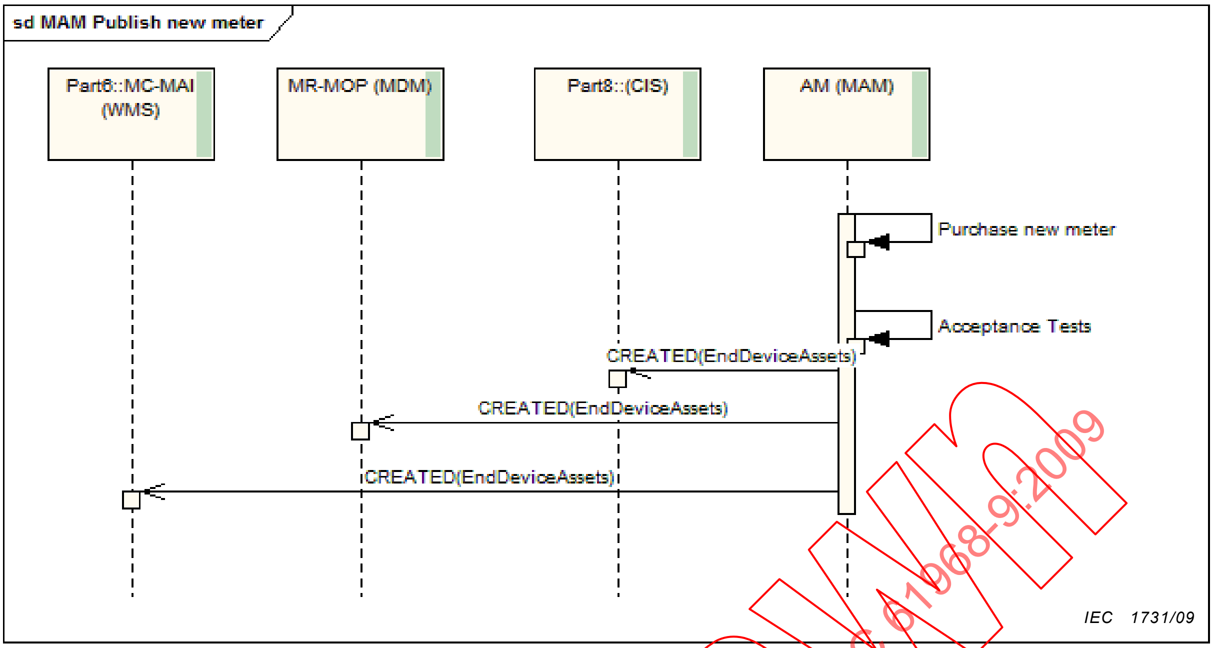


Figure 14 – Example of meter asset management publications

This information is conveyed by creating a meter asset (a subclass of EndDeviceAsset) and ActivityRecord for the new hardware. Test results from the meter vendor as well as any acceptance testing form the start of the ActivityRecord for the EndDeviceAsset.

5.3.2.3 End device retested

When a meter has been changed out (see the section on meter installation and removal), it is typically tested in the meter shop and recalibrated. The test results are added to the ActivityRecord. News of the recalibration, and the results can be sent from the WMS to the MAM as described in Figure 15.

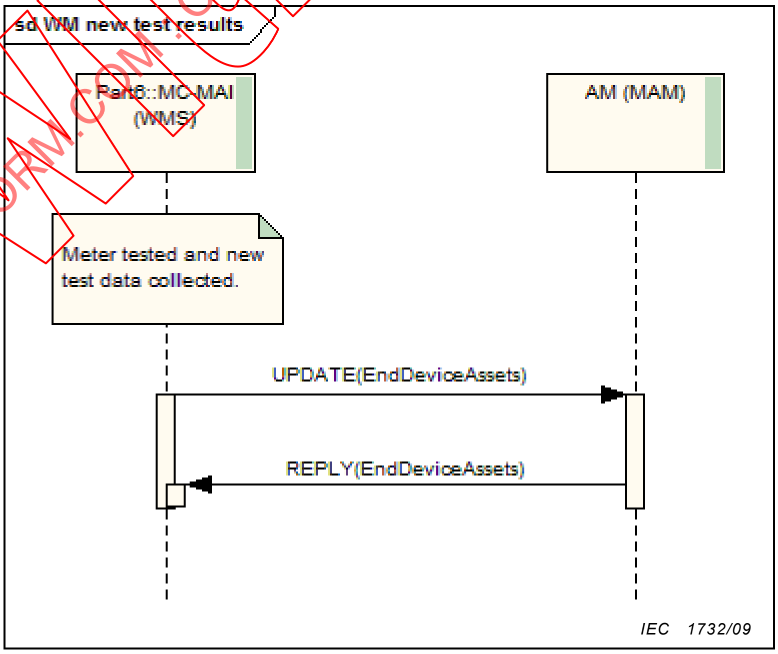


Figure 15 – Example of WM to MAM exchange – test results

5.3.2.4 End device removed from service

When an end device has reached the end of its useful life, it will be destroyed and the inventory adjusted to show that it is no longer an asset of the utility. This might happen when a meter (perhaps after failing a test) is deemed un-repairable, or when a load control unit is unresponsive and considered lost.

A utility might also sell a quantity of used meters in an effort to standardize around a particular brand or model. This too will remove the product from active inventory. Adjustments to inventory can be communicated using a method similar to that described in Figure 16.

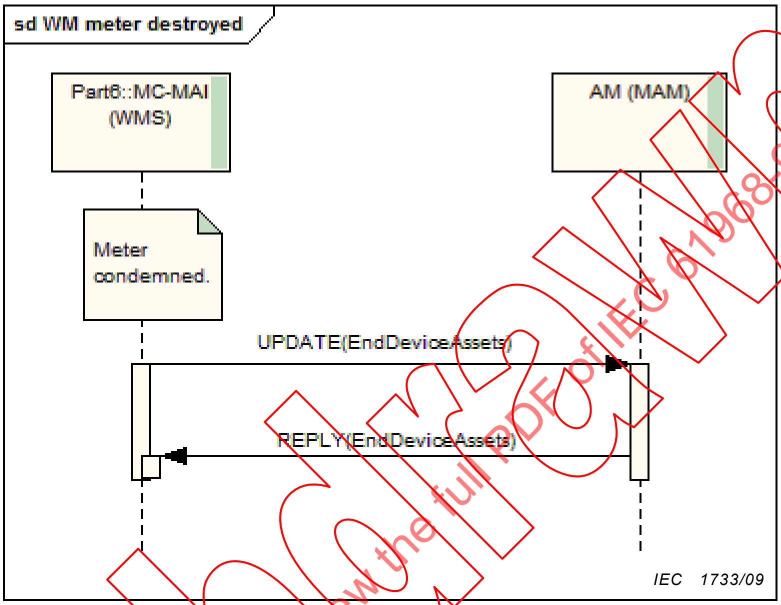


Figure 16 – Example of WM to MAM exchange – asset status update

5.3.2.5 Customer data synchronization

When changes are made to Customers, CustomerAccounts, CustomerAgreements, MeterAssets (EndDeviceAssets), ServiceDeliveryPoints or ServiceLocations, these changes will need to be published for consumption by systems such as OMS and MS. Often the Customer Billing system will be the master repository for this information.

Initially, a newly installed system may wish to immediately subscribe to changes in the data of interest (as shown by example of Figure 17.) The system however will require the history of the data as well, and may utilize “GET” and “REPLY” verbs to fill particular gaps, as shown by the example of Figure 18. Some MS may obtain customer data prepared for it by the MDM, while other MS may receive customer data directly from the CIS.

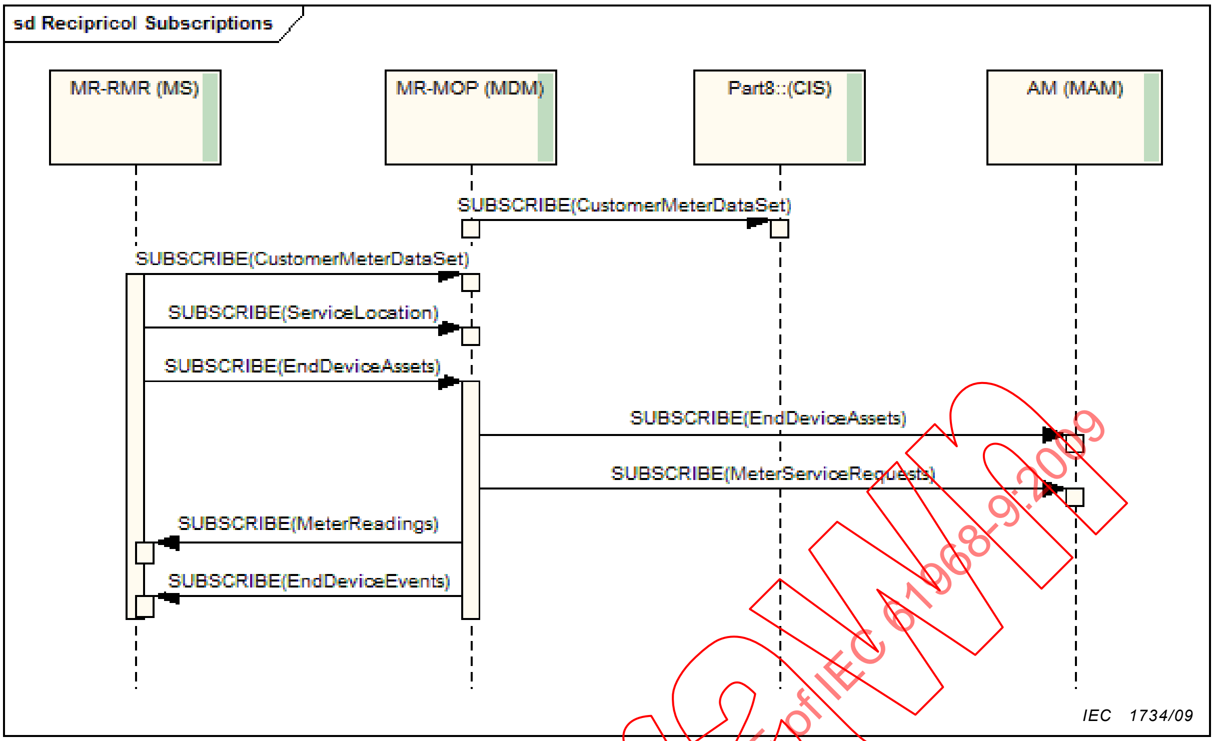

sd Reciprical Subscriptions
MR-RMR (MS)
MR-MOP (MDM)
Part8::(CIS)
AM (MAM)
SUBSCRIBE(CustomerMeterDataSet)
SUBSCRIBE(CustomerMeterDataSet)
SUBSCRIBE(ServiceLocation)
SUBSCRIBE(EndDeviceAssets)
SUBSCRIBE(EndDeviceAssets)
SUBSCRIBE(MeterServiceRequests)
SUBSCRIBE(MeterReadings)
SUBSCRIBE(EndDeviceEvents)
IEC 1734/09

Figure 17 – Example of reciprocal subscriptions (with MDM present)

Metering systems are typically configured to read meters by group or category. Some metering systems may define particular groups or categories of meters directly in the MS. Other MS may subscribe to a meter group or category definition from a MDM. Either case of meter group definition may result in the creation of associated measurements in the MS.

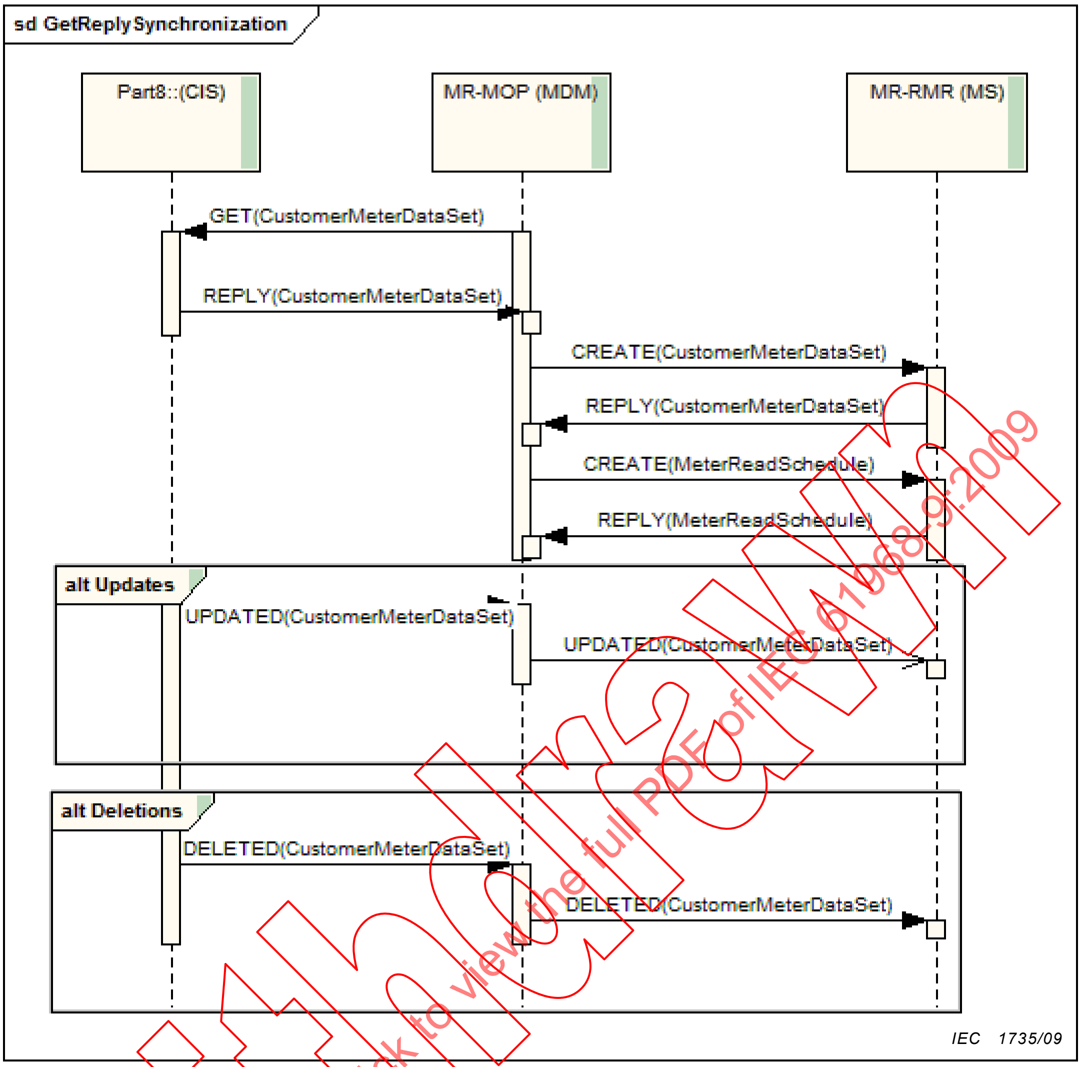

sd GetReplySynchronization
Part8::(CIS)
MR-MOP (MDM)
MR-RMR (MS)
GET(CustomerMeterDataSet)
REPLY(CustomerMeterDataSet)
CREATE(CustomerMeterDataSet)
REPLY(CustomerMeterDataSet)
CREATE(MeterReadSchedule)
REPLY(MeterReadSchedule)
alt Updates
UPDATED(CustomerMeterDataSet)
UPDATED(CustomerMeterDataSet)
alt Deletions
DELETED(CustomerMeterDataSet)
DELETED(CustomerMeterDataSet)
IEC 1735/09

Figure 18 – Example of synchronization exchanges

5.3.3 Message format

Messages are implemented using EndDeviceAssets and CustomerMeterDataSet messages. The message format for EndDeviceAssets is described in Figure 19 and the message format for CustmerMeterDataSet is described in Figure 20.

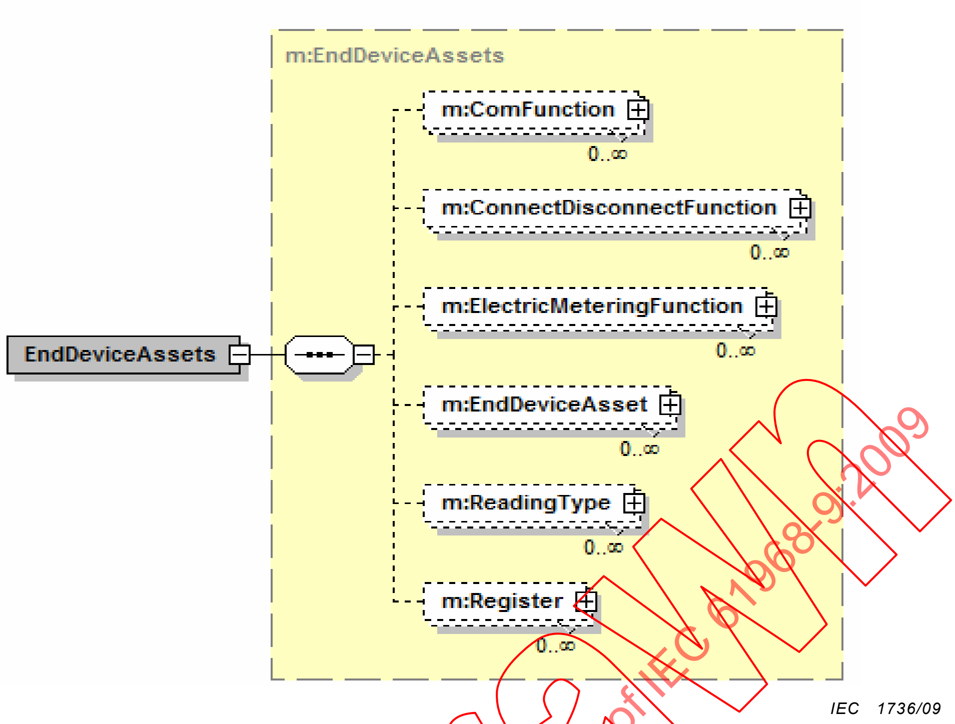


Figure 19 – End device asset message format

The detailed XML schema for an EndDeviceAssets message is described in Clause H.10.

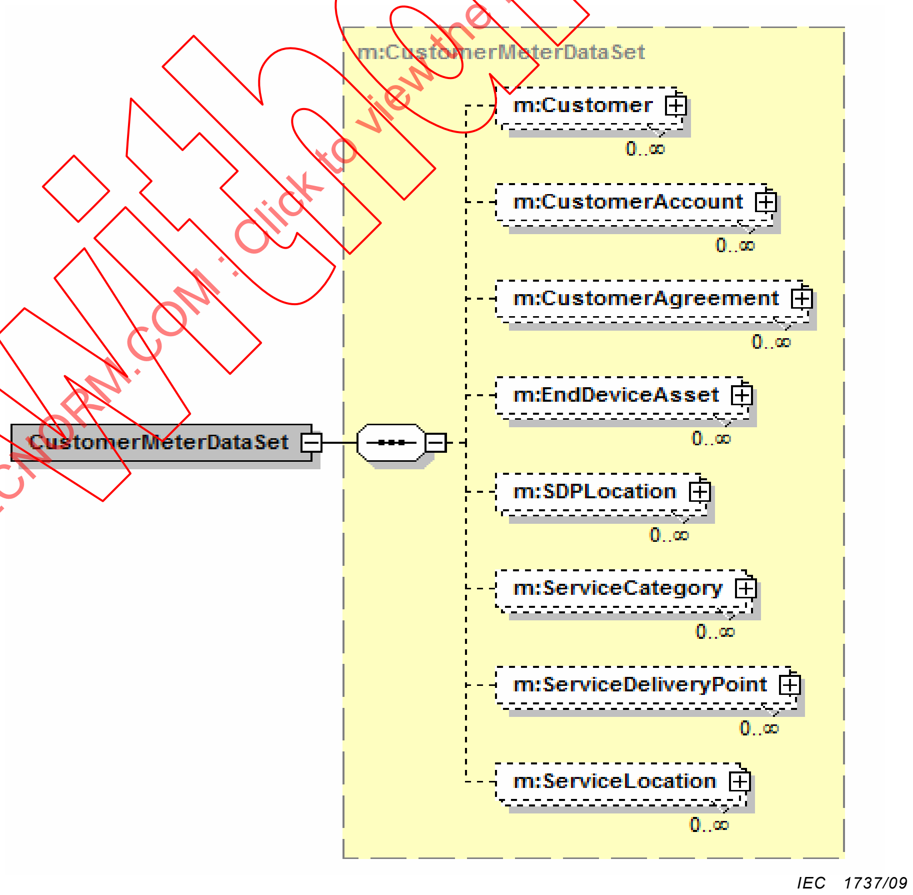


Figure 20 – Customer meter data set message format

The detailed XML schema for a CustomerMeterDataSet is described in Clause H.2.

5.4 Meter reading messages

5.4.1 General

Whenever a “measurement” is required from an end device, a MeterReading exchange is quite likely the appropriate tool for the job. The end device may have metrology capability, and if so, is most likely called a “meter.” But there are other potential applications for MeterReading. If a data consumer wishes to measure the position of the switch on a connect/disconnect switch, this can be done as a MeterReading exchange. If a data consumer wishes to measure the “energization” status of the end device, this can be done with a MeterReading exchange or alternatively, with an EndDeviceEvent exchange as described in 5.2.

5.4.2 Applications

5.4.2.1 Periodic meter reads

It is necessary to periodically gather meter readings from a MS for billing through the customer billing system. The request for meter reading should specify a meter or group of meters, a type of reading to collect, and a frequency and duration of interest. The scheduled frequency may consist of regular or irregular periods.

The MeterReadSchedule request may be initiated to the MS from any of the following.

- The CIS (in an effort to collect billing determinants).
- A planning and scheduling application (in an effort to acquire engineering data about the distribution network).
- An OMS (to establish a stream of status information).
- A meter data management system (in an effort to broker data for any or all of the above applications).
- The MS itself may also self-initiate a MeterReadSchedule.

An example for one such exchange (this one using a MDM) is shown in Figure 21.

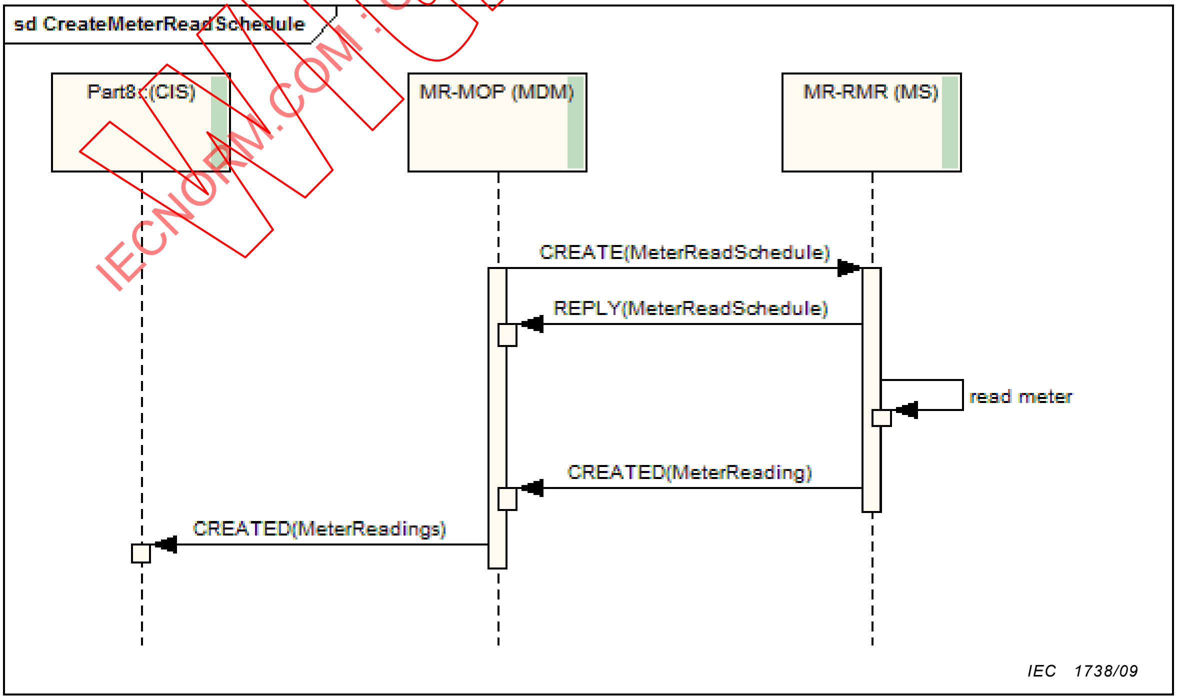
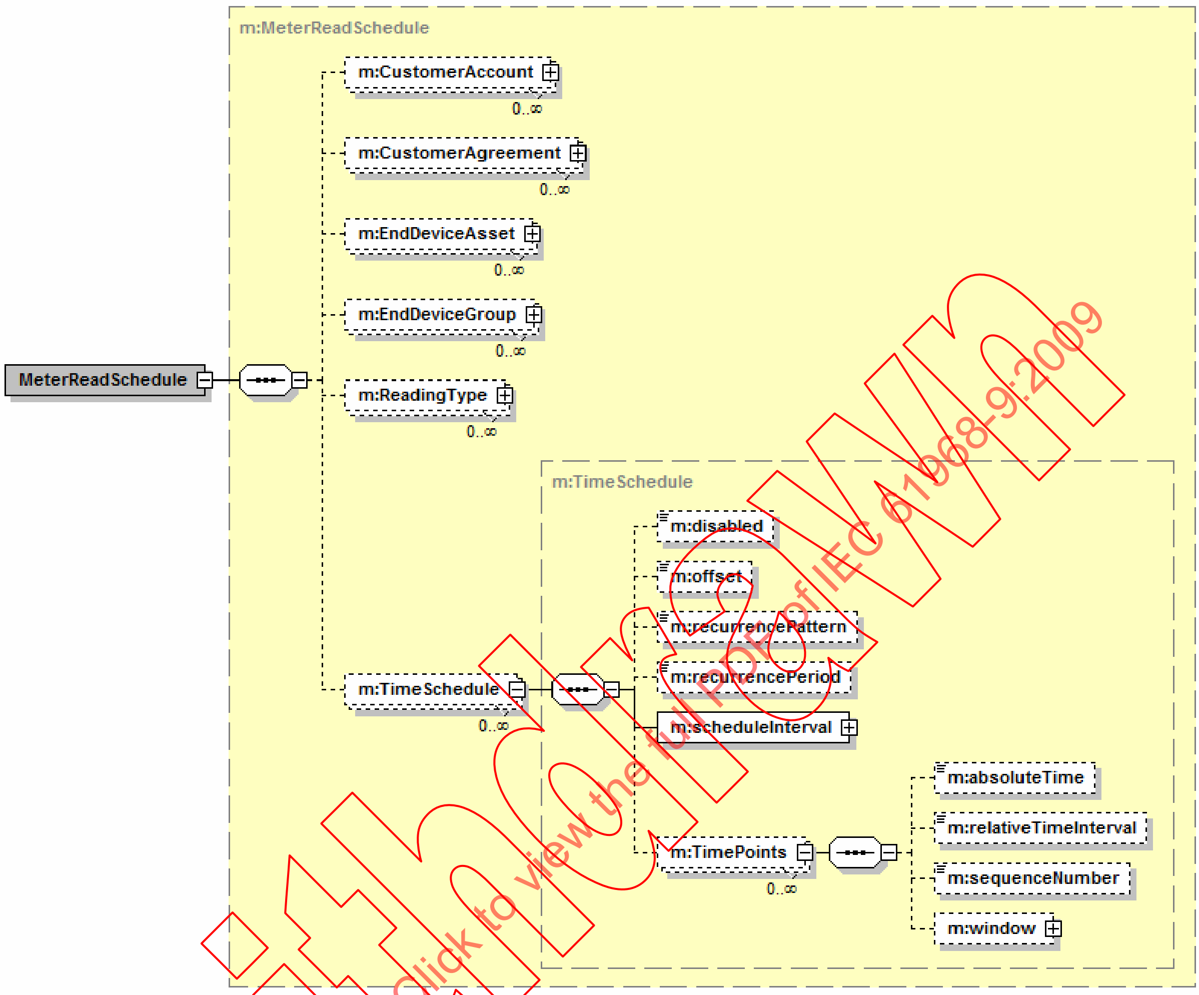


Figure 21 – Example use of meter read schedule to create subscription

Figure 22 shows the structure used for the payload of a MeterReadSchedule request message:



IEC 1739/09

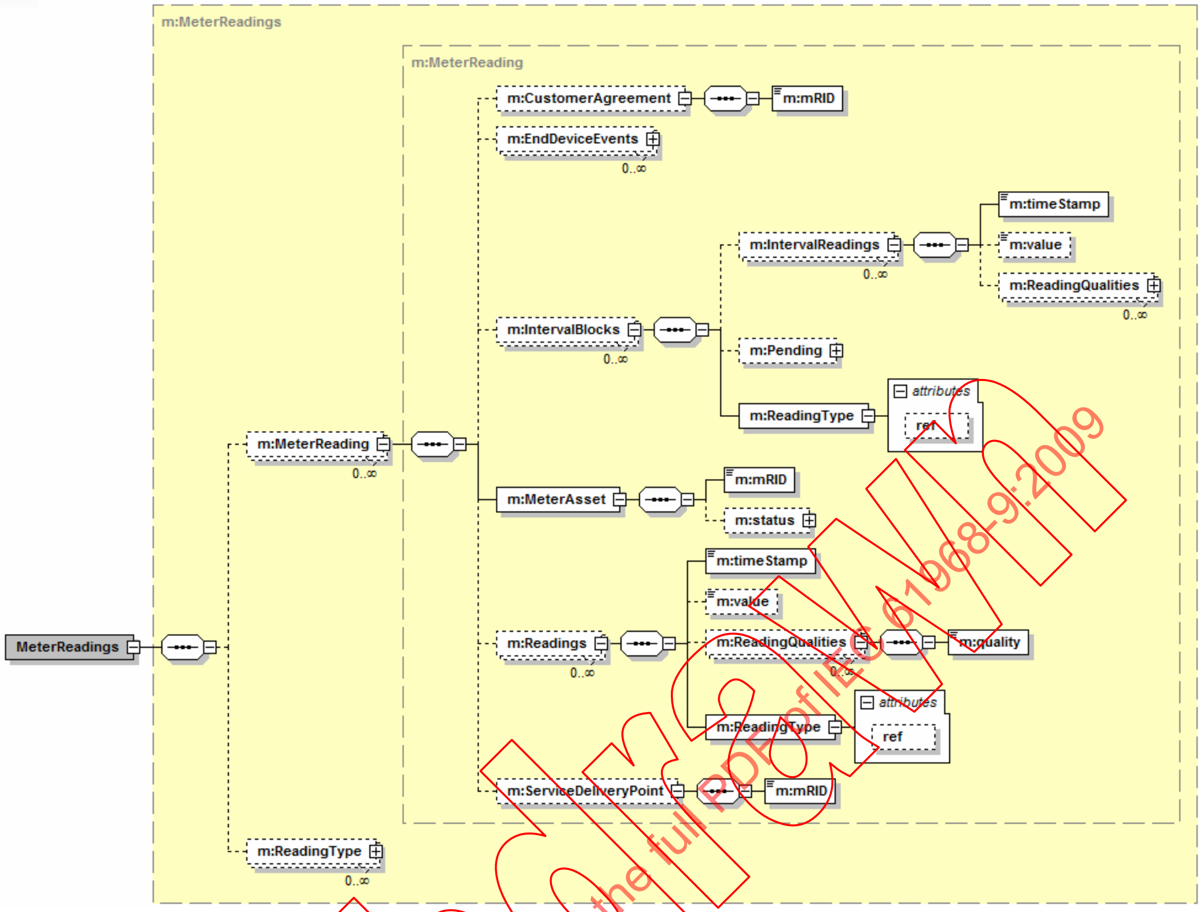
Figure 22 – Meter read schedule message format

Some metering systems may have the ability to decouple MeterReading collection from MeterReading reporting. For metering systems without this capability, readings are reported immediately upon collection. In the MeterReadSchedule request, MeterReadings may be requested using a variety of parameters, including:

- specific meter, using the EndDeviceAsset mRID;
- specific meter, using a serialNumber as known to the Metering System;
- EndDeviceGroups, where a EndDeviceGroup identifies a groupAddress used within the Metering System;
- by TimeSchedule;
- set of desired ReadingTypes.

The detailed MeterReadSchedule XML schema is provided in Clause H.6. Details related to the qualification of request messages are described in Annex J.

Figure 23 shows the message format used to present meter readings from one or more end devices.



IEC 1740/09

Figure 23 – Meter readings message format

The MeterReadings message structure allows for:

- readings from one or more meters;
- reading values each have an associated reading type, timestamp and value;
- many quality values can be associated with each reading value;
- readings can be supplied in the form of interval blocks, where readings of a common reading type are grouped together;
- event histories can be returned with meter readings.

Key to the reporting of a value from a meter is the reading type. The ReadingType class in the CIM is defined to allow capture of the following information as related to the description of a reading type:

- mRID (to uniquely identify a reading type);
- channel number;
- default quality;
- default value data type;
- dynamic configuration;
- default forward chronology;
- interval length (in seconds);
- measurement type category;
- unit of measure;

- register digits.

The details of this and other classes specific to metering are described in the metering package of the CIM. A more thorough discussion of reading types is provided in Annex C.

Readings are not limited to metering devices. Any EndDevice may potentially produce a measurement. The asset producing the reading is identified by the meterAsset class. It is expected that all assets have a unique mRID so that meters can be distinguished from remote connect/disconnect switches, and from other devices.

Interval data has unique timestamp requirements. The extended capabilities of ISO 8601 provide the means to specify both a formal interval length (as an ISO 8601 “period” of time), as well as a fixed reference point.

The following XML provides an XML example payload of a MeterReadings message.

```
<?xml version="1.0" encoding="UTF-8"?>
<m:MeterReadings xsi:schemaLocation="http://iec.ch/TC57/2009/MeterReadings# MeterReadings.xsd"
xmlns:m="http://iec.ch/TC57/2007/MeterReadings#" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <m:MeterReading>
                <m:CustomerAgreement>
                        <m:mRID>35437383</m:mRID>
                </m:CustomerAgreement>
                <m:EndDeviceEvents>
                        <m:mRID>47474</m:mRID>
                        <m:category>6.6.1.47.1</m:category>
                        <m:createdDateTime>2009-12-17T09:37:53.0Z</m:createdDateTime>
                </m:EndDeviceEvents>
                <m:IntervalBlocks>
                        <m:IntervalReadings>
                                <m:timeStamp>2009-12-17T09:30:47.0Z</m:timeStamp>
                                <m:value>25.44</m:value>
                        </m:IntervalReadings>
                        <m:IntervalReadings>
                                <m:timeStamp>2009-12-17T09:45:47.0Z</m:timeStamp>
                                <m:value>37.57</m:value>
                        </m:IntervalReadings>
                        <m:IntervalReadings>
                                <m:timeStamp>2009-12-17T10:00:47.0Z</m:timeStamp>
                                <m:value>23.41</m:value>
                        <m:ReadingType ref="2.6.7.1.0.12.0.0.0.3.72"/>
                </m:IntervalBlocks>
                <m:MeterAsset>
                        <m:mRID>6468822</m:mRID>
                        <m:status>
                                <m:value>OK</m:value>
                        </m:status>
                </m:MeterAsset>
                <m:Readings>
                        <m:timeStamp>2009-12-17T09:30:47.0Z</m:timeStamp>
                        <m:value>3.14159E0</m:value>
                        <m:ReadingType ref="0.0.1.0.2.13.2.0.0.0.111"/>
                </m:Readings>
                <m:Readings>
                        <m:timeStamp>2009-12-17T09:30:47.0Z</m:timeStamp>
                        <m:value>83837733</m:value>
                        <m:ReadingQualities>
                                <m:quality>2.5.256</m:quality>
                        </m:ReadingQualities>
                        <m:ReadingType ref="0.0.1.20.0.12.0.0.0.3.72"/>
                </m:Readings>
                <m:ServiceDeliveryPoint>
                        <m:mRID>847647647</m:mRID>
                </m:ServiceDeliveryPoint>
        </m:MeterReading>
        <m:ReadingType>
                <m:mRID>2.6.7.1.0.12.0.0.0.3.72</m:mRID>
                <m:aliasName>15-minute Incremental IntervalData Forward Energy  (kWh)</m:aliasName>
                <m:unit>kWh</m:unit>
        </m:ReadingType>
        <m:ReadingType>
                <m:mRID>0.0.1.0.2.13.2.0.0.0.111</m:mRID>
                <m:aliasName>BulkQuantity LoadInterrupt n0 (Count)</m:aliasName>
                <m:unit>count</m:unit>
        </m:ReadingType>
        <m:ReadingType>
                <m:mRID>0.0.1.20.0.12.0.0.0.3.72</m:mRID>
                <m:aliasName>BulkQuantity Total Energy (kWh)</m:aliasName>
                <m:unit>kWh</m:unit>
        </m:ReadingType>
</m:MeterReadings>
```

5.4.2.2 Manual meter reading

Meter readings can be obtained manually by a meter reader. Data collected may be maintained by the meter data manager. Meters may collect a number of different measurement types. Some types of meters may measure more than one phase, or may collect values for non-electrical measurements, such as water or gas.

A meter reader may input data shown on the panel of meter into a handheld device, which could be regarded as a metering system. Data input occurs several hundred times per day. A meter reader may present an account of the read to the customer. Note that this account is not an invoice. Billing would be generated normally by the CIS even in the manual reading case. Figure 24 shows such an exchange.

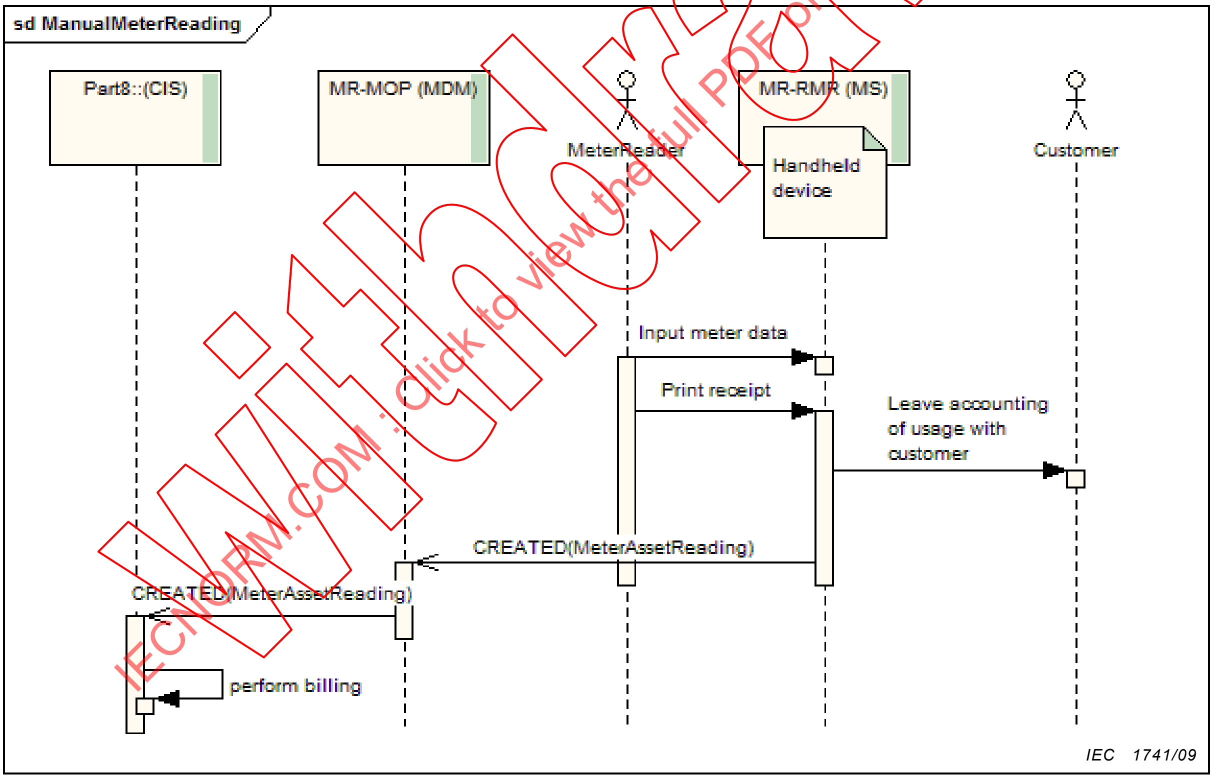


Figure 24 – Example manual meter reading exchange

At the end of the working day, all data stored in the handheld device is sent to the MDM via a communication network. The MDM publishes a message to the CIS informing of the creation of MeterAssetReading.

Schedules for walking a manual meter reading route are decided well in advance of performing the work.

5.4.2.3 On-request meter read

Meter read requests are issued to a MS in order to obtain meter readings on a per request basis. The MS will send a request to the desired meters. These can be used for billing inquiries, outage extent verification, and verification of restoration.

Many utilities have a policy of routing all revenue readings through the MDM so that all data will receive the same level of validation. However, not all MDM systems are able to offer validation services for outage data. For those that do, the utility must weigh the value of MDM validation over the time delay it introduces, as well as the ability of the particular OMS solution to reject incongruent data. For this reason, the example diagram shows revenue readings routed through the MDM, but outage data routed around it.

It is important to note that not all metering systems support 'on request' readings. For those that do, the implementation can also vary significantly.

On-request reads may be initiated to the MS from any of the following.

- The CIS (in an effort to collect billing determinants).
- A planning and scheduling application (in an effort to acquire engineering data about the distribution network).
- An OMS (in order to verify if a customer is affected by an outage or has been restored).
- A meter data management system (in an effort to broker data for any or all of the above applications).
- The MS itself may also directly initiate a meter read.

An example for one such exchange (this one using a MDM) is shown in Figure 25.

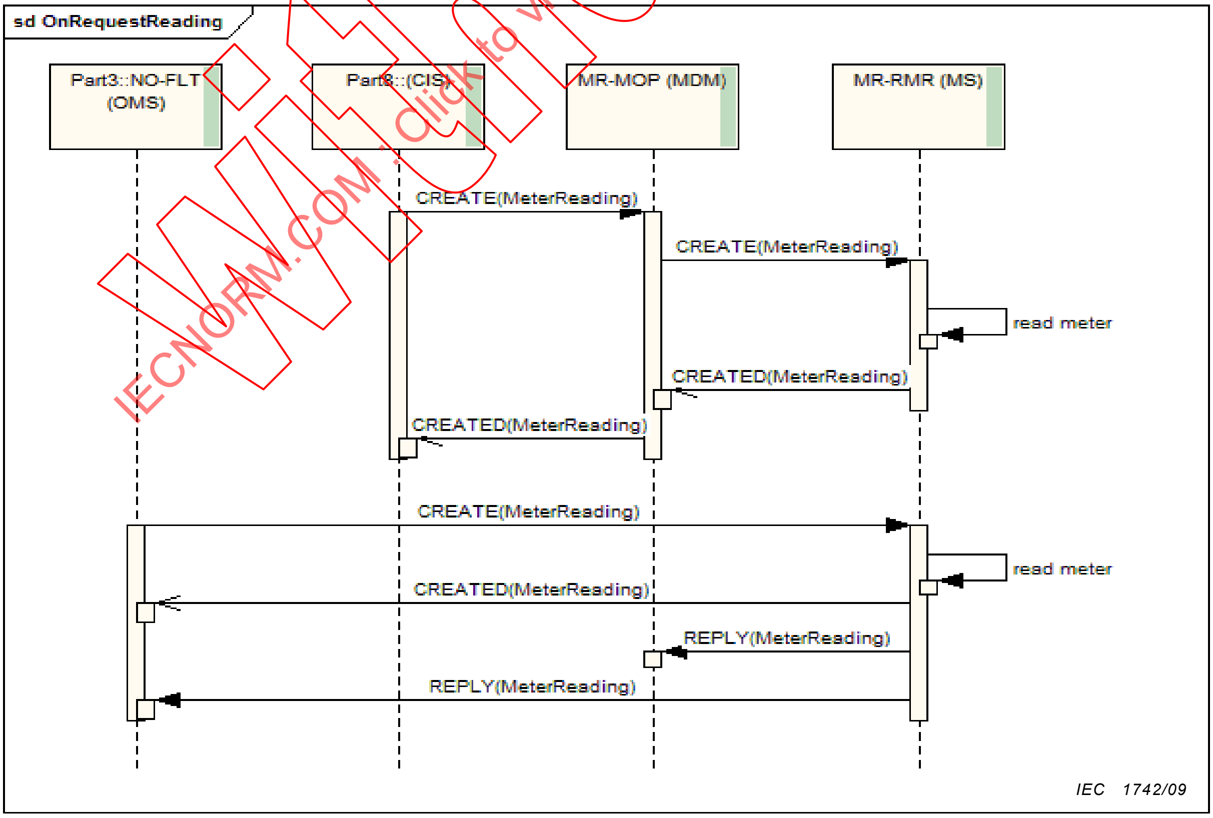


Figure 25 – Example on-request meter read exchange

5.4.2.4 Historical meter data access

A distribution network planner may use historical meter reading values as load information for capacity planning purposes (see the request/reply example in Figure 26). This would permit usage to be aggregated to determine loads for a transformer or feeder.

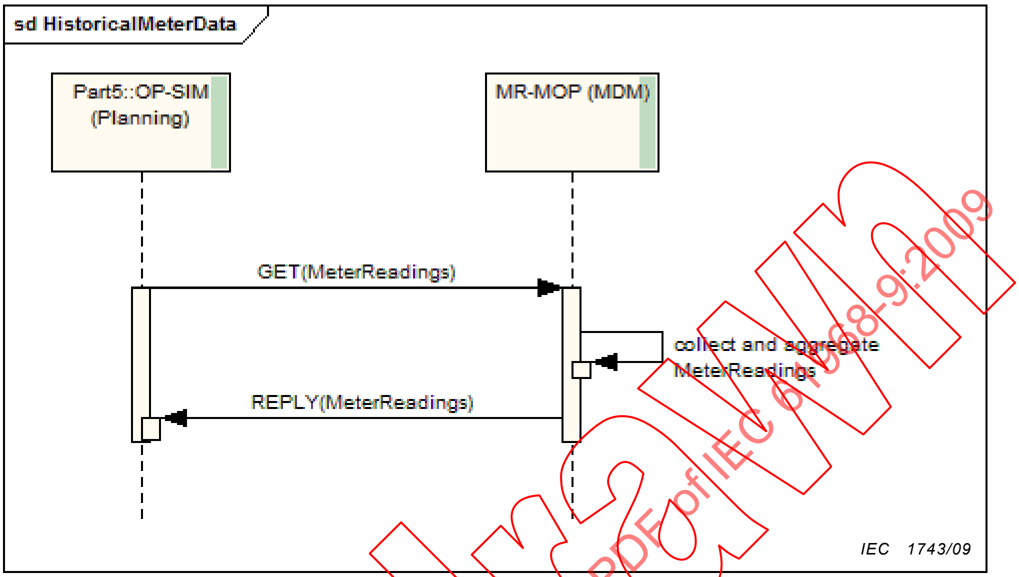


Figure 26 – HistoricalMeterData exchange

It is important to note the use of request parameters to qualify requests for meter data, filtering the results to obtain data for specific meters within specific timeframes.

5.4.2.5 Billing inquiry

A customer or an internal source may identify a customer billing issue. A meter read request in combination with historical meter reads may be used to resolve the billing issue. (The top of Figure 27 shows an inquiry being satisfied by data which recently arrived, while later on in the day an inquiry is made which requires a fresh reading from the meter.)

In some cases, the desired data may be accessible from the MDM. In other cases, it may be necessary to issue a read request remotely through the MS, or manually through a meter service request.

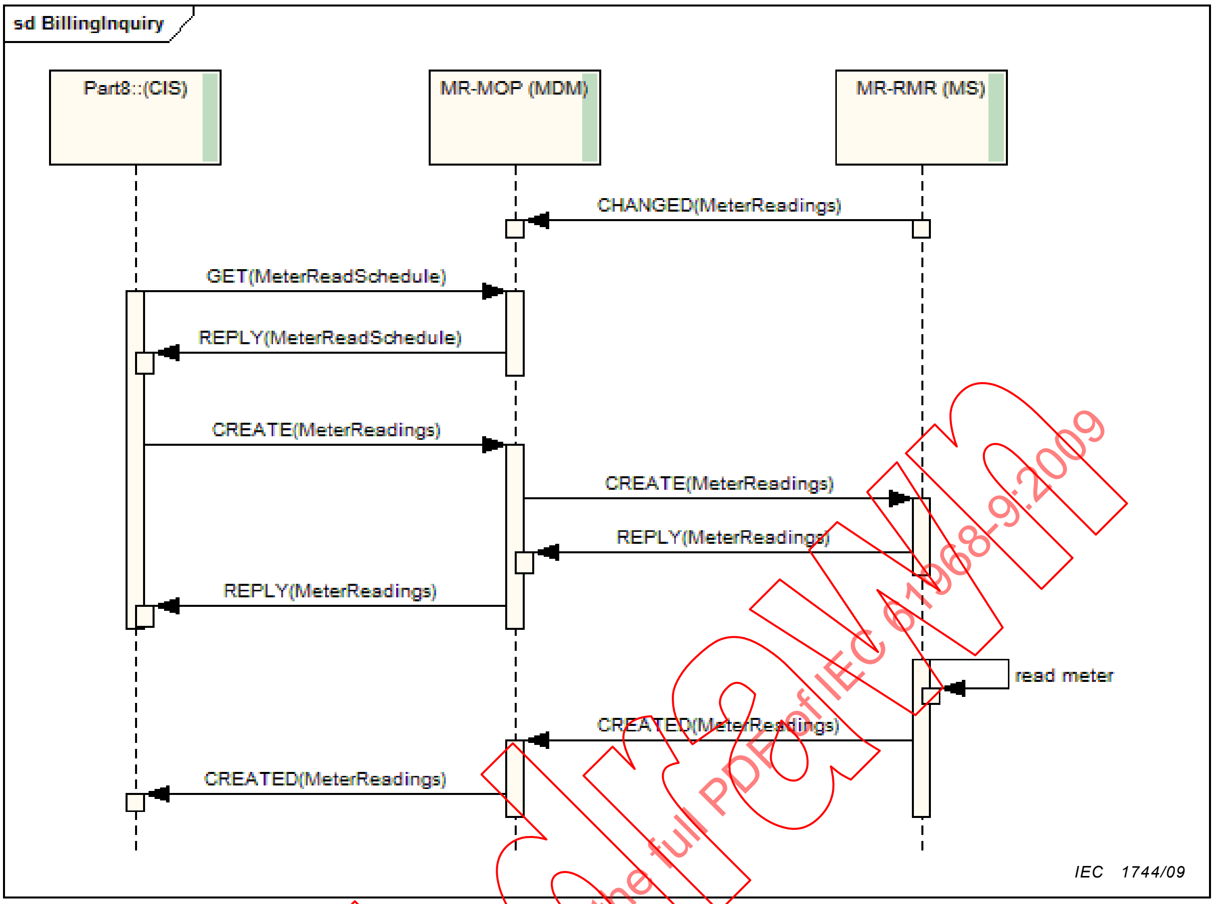
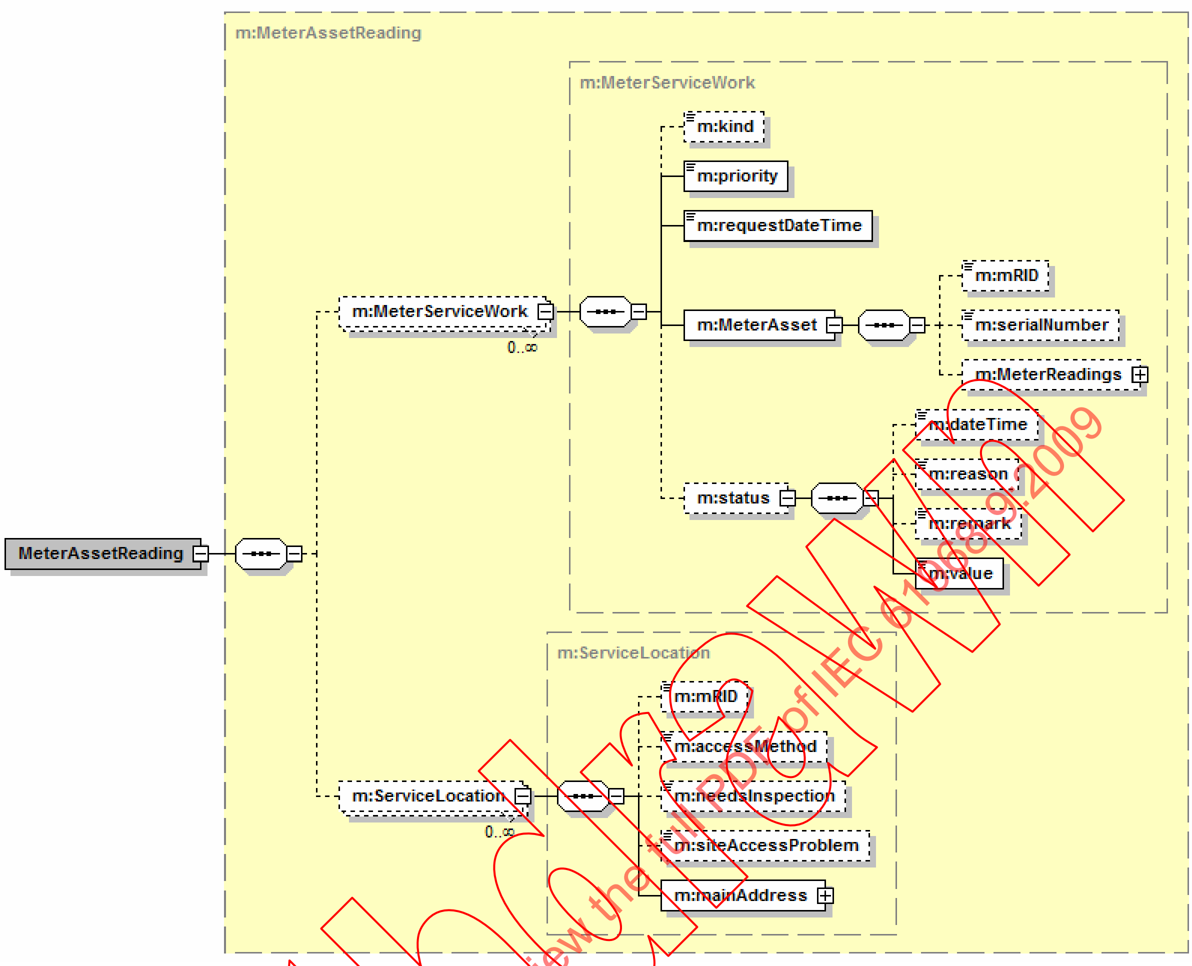


Figure 27 – Example billing inquiry message exchange

5.4.3 Message format

The message format used to convey a meter asset reading is described in Figure 28.



IEC 1745/09

Figure 28 – Meter asset reading message format

A meter asset reading message may contain many MeterServiceWork items. Each MeterServiceWork item identifies a MeterAsset and Customer. There may be many Readings and EndDeviceEvents obtained from the MeterAsset. The detailed XML schema for a meter asset reading is described in Clause H.3.

5.5 End device control messages

5.5.1 General

There are many types of end device control messages. These are used to send instructions to one or more end devices.

5.5.2 Applications

5.5.2.1 Load control

Load control (a.k.a. direct load control) requests can often be made to a MS for the purpose of load curtailment. This request would typically be initiated from network operations. Not all MS will have load control capabilities.

It should also be noted that this is different from a disconnect, where a disconnect results in the complete loss of power to a single customer.

A load control will typically result in the shedding of specifically configured loads (e.g. air conditioning, pumps, etc.)

The load control function executes load shed commands which are computed by load management software (contained in the network operations block). The load management software is used to compute an appropriate load-shed amount, and could potentially include many factors including the predicted duration of generation shortfalls, historical usage patterns, real-time usage data, and weather.

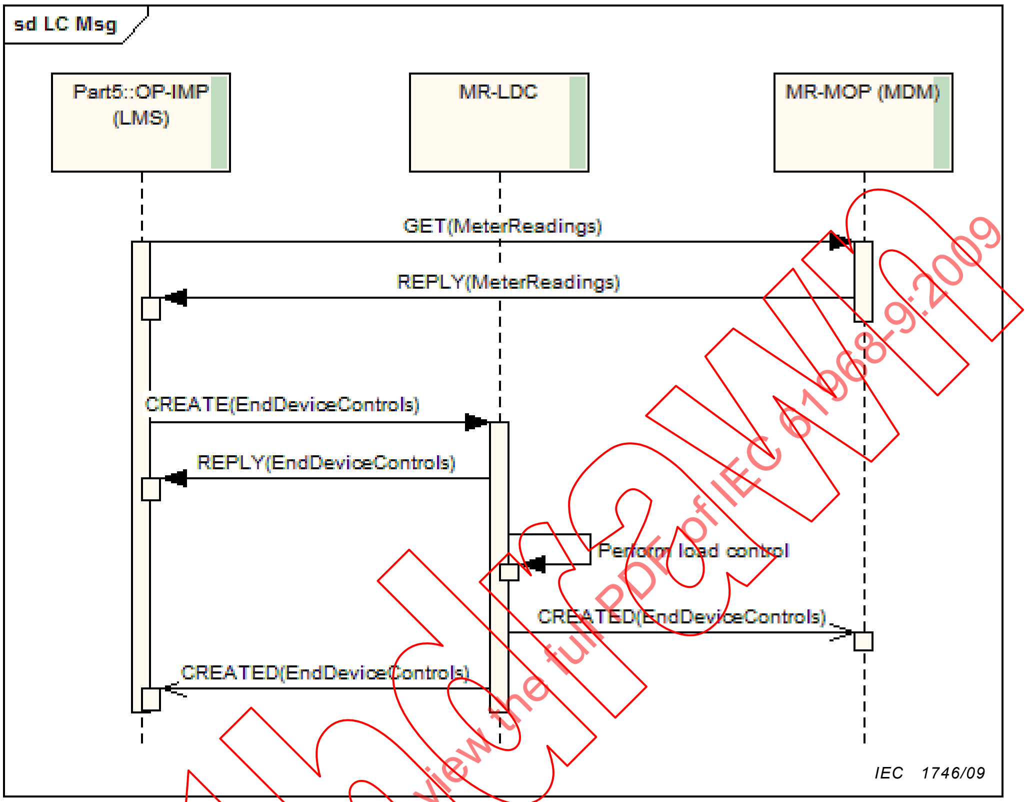


Figure 29 – Example load control message exchange

The example in Figure 29 shows how load history may be obtained from the MDM in the form of MeterReadings and used to perform load analysis. It also shows load control commands being issued by the LMS to the load control system. Deployments that don't support a MDM would interact directly with the MS instead.

Load control commands are implemented as a type of EndDeviceControl, where the command can be addressed by EndDeviceGroup (using a group address), by EndDeviceAsset (using mRID) or by CustomerAgreement. This is described below in Figure 30.

5.5.2.2 Load control device installation

The installation of load control units is usually more labour intensive than installing a meter, but the data exchange requirements are simpler. LC units are different from meters in the sense that they don't require periodic recalibration. Once an LC unit is installed, it will probably remain in place for the life of the unit, even if the customer withdraws from the LC program, or if there is a change in tenancy.

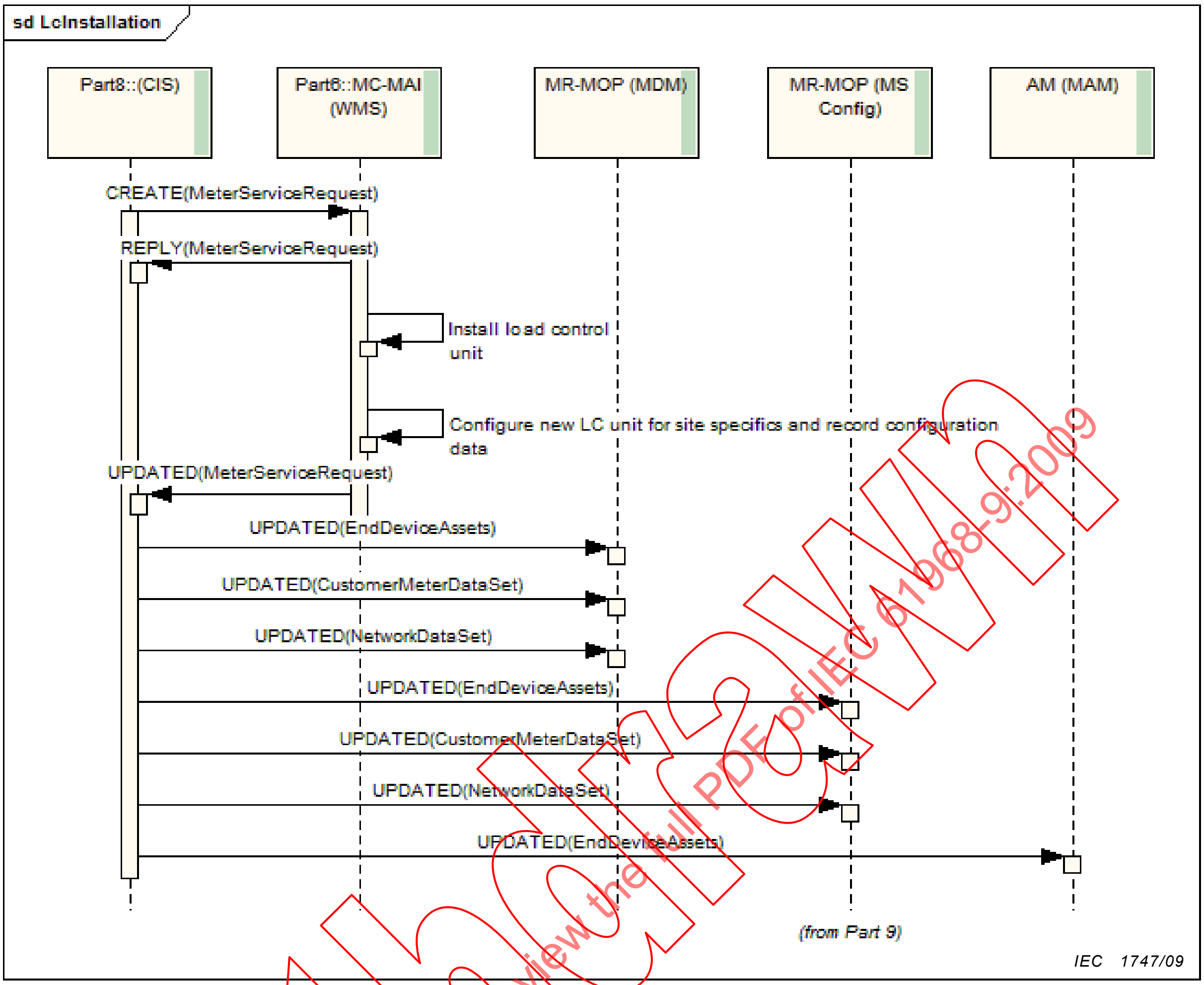

sd LcInstallation
Part8::(CIS)
Part6::MC-MAI (WMS)
MR-MOP (MDM)
MR-MOP (MS Config)
AM (MAM)
CREATE(MeterServiceRequest)
REPLY(MeterServiceRequest)
Install load control unit
Configure new LC unit for site specifics and record configuration data
UPDATED(MeterServiceRequest)
UPDATED(EndDeviceAssets)
UPDATED(CustomerMeterDataSet)
UPDATED(NetworkDataSet)
UPDATED(EndDeviceAssets)
UPDATED(CustomerMeterDataSet)
UPDATED(NetworkDataSet)
UPDATED(EndDeviceAssets)
(from Part 9)
IEC 1747/09

Figure 30 – Example message exchange for LC unit installation

Before travelling to the service location, the installer will know the customer account data, the type of load control unit to be installed, and the ratings of the device to be controlled. The installation data can confirm device ratings, and where appropriate, document the port number of the LC unit wired to the device. The results of the installation can be published to all stakeholders as depicted in Figure 30.

5.5.2.3 Change of customer program

Particular metering solutions are usually chosen to fulfil specific metering needs as a result of customer program enrolment. In many cases, a flexible metering solution can provide coverage for a range of customer programs. When a customer transitions from enrolment in one program to another, it may simply require a change to the meter (or communication module) configuration. In extreme cases, it may require a meter change out. In other cases, it may require no change at all. Changes to a configuration will need to be communicated amongst the stakeholders. An example exchange in which the deployment involves a MDM is described in Figure 31. Another example in which the deployment lacks a MDM is described in Figure 32.

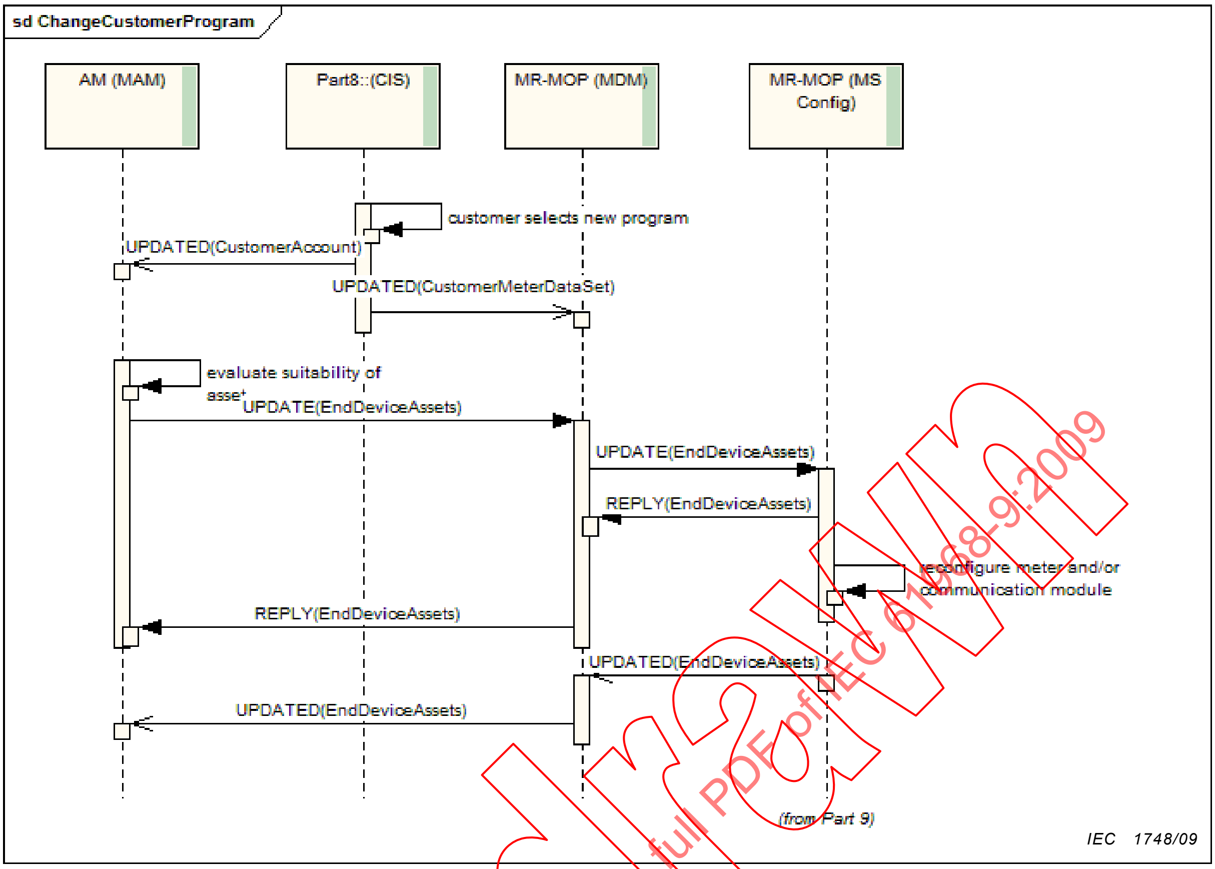


Figure 31 – Example message exchange for change of customer program (meter reconfigured)

Changes to the meter configuration may be expressed as changes to the configuration of an EndDeviceAsset.

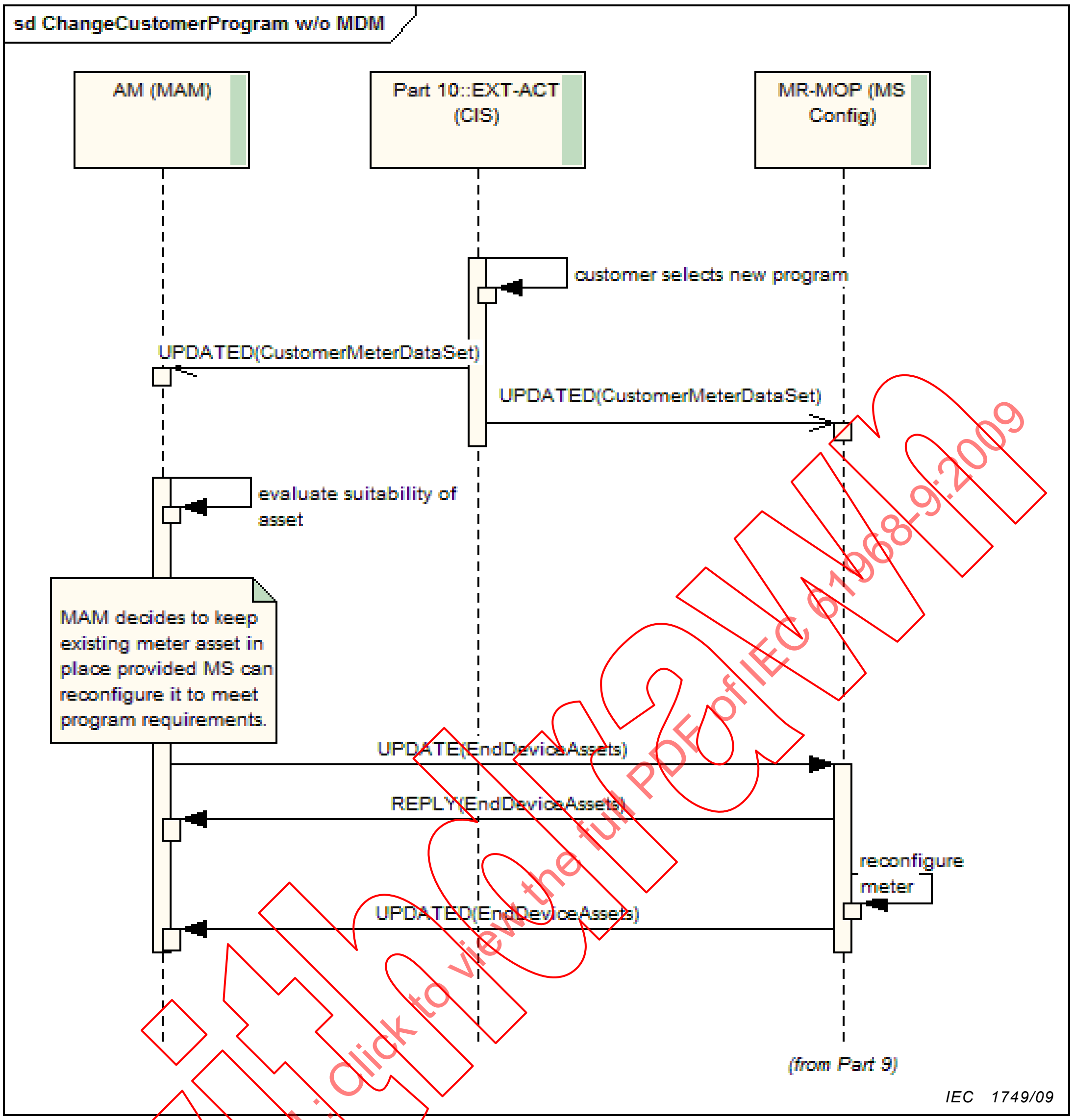

sd ChangeCustomerProgram w/o MDM
AM (MAM)
Part 10::EXT-ACT (CIS)
MR-MOP (MS Config)
customer selects new program
UPDATED(CustomerMeterDataSet)
UPDATED(CustomerMeterDataSet)
evaluate suitability of asset
MAM decides to keep existing meter asset in place provided MS can reconfigure it to meet program requirements.
UPDATE(EndDeviceAssets)
REPLY(EndDeviceAssets)
reconfigure meter
UPDATED(EndDeviceAssets)
(from Part 9)
IEC 1749/09

Figure 32 – Example message exchange for meter reconfiguration due to change of customer program where there is no MDM

In other cases, a change to the customer program might not be accommodated with a meter reconfiguration. The program change may require a meter changeout. (Please refer to 5.6 for more information on meter changeouts.) Meter changeouts entail a different workflow than meter reconfigurations. This will be reflected in the message exchange between systems. An example exchange is depicted in Figure 33.

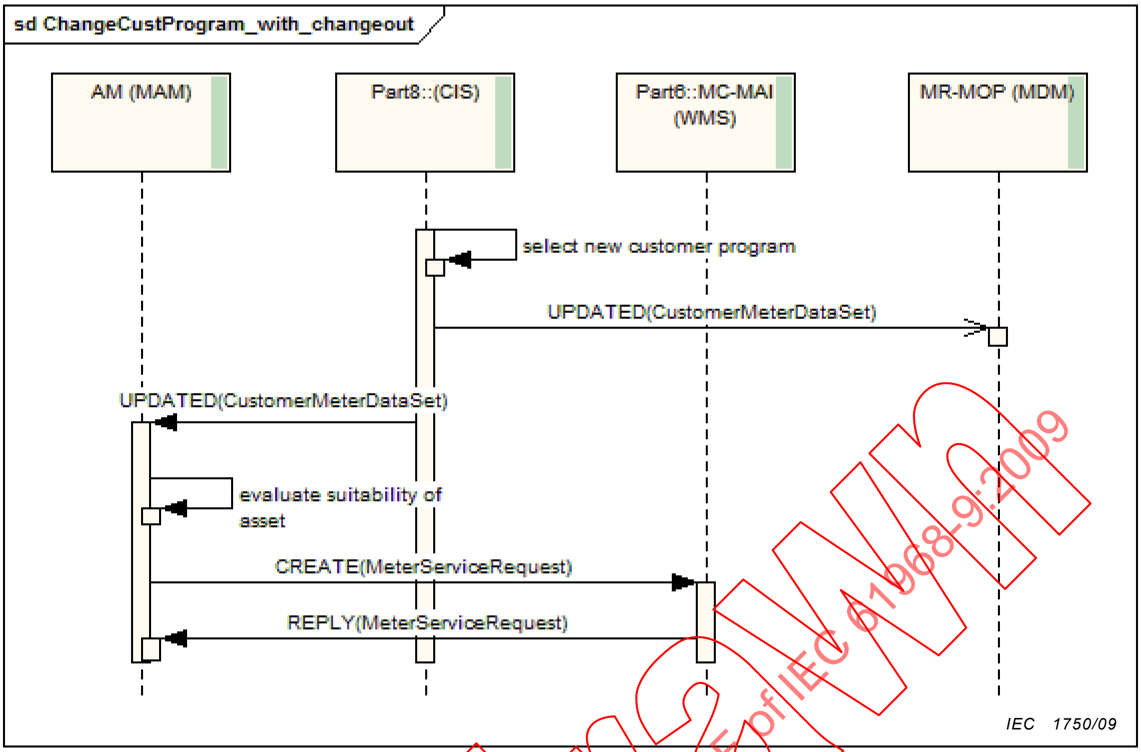

sd ChangeCustProgram_with_changeout
AM (MAM)
Part8::(CIS)
Part6::MC-MAI (WMS)
MR-MOP (MDM)
select new customer program
UPDATED(CustomerMeterDataSet)
UPDATED(CustomerMeterDataSet)
evaluate suitability of asset
CREATE(MeterServiceRequest)
REPLY(MeterServiceRequest)
IEC 1750/09

Figure 33 – Example message exchange for change of customer program with meter changed out

See Subclause 5.6.2.1 on meter installation and removal for additional details on the MeterServiceRequest message.

5.5.2.4 Meter disconnect and reconnect

For a variety of reasons, such as non-payment, it may be necessary to disconnect or reconnect a customer. When disconnected, recorded usage should be zero and out of power complaints should be ignored. When it is not possible to perform a disconnect or reconnect remotely through a MS, a meter service request will typically be issued to perform the disconnect or reconnect manually. Example message exchanges for remote operation are depicted below in Figure 34 and Figure 35.

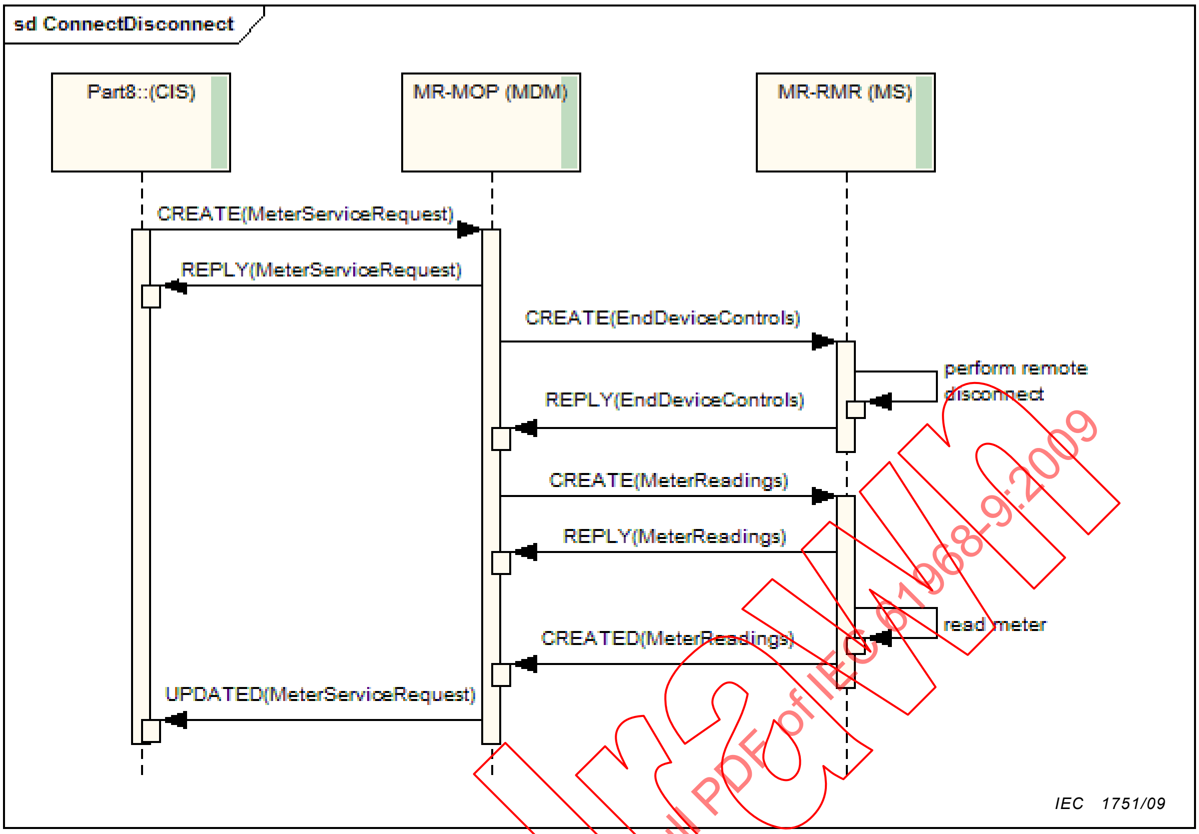


Figure 34 – Example message exchange for meter connect/disconnect

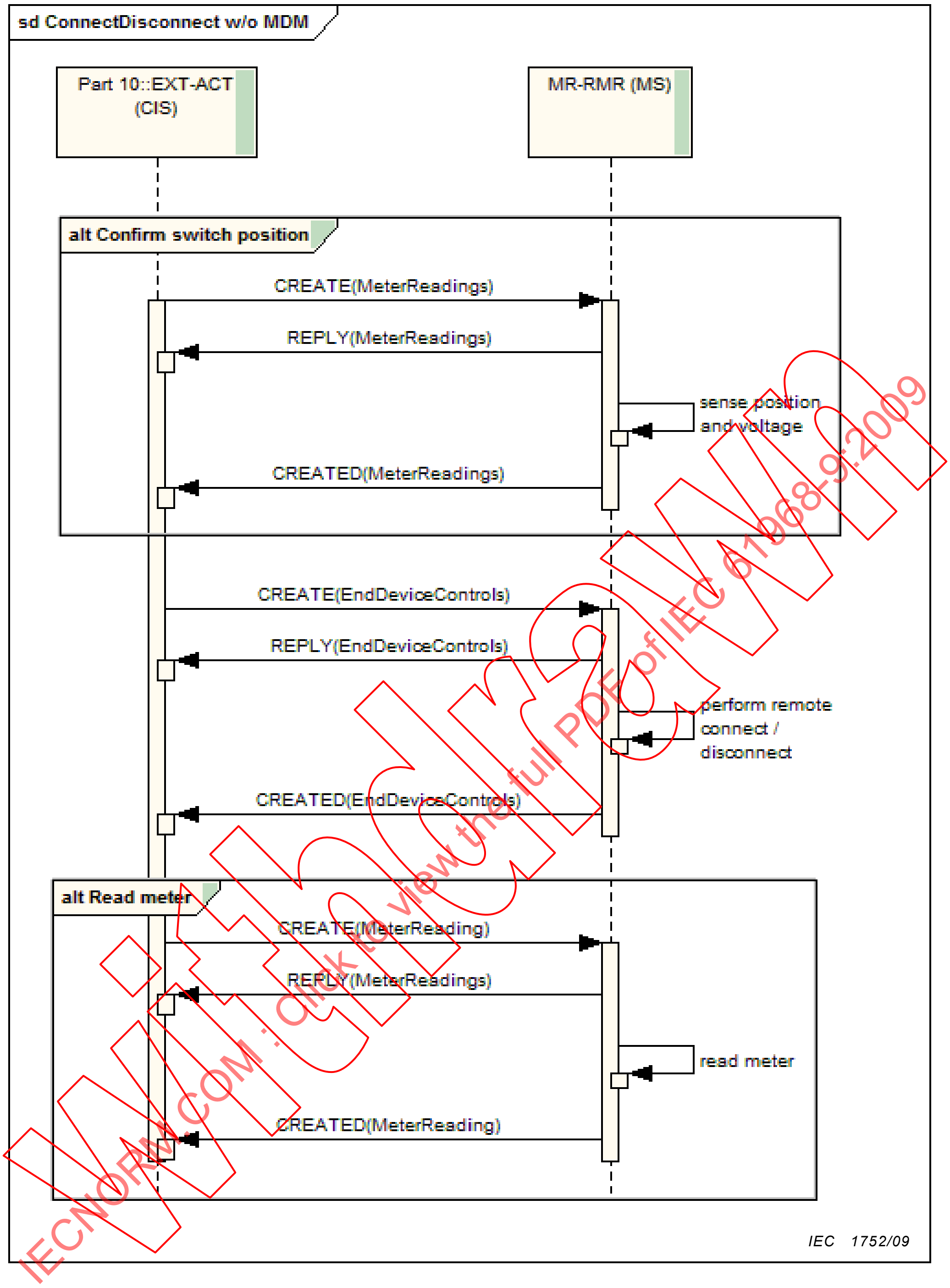


Figure 35 – Example message exchange for remote connect/disconnect operation directly between a CIS and MS

5.5.2.5 Real-time pricing

Real-time pricing signals and/or schedules can be sent to a meter via the MS. There are several ways this can be accomplished, such as:

- price signal issued in real time identifying a price for a given time interval;
- time of use (TOU) schedules published, which cause changes in the accumulation for each TOU tier;
- energy price schedules published in advance.

Often the EndDeviceGroup can be used to differentiate meters with different contracts or tariffs.

The example in Figure 36 shows a price signal being sent from network operations to the MS. The MS then acts as a network service provider to communicate the price in real time to meters and other equipment.

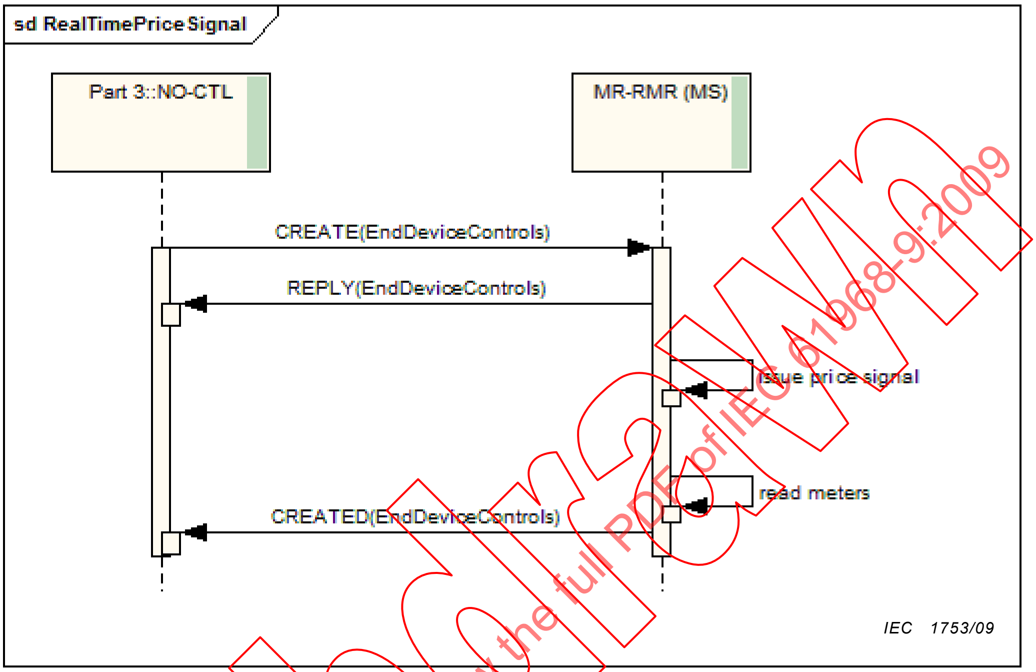
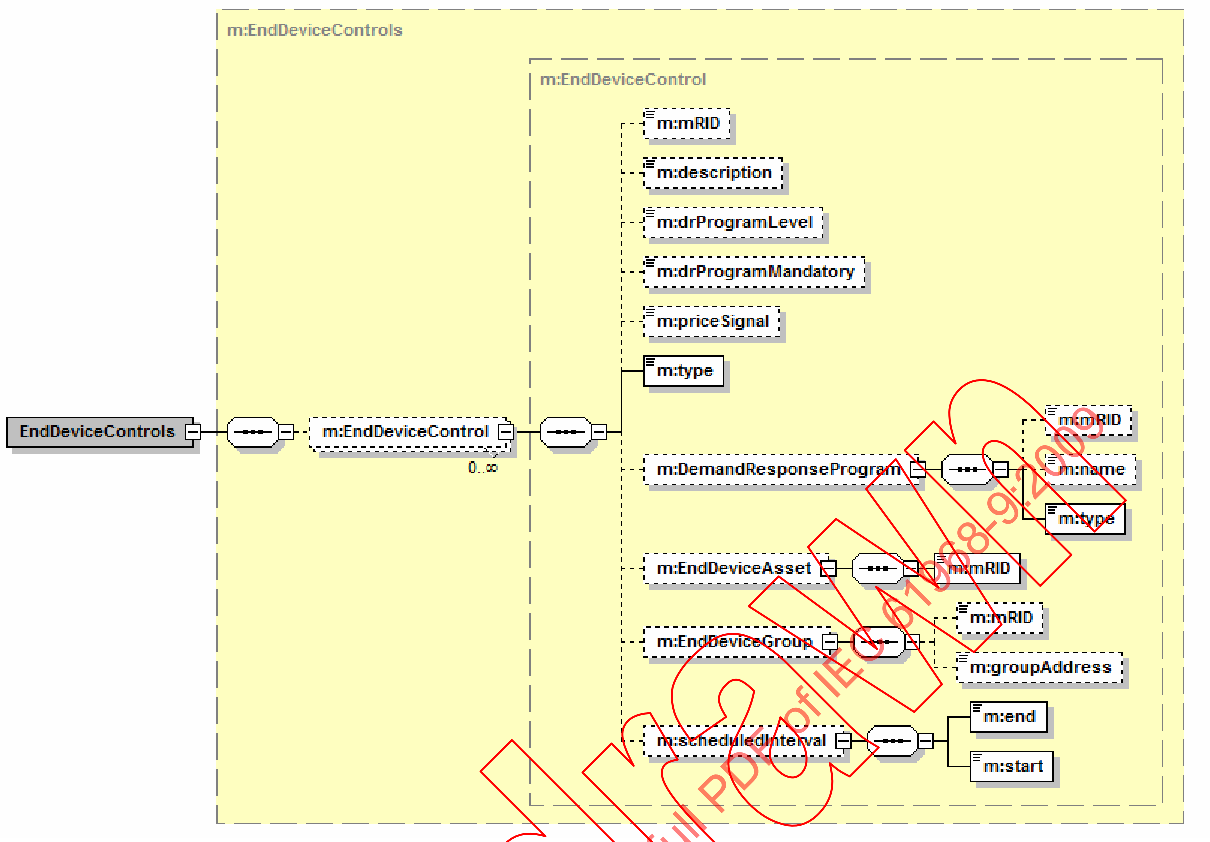


Figure 36 – Example message exchange for real-time price signal

Price signals are implemented as a subtype of EndDeviceControl, where the price is a message parameter.

5.5.3 Message format

Figure 37 describes the structure of an EndDeviceControls message.



IEC 1754/09

Figure 37 – End device controls message format

Within this message payload structure, specific meters may be addressed by CustomerAgreement, EndDeviceAsset (e.g. meter) and/or EndDeviceGroup. Aside from at least one address, the only required element is the typeControl. The typeControl element identifies the type of control to be performed. Examples of the different control types that could be used include, but are not limited to:

- load control signal;
- price signal or schedule;
- remote disconnect;
- remote reconnect;
- demand reset;
- demand reduction signal.

The XML schema for this structure is defined in Clause H.4.

5.6 Meter service requests

5.6.1 General

Meter service requests are a subclass of Work, where an EndDeviceAsset is involved.

5.6.2 Applications

5.6.2.1 Meter installation and removal

It may be necessary to install, remove or configure meters as a consequence of the registration of a new customer, removal of a customer or the switch of a customer from one supplier to another. There may also be the need to change out a meter which involves the removal of the old meter, installation of the new meter and configuration of the new meter as needed by the metering system.

Prior to meter installation, subscriptions have been established between the MS and the MDM to receive updates to customer data (see the section on data synchronization for more details). The workflow for a meter change out is shown in Figure 38. This encompasses both the meter installation and meter removal processes.

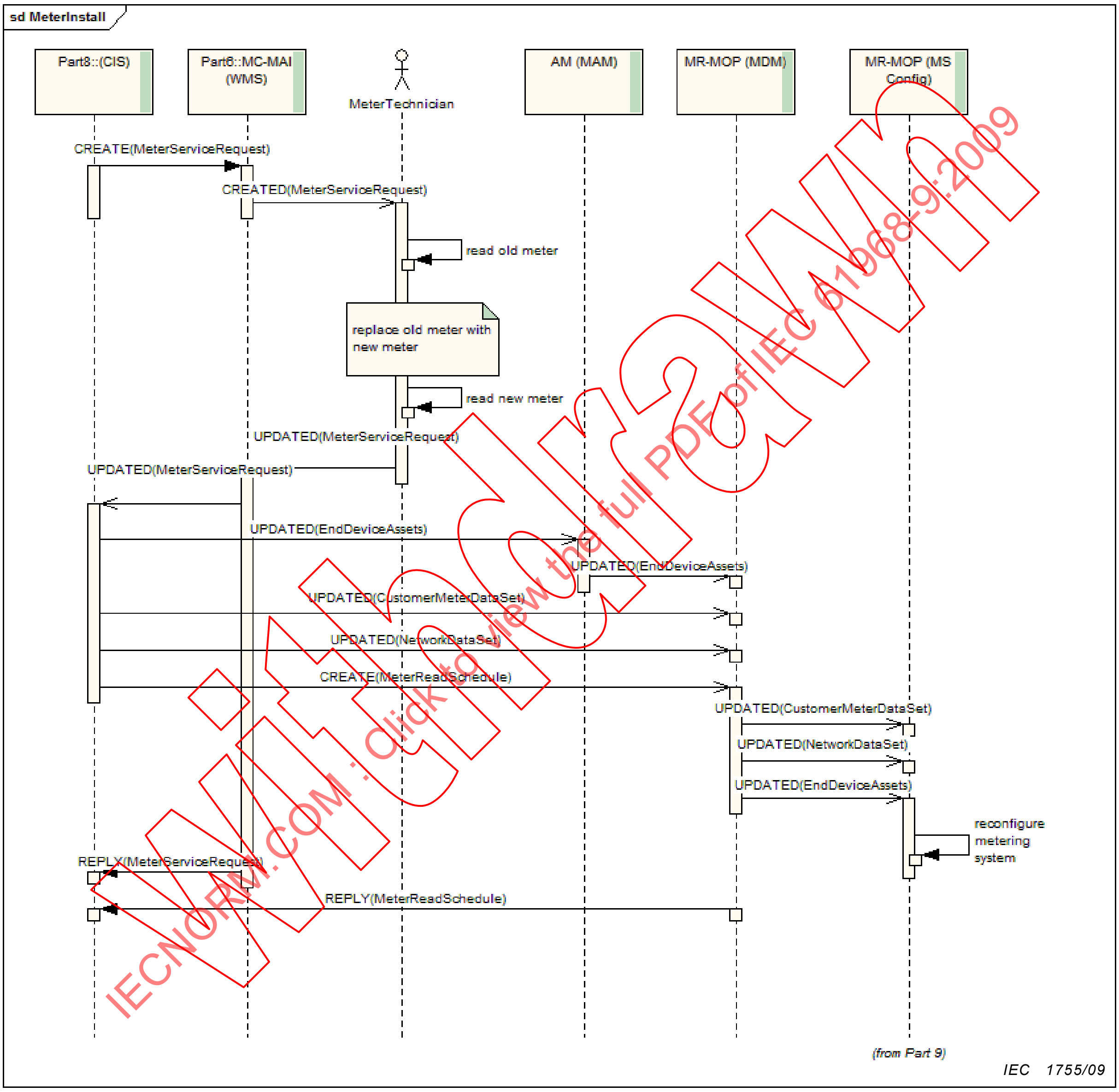


Figure 38 – Example meter installation and removal message exchange

Figure 42 shows the message payload structure used for MeterServiceRequest messages.

The InitialReading and FinalReading are of type MeterReading. MeterReading contains “ServiceDeliveryPoint” which is needed (in addition to ServiceLocation) to help identify the location of the new and old meters.

5.6.2.2 Meter replaced due to end device event

EndDeviceEvents (as described in 5.2) can cause an evaluation to occur which results in a request for a meter change out. Meter events may be published from the MS or MDM. Figure 39 describes an example exchange where the MDM has identified a problem, and brought it to the attention of the WMS which decides to change out the meter.

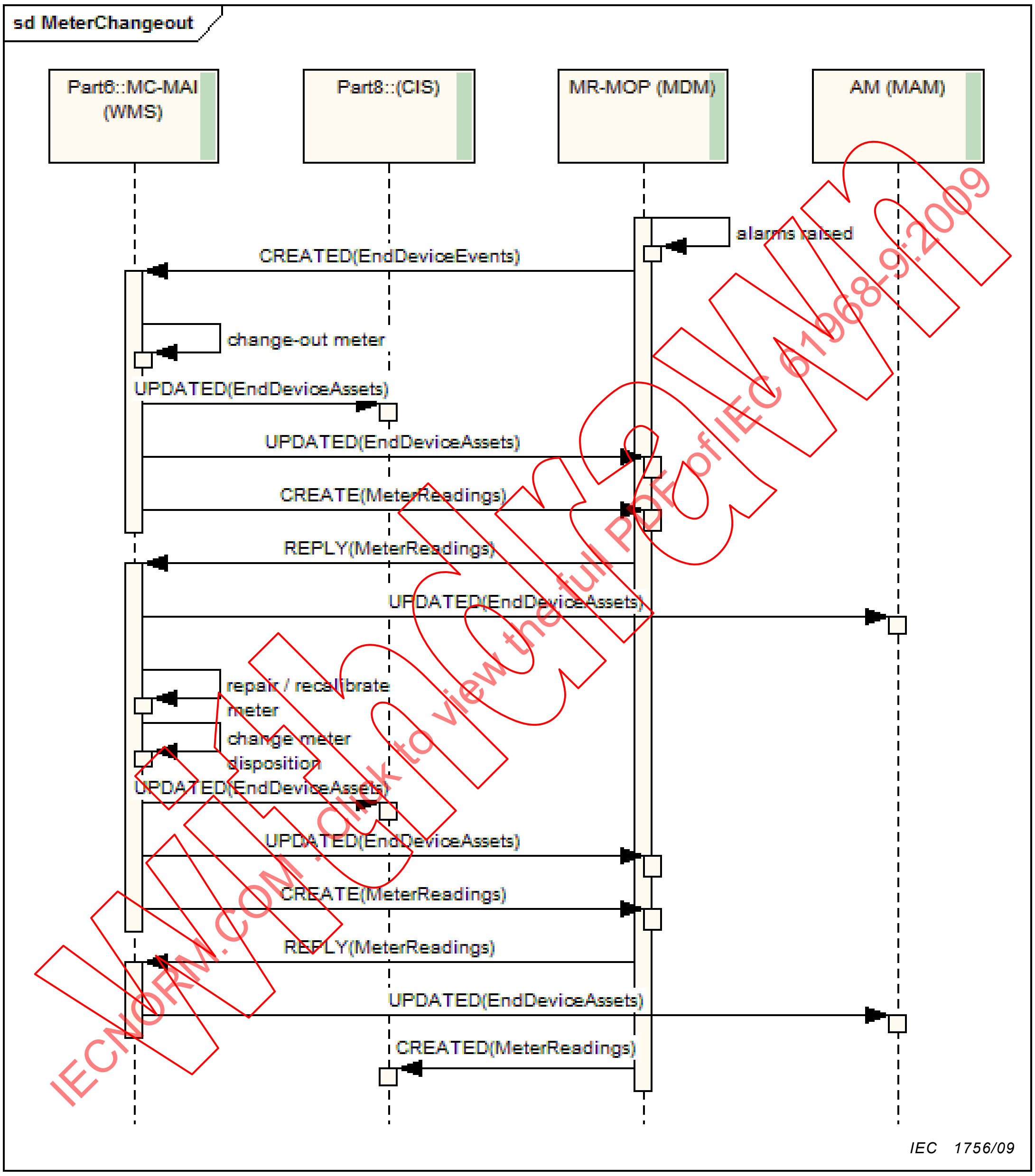


Figure 39 – Example end device event message exchange due to meter changeout

5.6.2.3 Meter changed-out due to CIS alarms or customer complaints

Data analysis by the CIS or customer billing complaints may cause the customer service department to request a site visit to the customer location and possibly a meter change out. Figure 40 describes an example exchange where the MDM has identified a problem and brought it to the attention of the WMS which decides to change out the meter.

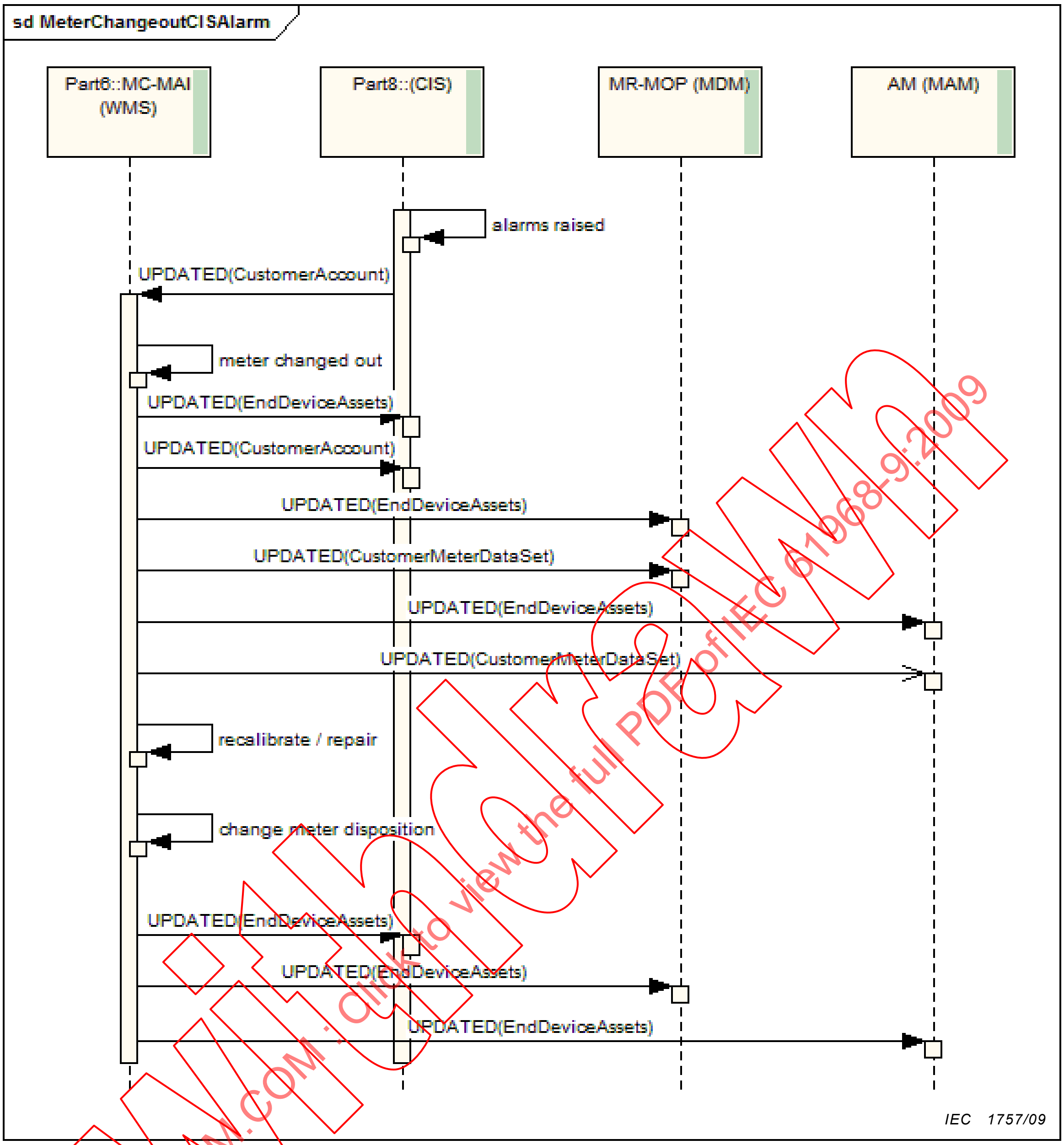


Figure 40 – Example message exchange due to CIS alarms

5.6.2.4 Meter changed-out for recalibration

The WMS may track when meter recalibration is due and change-out the meter. Some utilities may view the actual meter change out as a WMS issue as depicted in Figure 41, while others treat it as a MS issue.

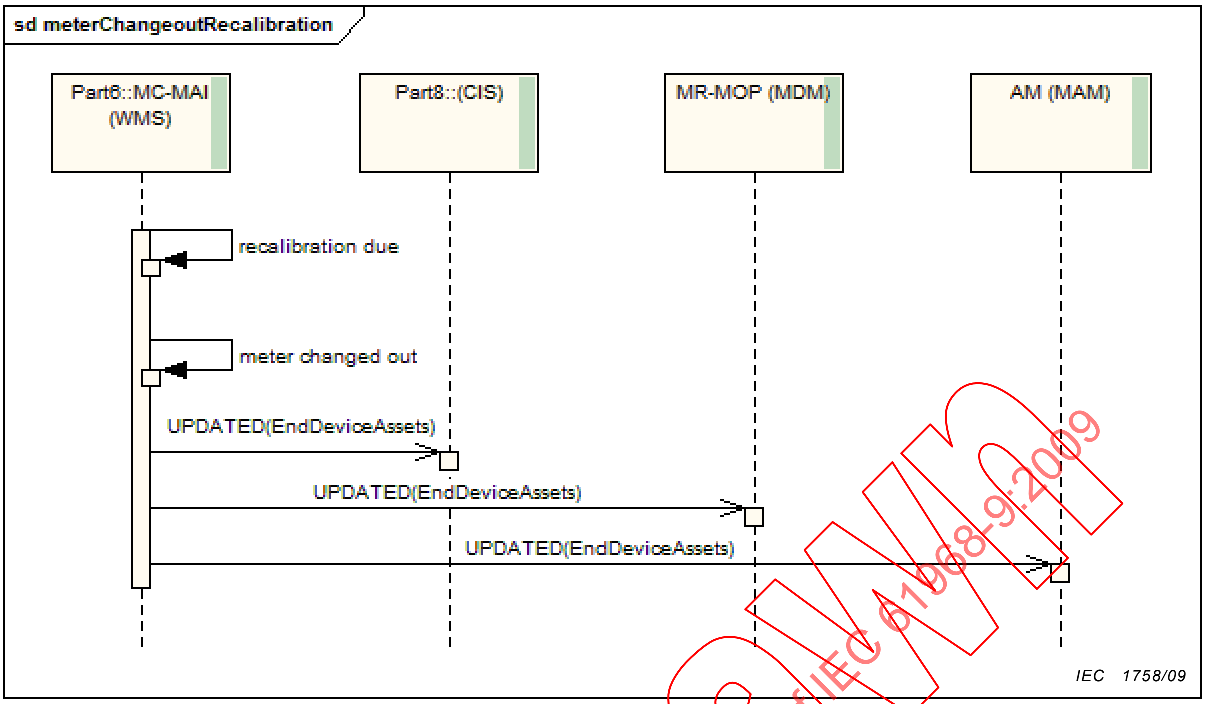
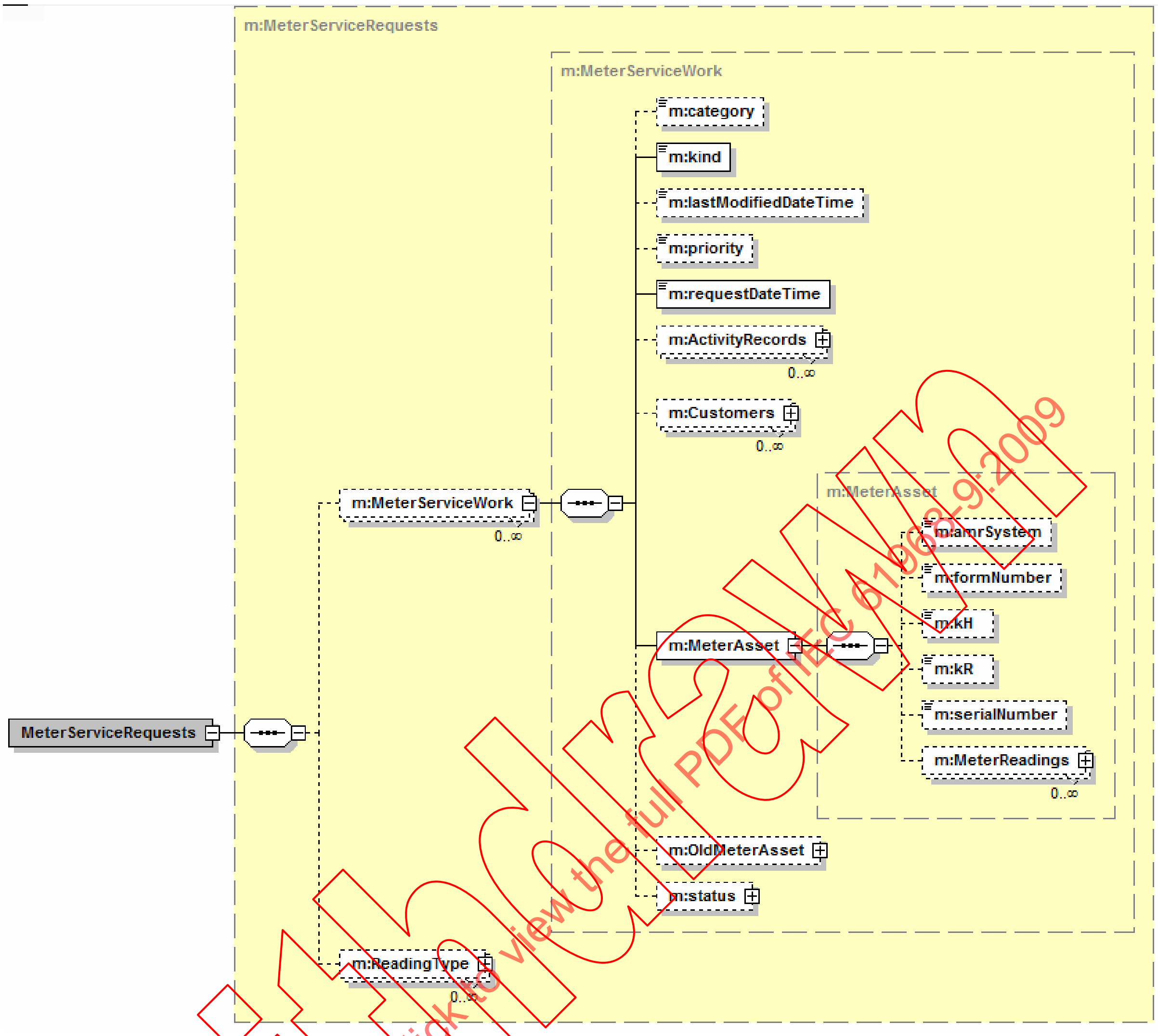


Figure 41 – Example message exchange when meter is changed out for recalibration

Changes are published using an EndDeviceAsset message where the ActivityRecord structure within the message may be leveraged.

5.6.3 Message format

The following message payload structure describes MeterServiceRequests. It can be used for zero or more MeterServiceWork objects.



IEC 1759/09

Figure 42 – Meter service requests message format

The message may include more than one MeterServiceWork item. Each item may refer to up to two meters to provide the means to replace a meter. Meter readings can be obtained as a part of the work. In the case of a meter replacement, readings for the old meter can be obtained as well as initial readings for the new meter. Location references can be supplied as a part of the customer data or for the meter asset, as appropriate for the work being done. The detailed, annotated XML schema is defined in Clause H.8.

5.7 Metering system events

5.7.1 General

There are different levels of need within the system for different kinds of messages.

The metering system (*per se*) may develop problems within its infrastructure which require a work order to fix. The data requirements for communicating this event are different than the requirements of EndDeviceEvent as described in 5.2. With an EndDeviceEvent, a simple error code can be used to report a problem. This mechanism works when the recipient is able to look up and interpret the meaning of the code. However, when a more detailed description of the problem is required, a more elaborate schema must be used. In some cases, the recipient is a human and requires a human-readable description of the problem and the particular piece of equipment involved. When identifying the asset, it will probably be necessary to use vendor-specific terminology to describe the hardware.

5.7.2 Applications

5.7.2.1 Meter system health

Metering system infrastructure issues, such as communication outages, may require EventMessaging to systems such as the LMS, MDM, and OMS; and WorkOrder messaging to communication network operations systems.

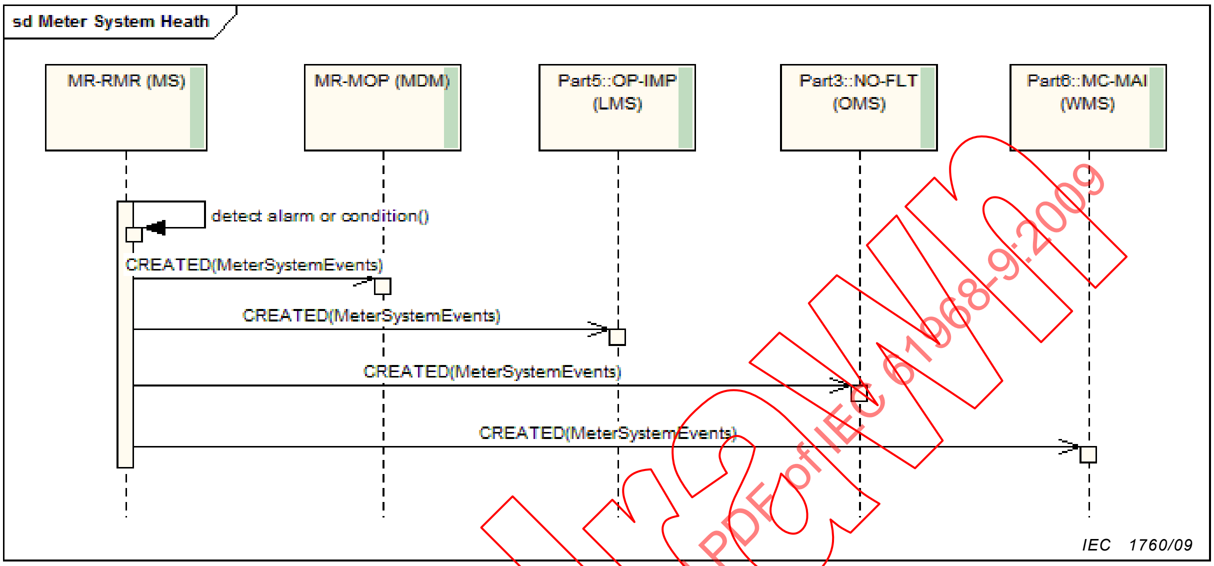

sd Meter System Heath
MR-RMR (MS)
MR-MOP (MDM)
Part5::OP-IMP (LMS)
Part3::NO-FLT (OMS)
Part6::MC-MAI (WMS)
detect alarm or condition()
CREATED(MeterSystemEvents)
CREATED(MeterSystemEvents)
CREATED(MeterSystemEvents)
CREATED(MeterSystemEvents)
IEC 1760/09

Figure 43 – Example of MeterSystemHealth message exchange

An example message exchange in which the EndDeviceEvent is communicated to various stakeholders is depicted above in Figure 43.

This type of message would carry updates to the CommEquipmentAsset, which in turn can carry the name and serial number of the affected equipment, the type of CommEquipment, the event code, as well as ActivityRecord status, statusReason, remarks, and a timestamp.

5.7.2.2 Firmware upgrade

For an automated metering system capable of two-way communication, it may be possible to upgrade the firmware of MS infrastructure in the field, or in some systems the communication assets in the end devices. Changes to the metering system that materially affect its operation must be communicated to all of the relevant stakeholders. An example of such a publication is depicted in Figure 44.

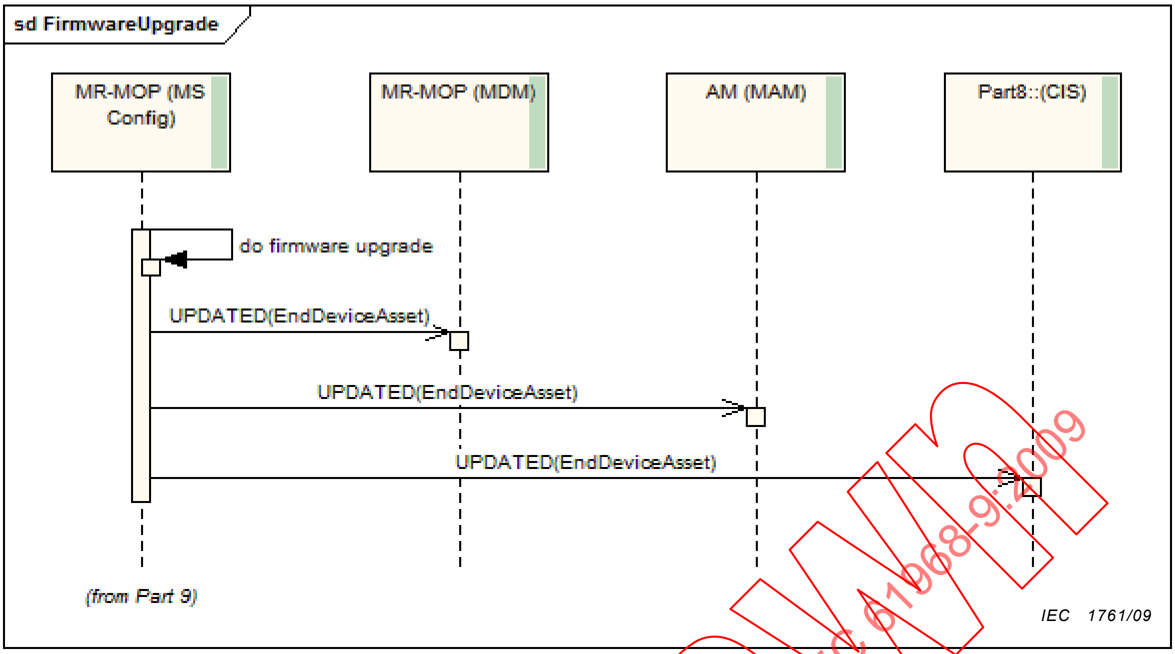


Figure 44 – Example of firmware upgrade message exchange

The firmware upgrade may cause the EndDeviceAsset or CommEquipmentAsset to change in a number of ways, including device functional characteristics, firmware version and revision number, or configuration (expressed as a programId).

5.7.3 Message formats

5.7.3.1 MeterSystemEvent

Figure 45 describes the structure of a meter system event message.

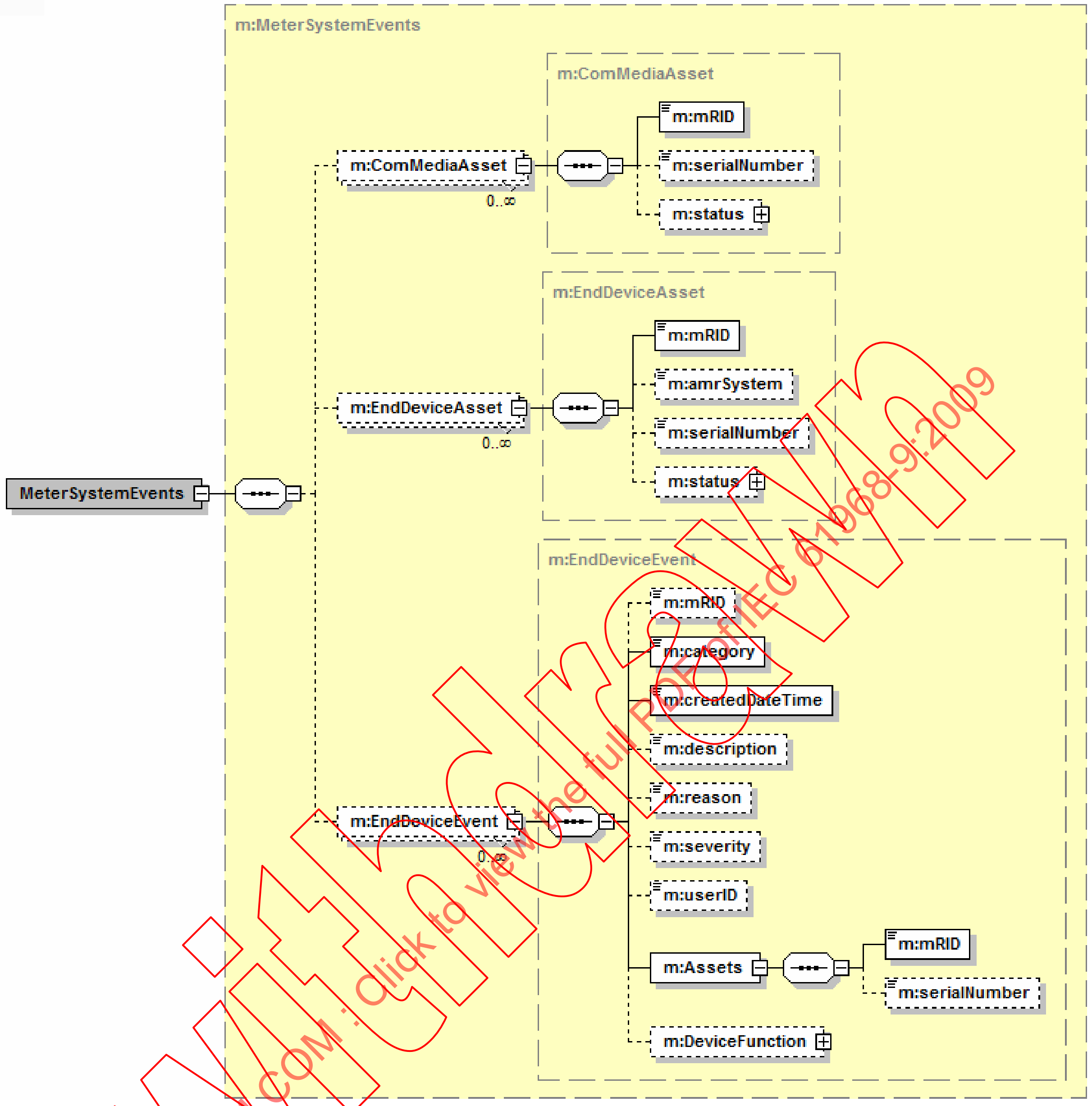
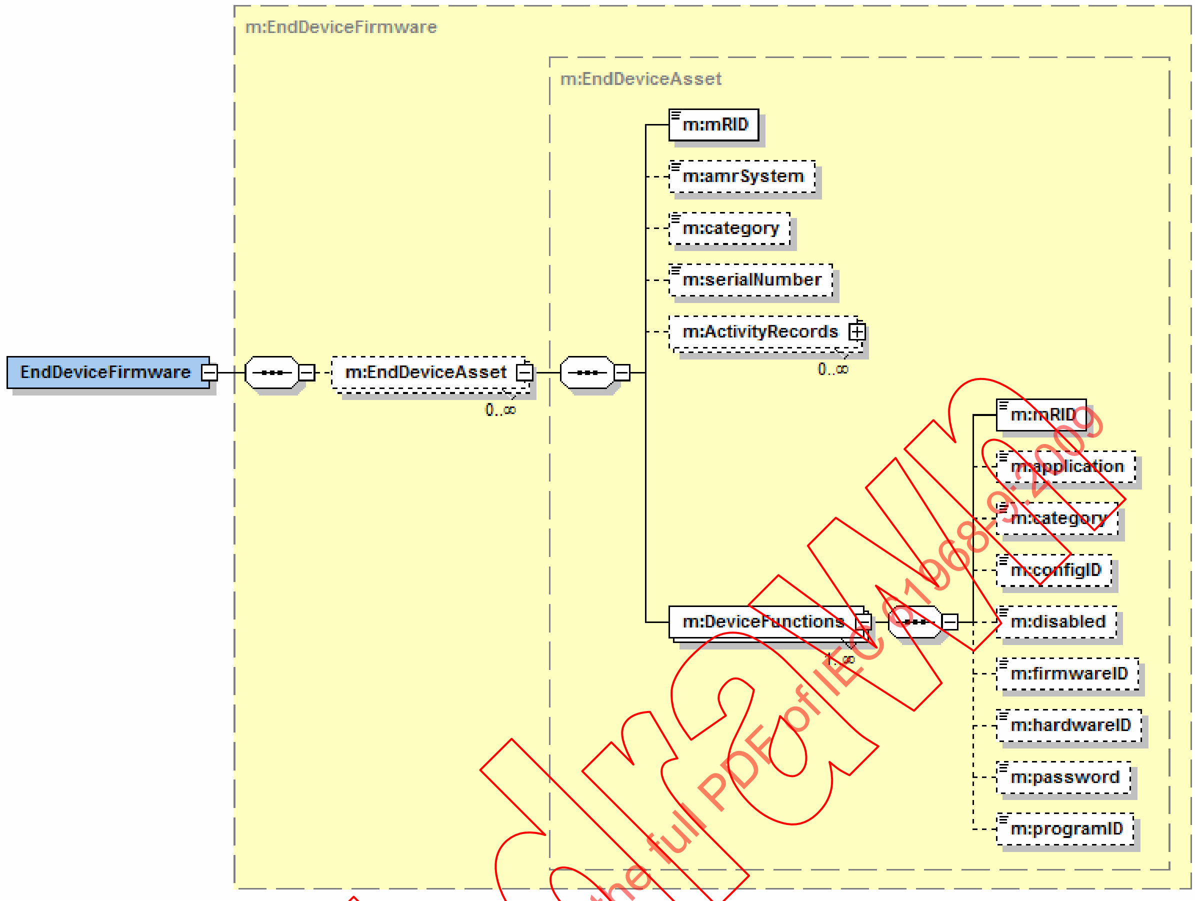


Figure 45 – Example of meter system event message format

The MeterSystemEvent schema supports the reporting of activity within any type of end device (including meters), as well as within the MS communication infrastructure. As described above, the MeterSystemEvent can be used to describe an event in the end device itself, or in the MS communication infrastructure. The detailed XML schema is described in Clause H.9.

5.7.3.2 End device firmware

The structure of an end device firmware message is described by Figure 46.



IEC 1763/09

Figure 46 – End device firmware message format

The EndDeviceFirmware message structure supports the naming of the affected device function, as well as additional parameter details that may be necessary in the report. It should be noted that a DeviceFunction can be used to describe software components as well as firmware or hardware. This structure, together with the ActivityRecord can effectively describe the nature of the event, when the event occurred, and what has happened. The detailed XML schema is provided in Clause H.11.

5.8 Customer switching

5.8.1 General

A customer in an open retail market can switch between energy suppliers. This may require reconfiguration and/or reinstallation of the meter. The reconfiguration or meter replacement may be a consequence of a customer changing energy programs at the time of the change of energy supplier. This process would likely involve an on request read as needed for final billing purposes. An example message exchange is provided below in Figure 47.

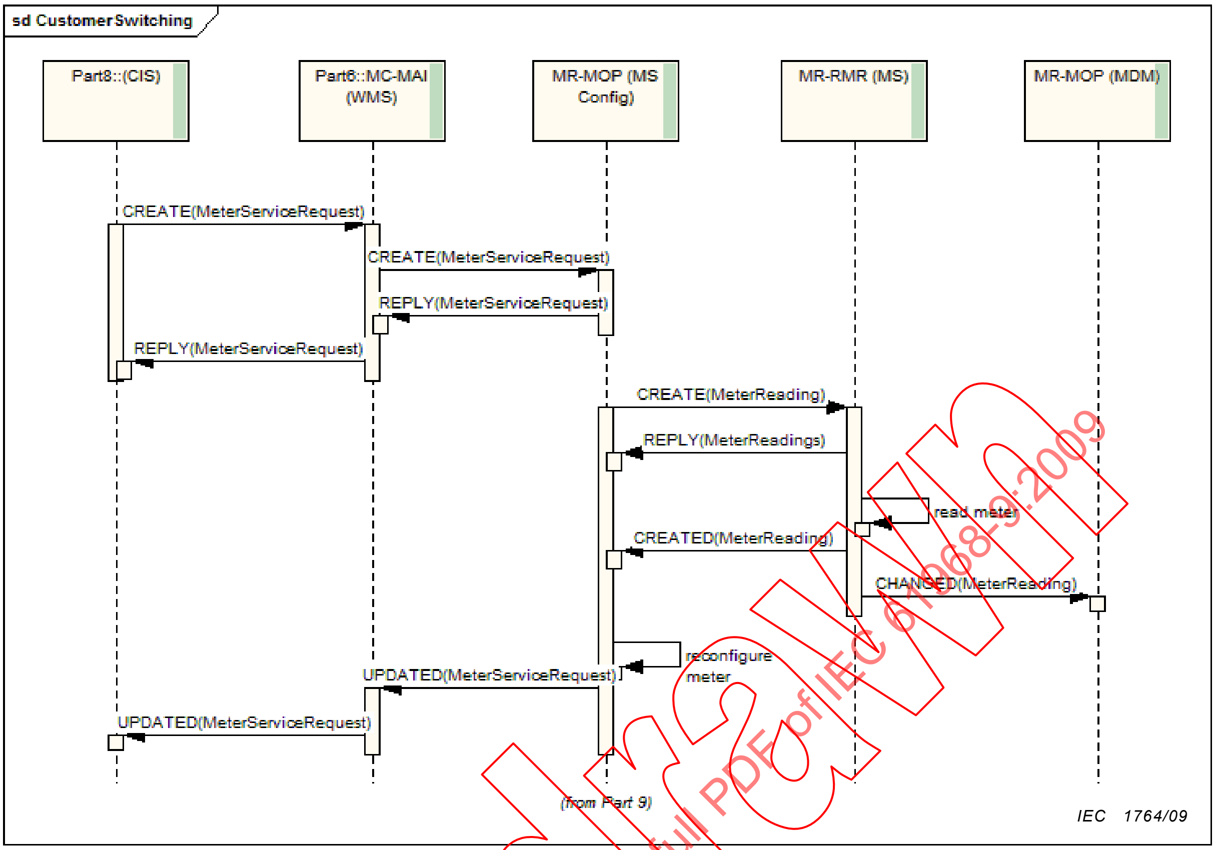


Figure 47 – Example of customer switching message exchange

5.8.2 Message format

The structure of a MeterServiceRequest was described by Figure 42. The structure of a MeterReading message was described by Figure 23.

5.9 Payment metering service messages

5.9.1 General

Payment metering services typically involve prepayment meters installed at customers' premises and token vending systems situated at convenient locations to serve the needs of customers. For the purpose of this standard, vending systems are regarded as being part of the metering system. Such vending systems typically require information about the meter, customer, service location, service delivery point, tariff, customer account, customer agreement and the supplier of the service that receives the revenue collected by the vending systems.

The information required by the vending systems usually originates in the CIS and billing system and is then transferred to the vending systems by means of the messages provided in 5.9.2.

Transaction and receipt information on the other hand originate in the vending systems when the customer makes a payment on an account or purchases a prepaid token at a point of sale. This information periodically needs to be transferred to the CIS and billing system for further processing, for which purpose the messages in 5.9.3 are provided.

5.9.2 MS information configuration

5.9.2.1 General

In order to operate as a payment metering system, MS typically requires to be configured with information about the supplier of the service, the service being provided, the customers of the supplier, customer agreements, customer accounts, debt to be collected against customer agreements, tariffs, service delivery points, service delivery point locations, service locations and meters deployed in the metering system.

This information typically originates in the CIS and billing system, from where it periodically gets transferred to MS for configuration and update. The messages defined in 5.9.2 make provision for transferring this information from the CIS and billing system.

5.9.2.2 Suppliers

Information about the supplier (retailer or utility for example) of the service may be configured into MS by means of SupplierConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 48 and the message format is given in Figure 49.

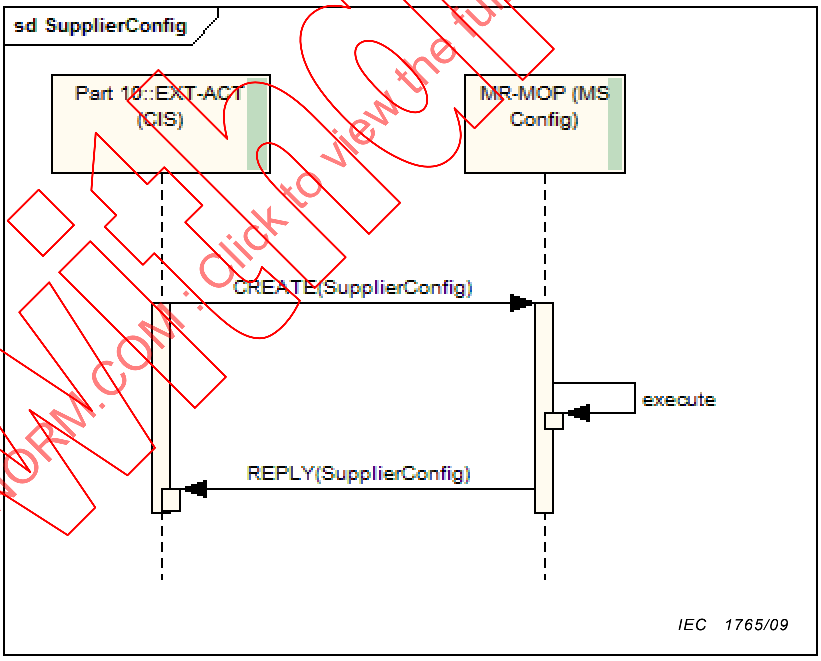
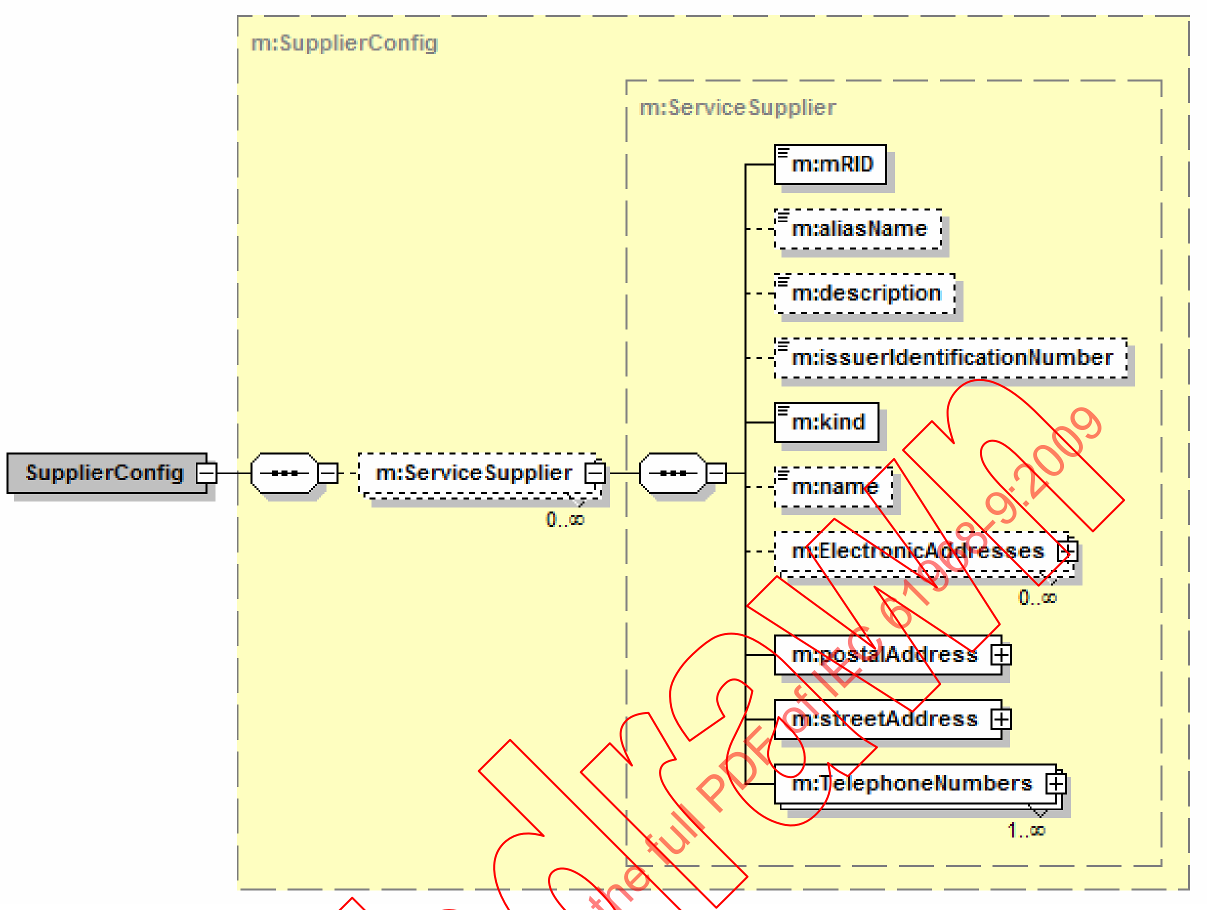


Figure 48 – Message exchange for transferring supplier information



IEC 1766/09

Figure 49 – Supplier configuration message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.12.

5.9.2.3 Customers

Information about customers of the supplier of the service may be configured into MS by means of CustomerConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 50 and the message format is given in Figure 51.

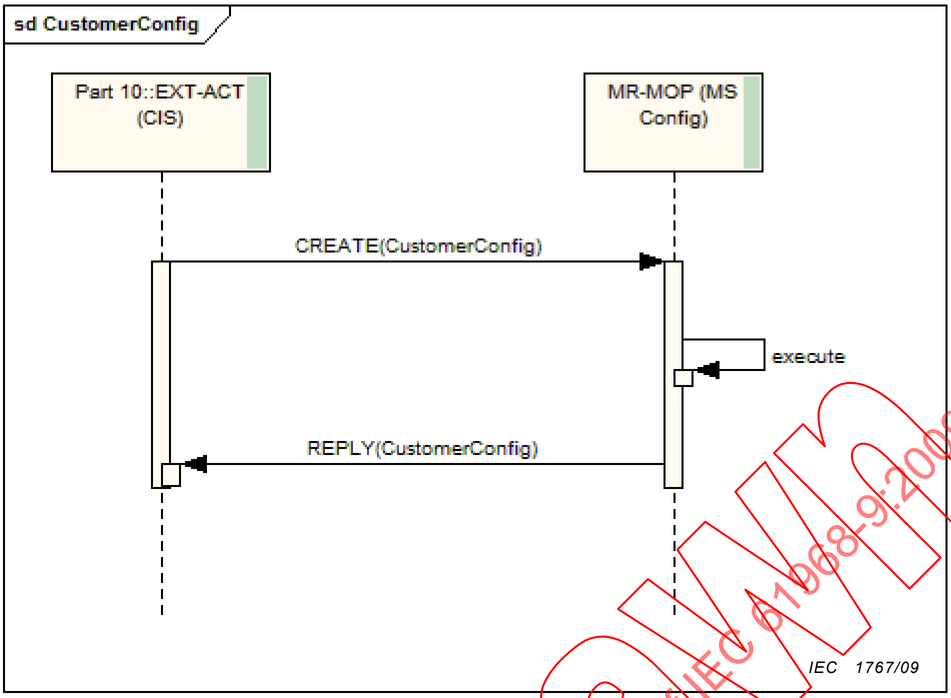


Figure 50 – Message exchange for transferring customer information

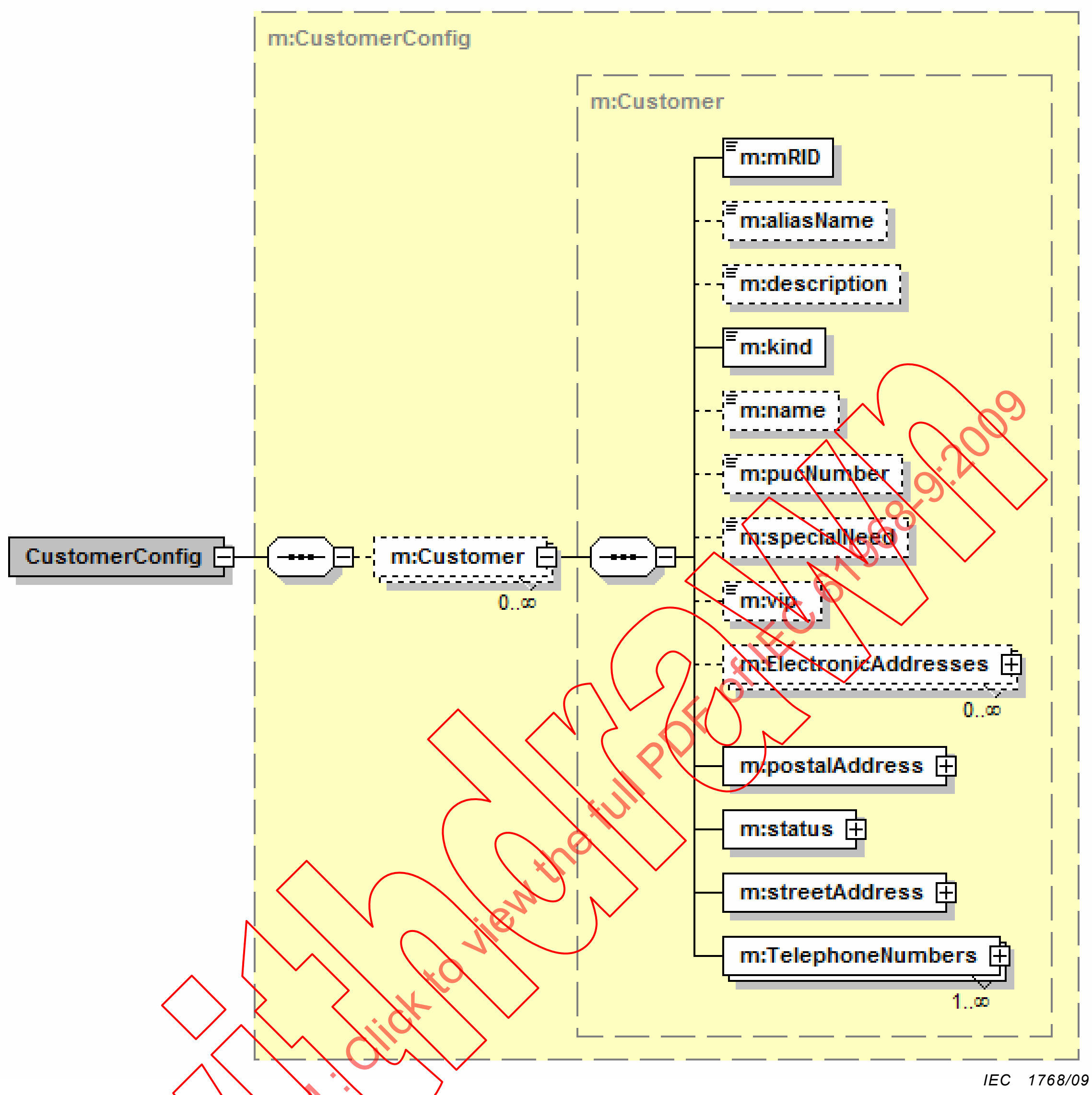


Figure 51 – Customer configuration message

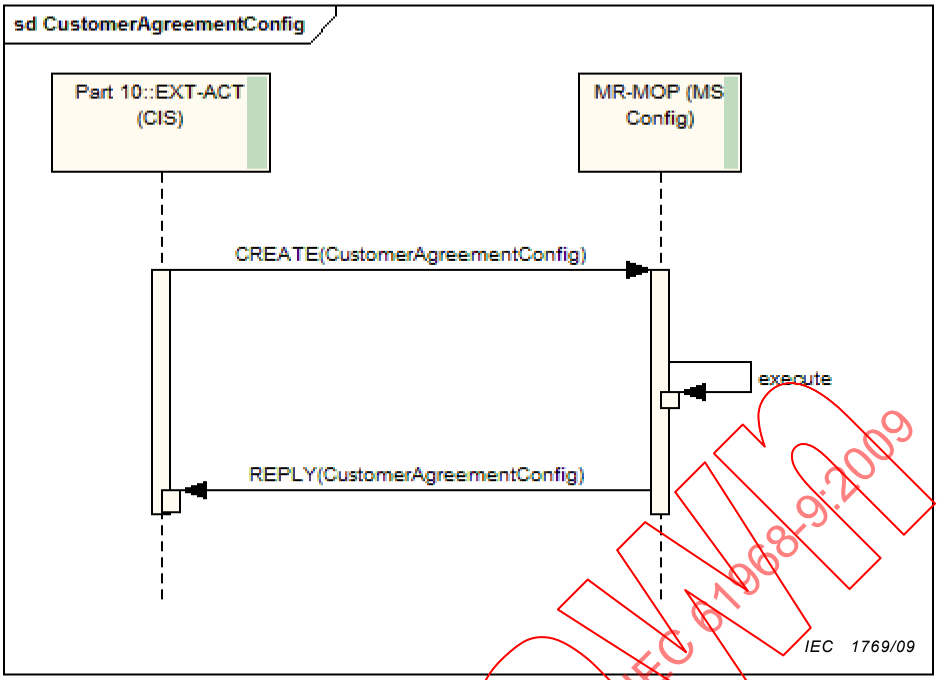
The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.13.

5.9.2.4 Customer agreements

Information about customer agreements may be configured into MS by means of CustomerAgreementConfig message payload.

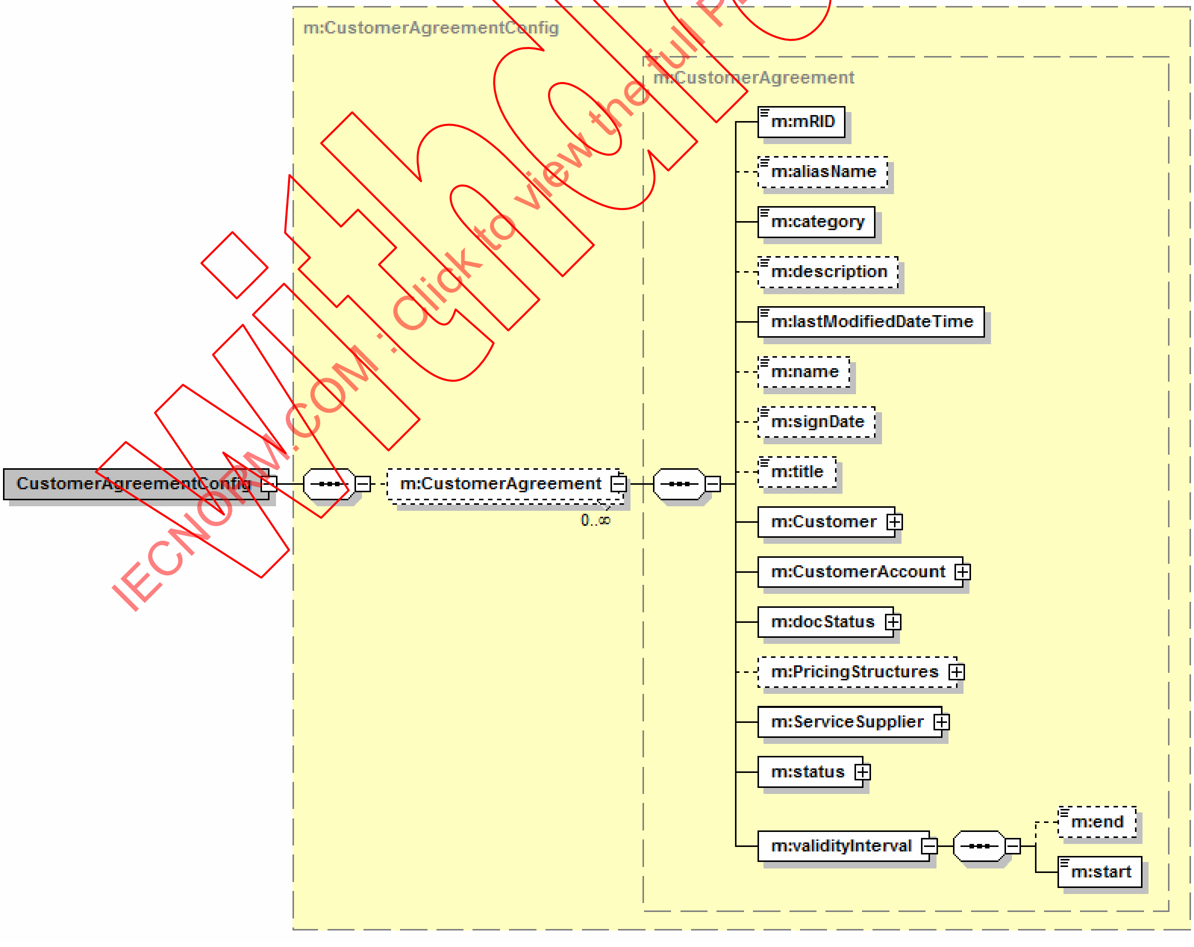
In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 52 and the message format is given in Figure 53.


sd CustomerAgreementConfig
Part 10::EXT-ACT (CIS)
MR-MOP (MS Config)
CREATE(CustomerAgreementConfig)
execute
REPLY(CustomerAgreementConfig)

IEC 1769/09

Figure 52 – Message exchange for transferring customer agreement information



IEC 1770/09

Figure 53 – Customer agreement configuration message

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.17.

5.9.2.5 Customer accounts

Information about customer accounts may be configured into MS by means of CustomerAccountConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 54 and the message format is given in Figure 55.

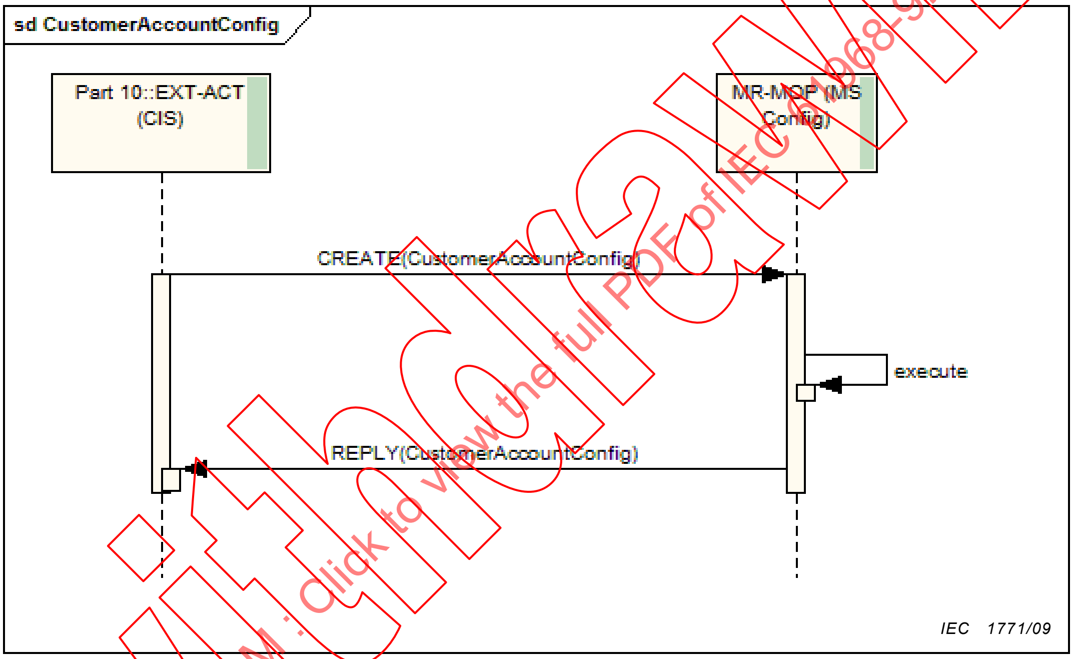
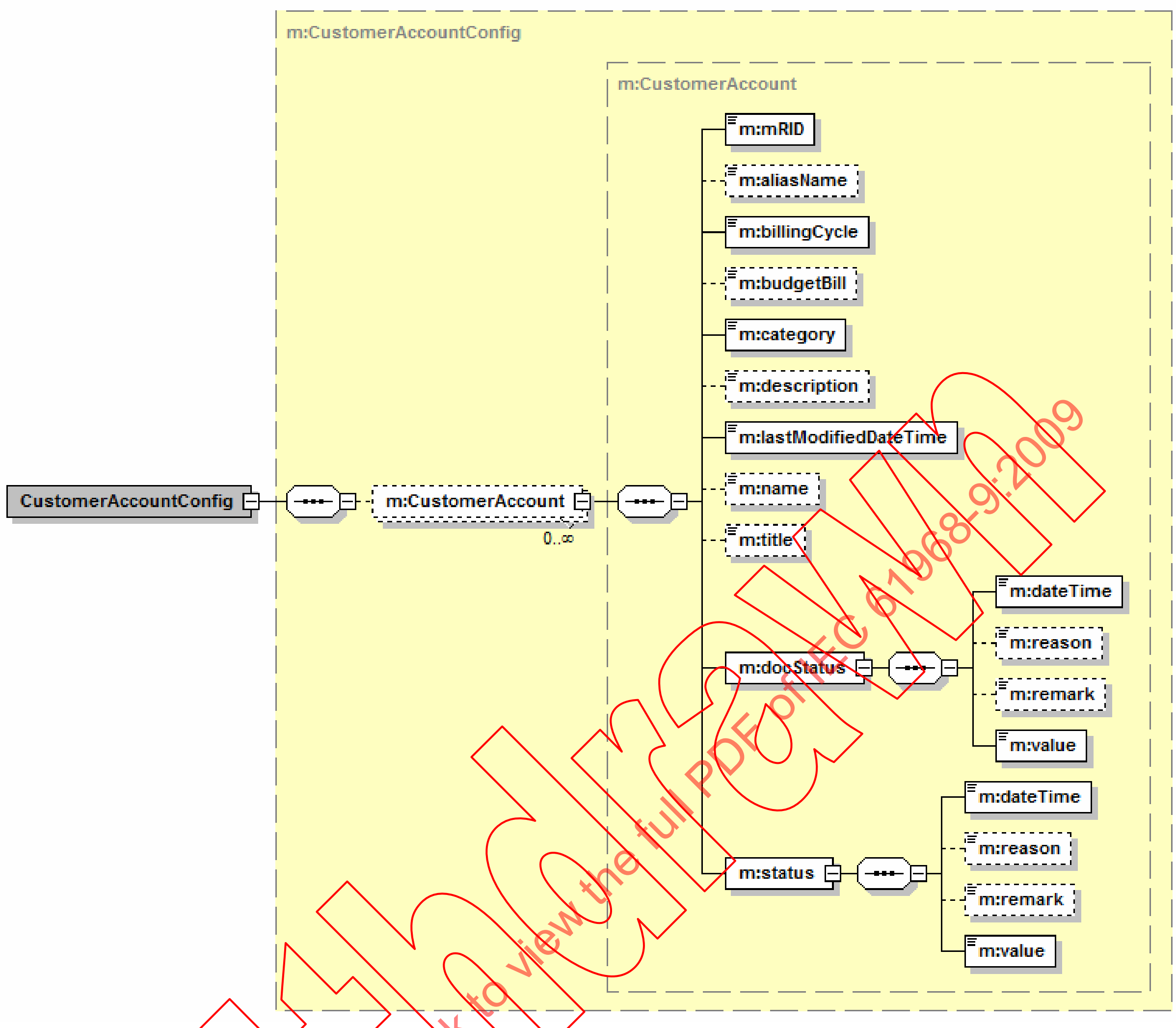


Figure 54 – Message exchange for transferring customer account information



IEC 1772/09

Figure 55 – Customer account configuration message

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.16.

5.9.2.6 Auxiliary agreements

Information about customer debt for collection may be configured into MS by means of AuxilaryAgreementConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 56 and the message format is given in Figure 57.

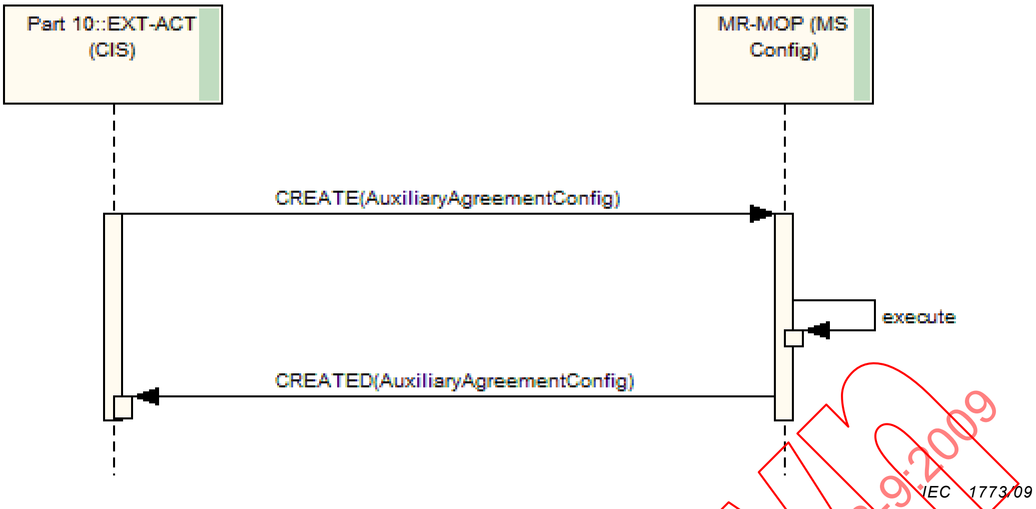

Part 10::EXT-ACT (CIS)
MR-MOP (MS Config)
CREATE(AuxiliaryAgreementConfig)
execute
CREATED(AuxiliaryAgreementConfig)
IEC 1773/09

Figure 56 – Message exchange for transferring auxiliary agreement information

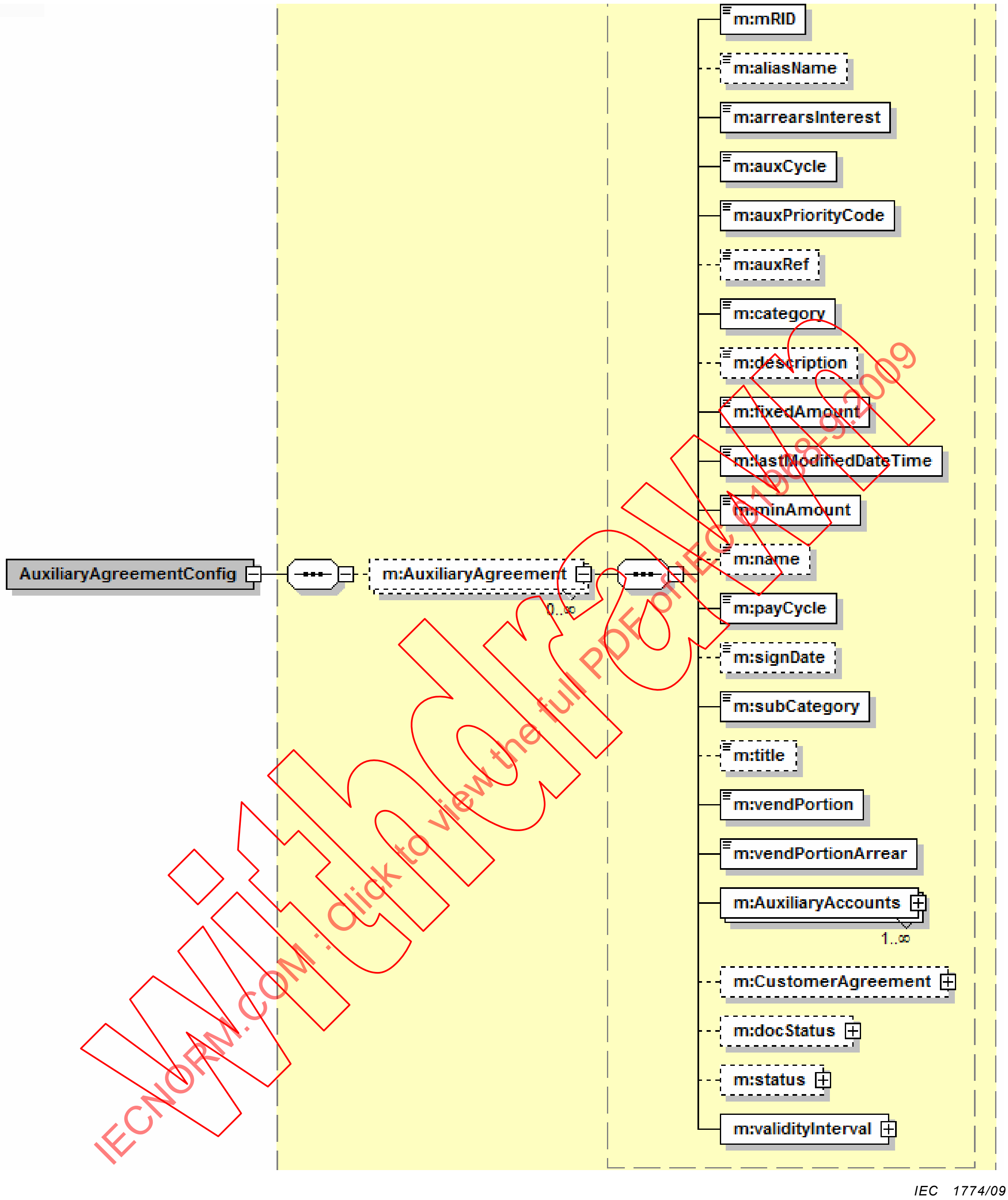


Figure 57 – Auxiliary agreement configuration message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.18.

5.9.2.7 Pricing structure and tariffs

Information about pricing structures may be configured into MS by means of PricingStructureConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 58 and the message format is given in Figure 59.

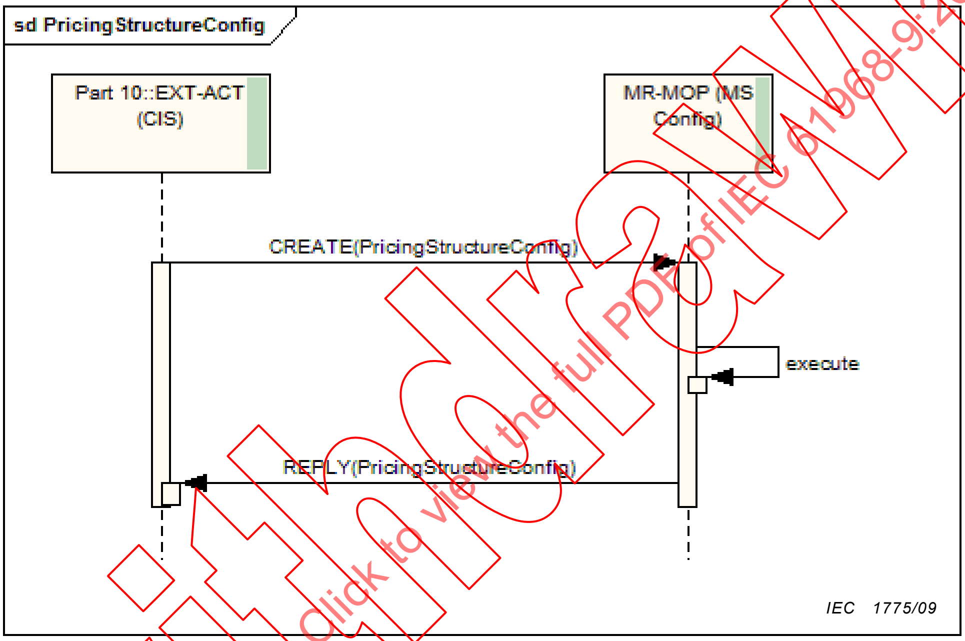
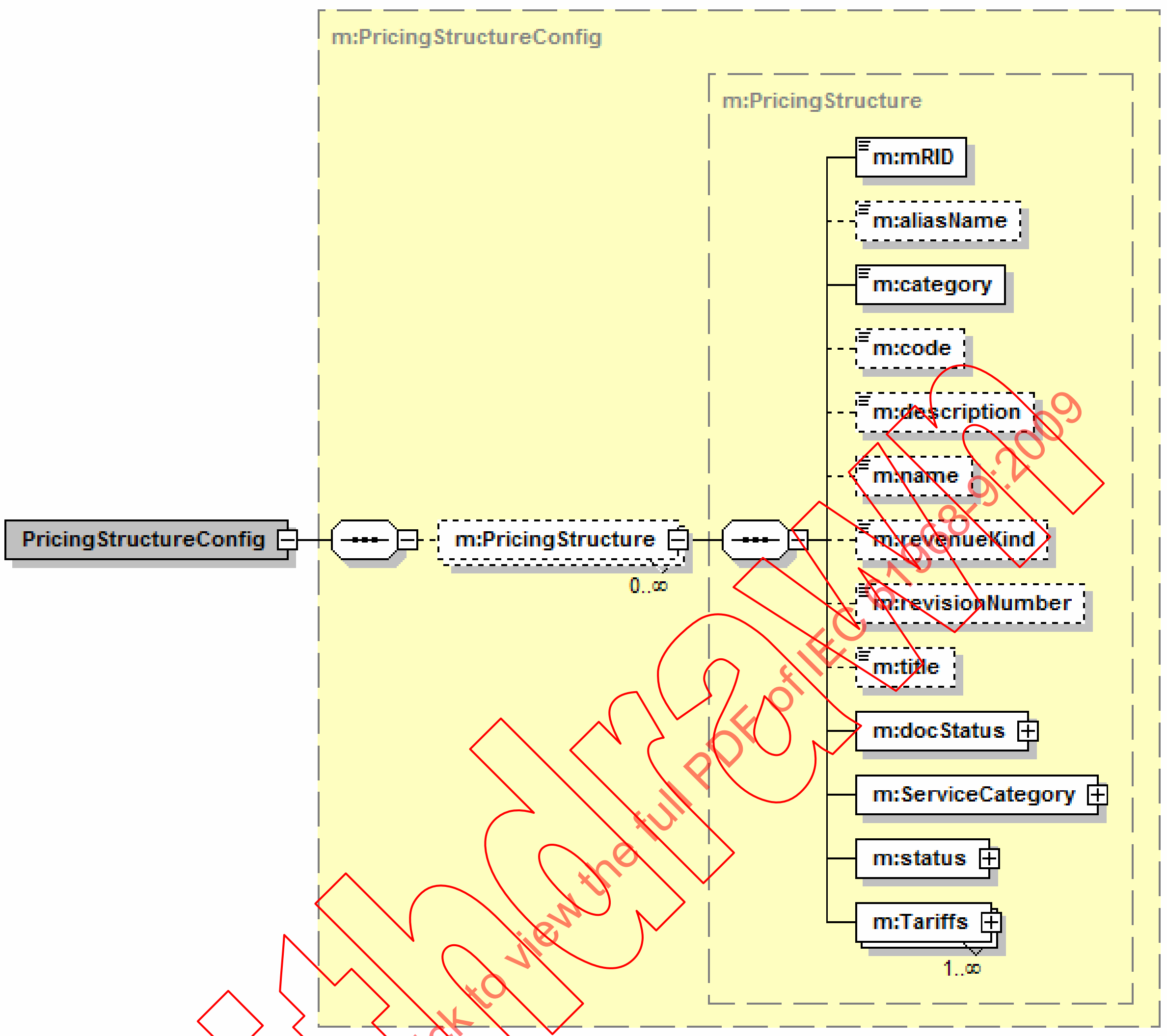

sd PricingStructureConfig
Part 10::EXT-ACT (CIS)
MR-MOP (MS Config)
CREATE(PricingStructureConfig)
execute
REPLY(PricingStructureConfig)
IEC 1775/09

Figure 58 – Message exchange for transferring pricing structures



IEC 1776/09

Figure 59 – PricingStructureConfig message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.15.

5.9.2.8 Service categories

Information about the categories of service (electricity, water, gas for example) may be configured into MS by means of ServiceCategoryConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 60 and the message format is given in Figure 61.

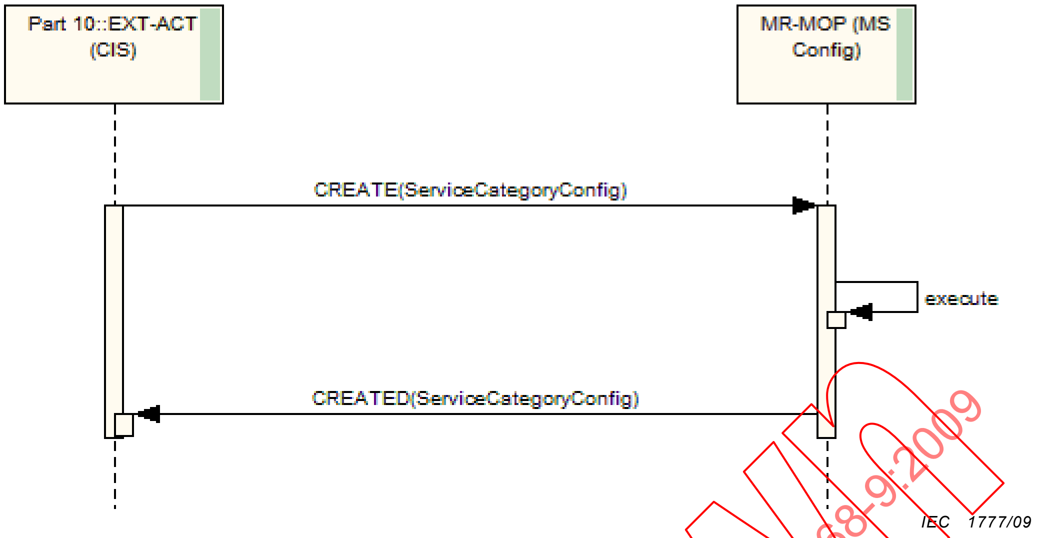


Figure 60 – Message exchange for transferring service category information

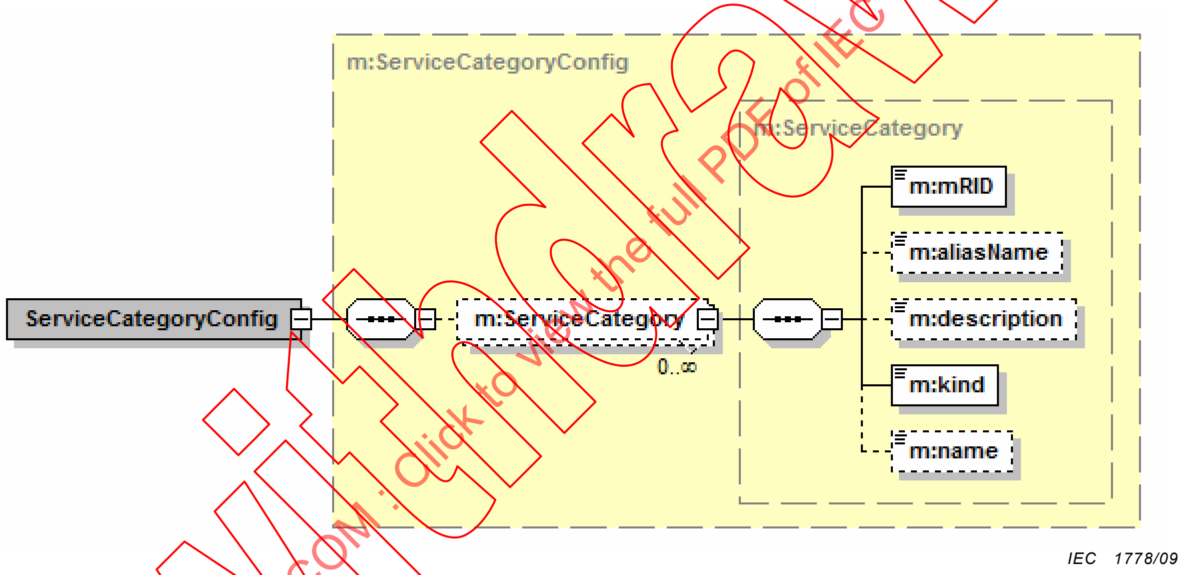


Figure 61 – ServiceCategoryConfig message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.14.

5.9.2.9 Service delivery points

Information about the service delivery points may be configured into MS by means of ServiceDeliveryPointConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 62 and the message format is given in Figure 63.

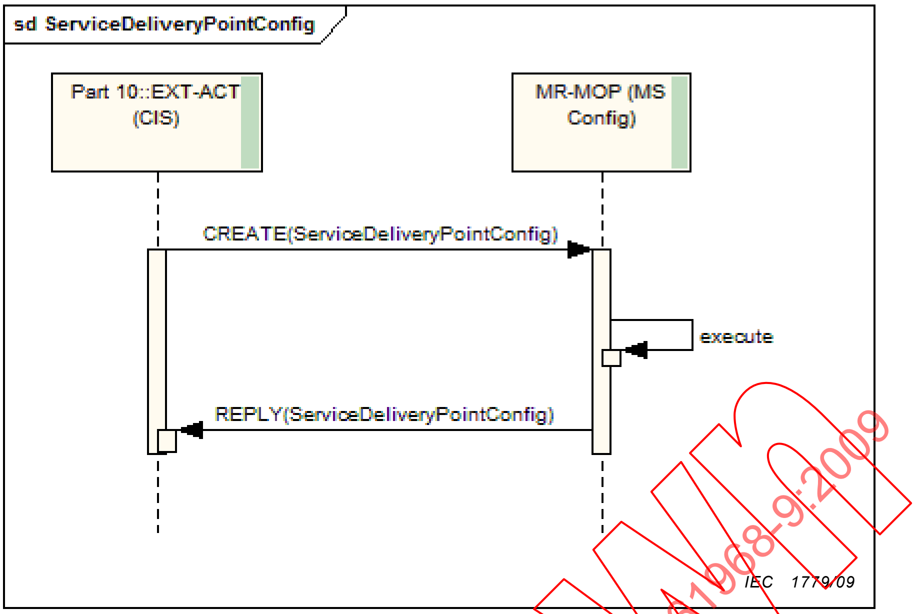

sd ServiceDeliveryPointConfig
Part 10::EXT-ACT (CIS)
MR-MOP (MS Config)
CREATE(ServiceDeliveryPointConfig)
execute
REPLY(ServiceDeliveryPointConfig)
IEC 1779/09

Figure 62 – Message exchange for transferring service delivery point information

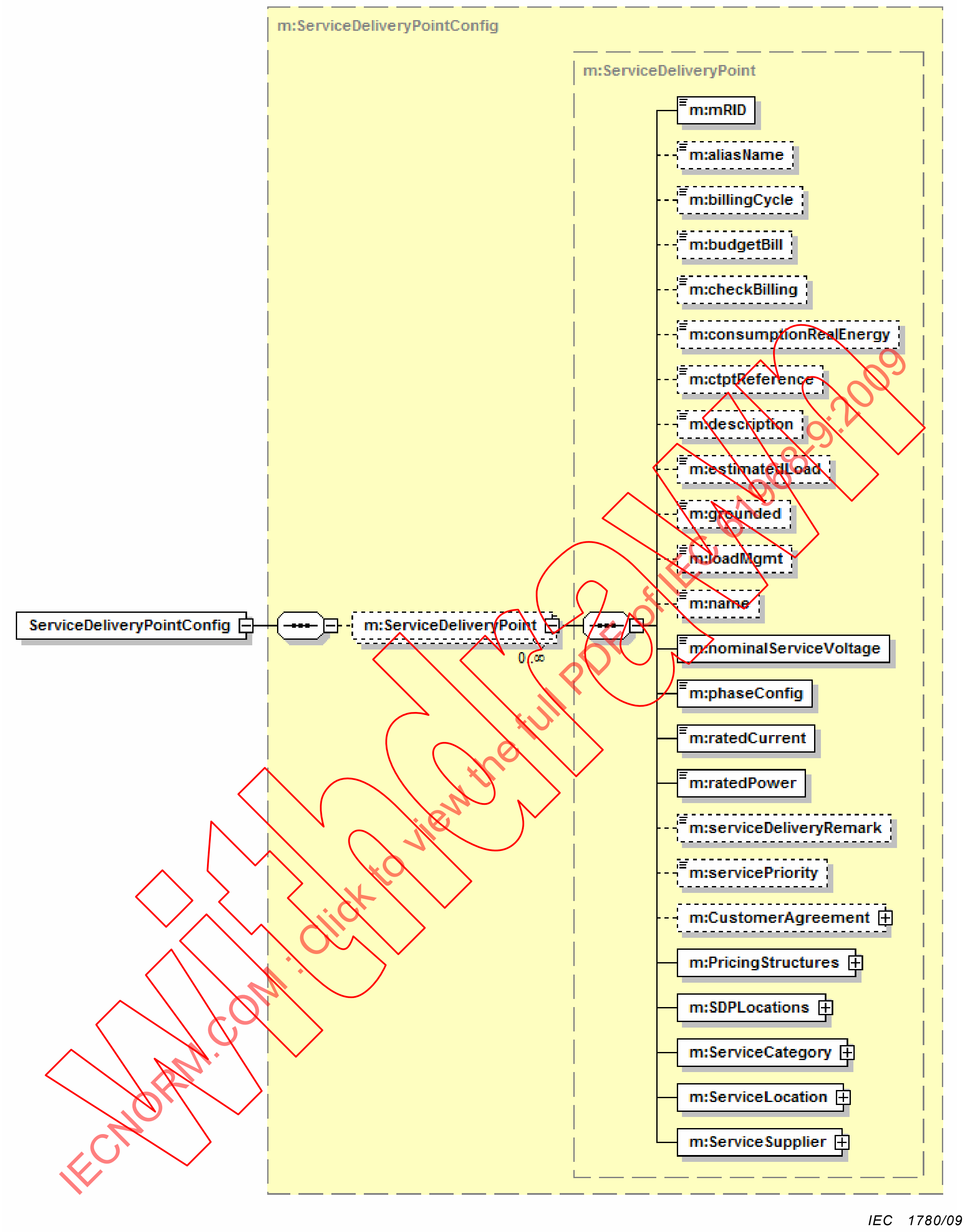


Figure 63 – ServiceDeliveryPointConfig message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.20.

5.9.2.10 Meters

Information about the meters in service may be configured into MS by means of the MeterAssetConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 64 and the message format is given in Figure 65.

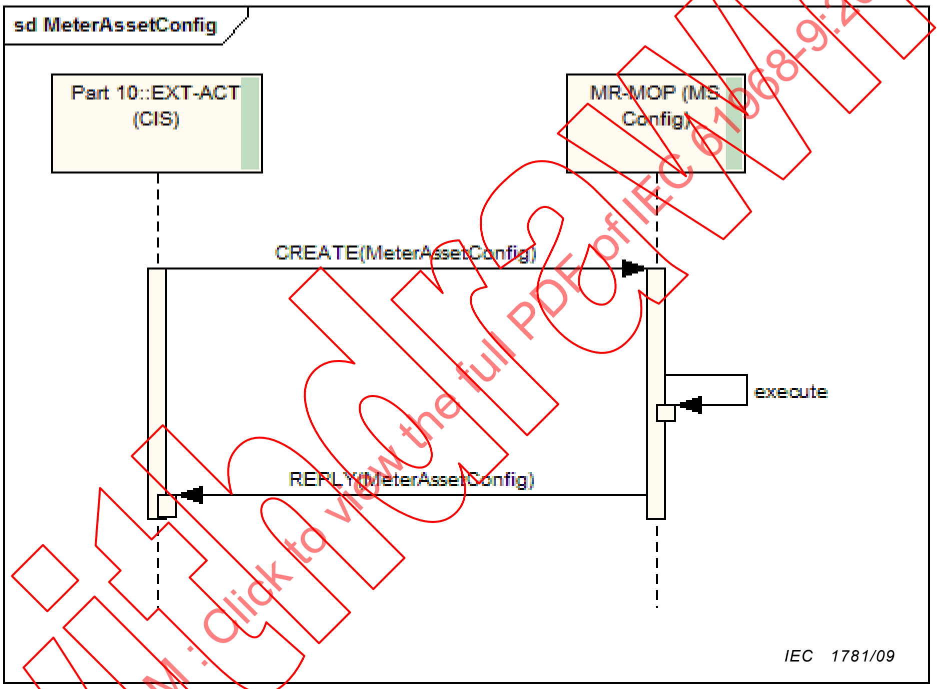
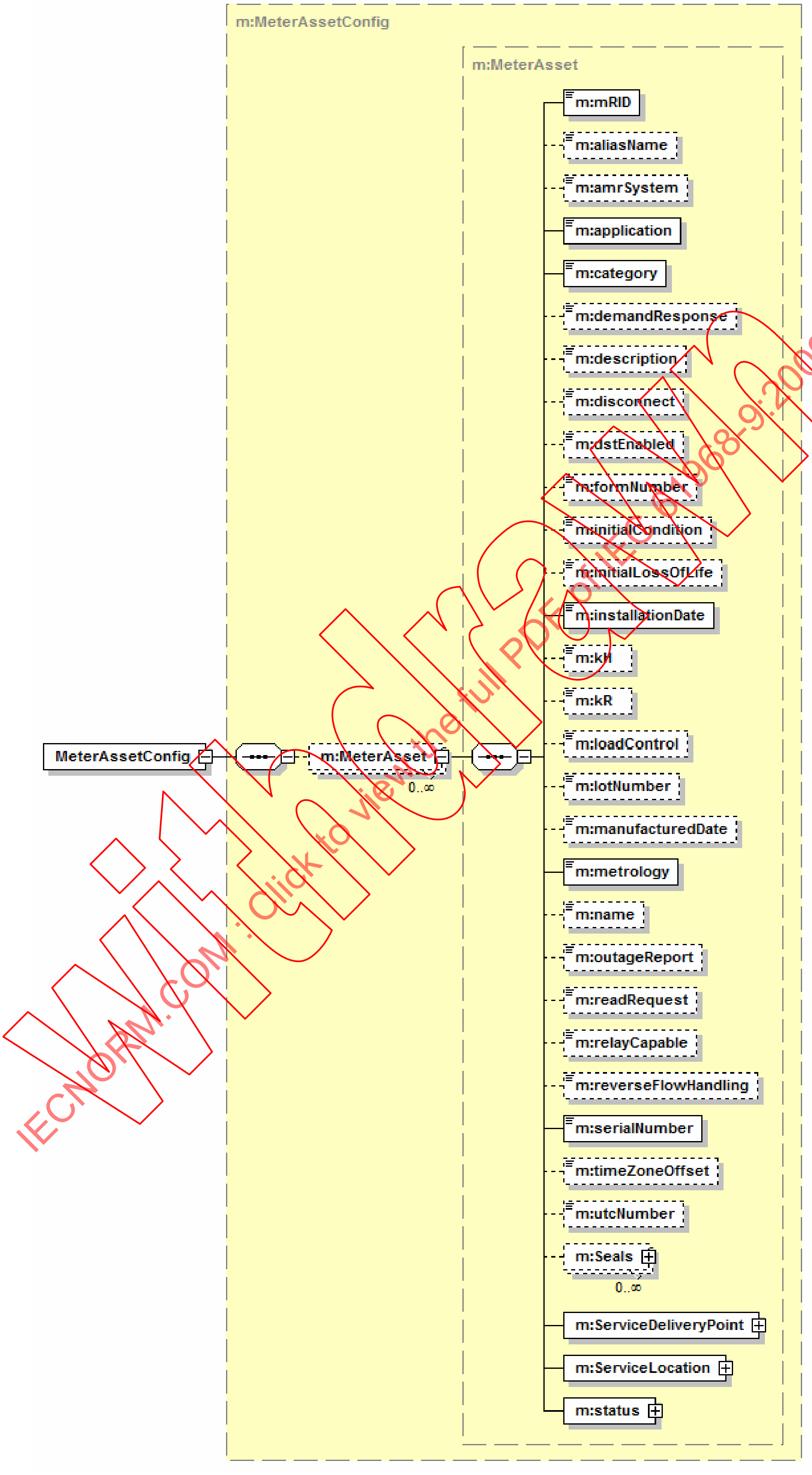


Figure 64 – Message exchange for transferring meter information



IEC 1782/09

Figure 65 – Meter asset configuration message

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.19.

5.9.2.11 Service delivery point locations

Information about the service delivery point locations may be configured into MS by means of the SDPLocationConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 66 and the message format is given in Figure 67.

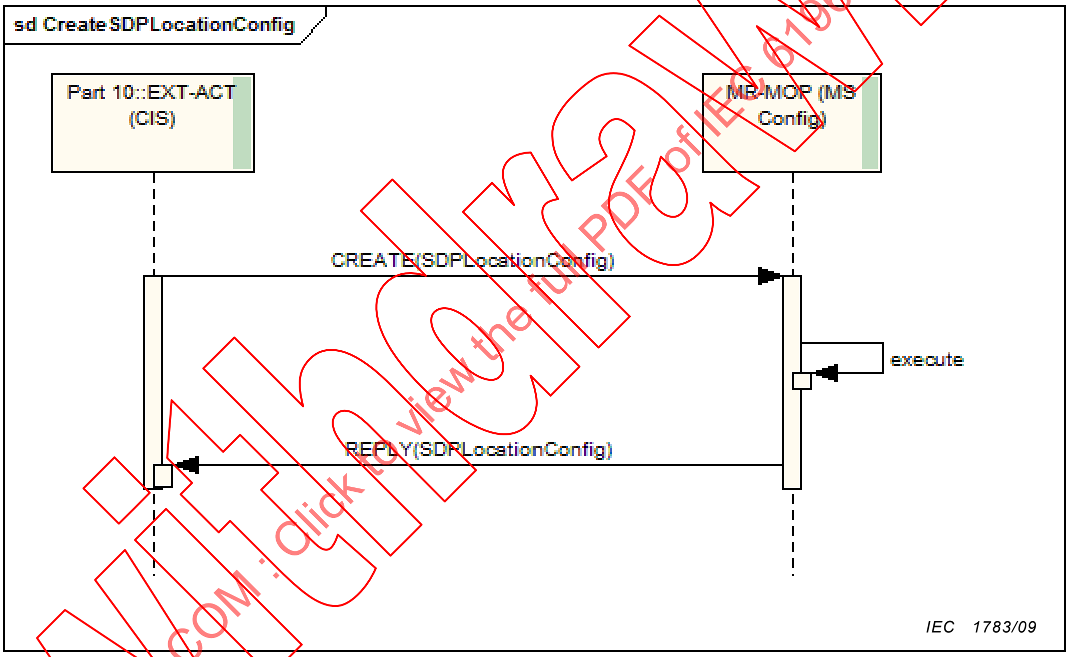
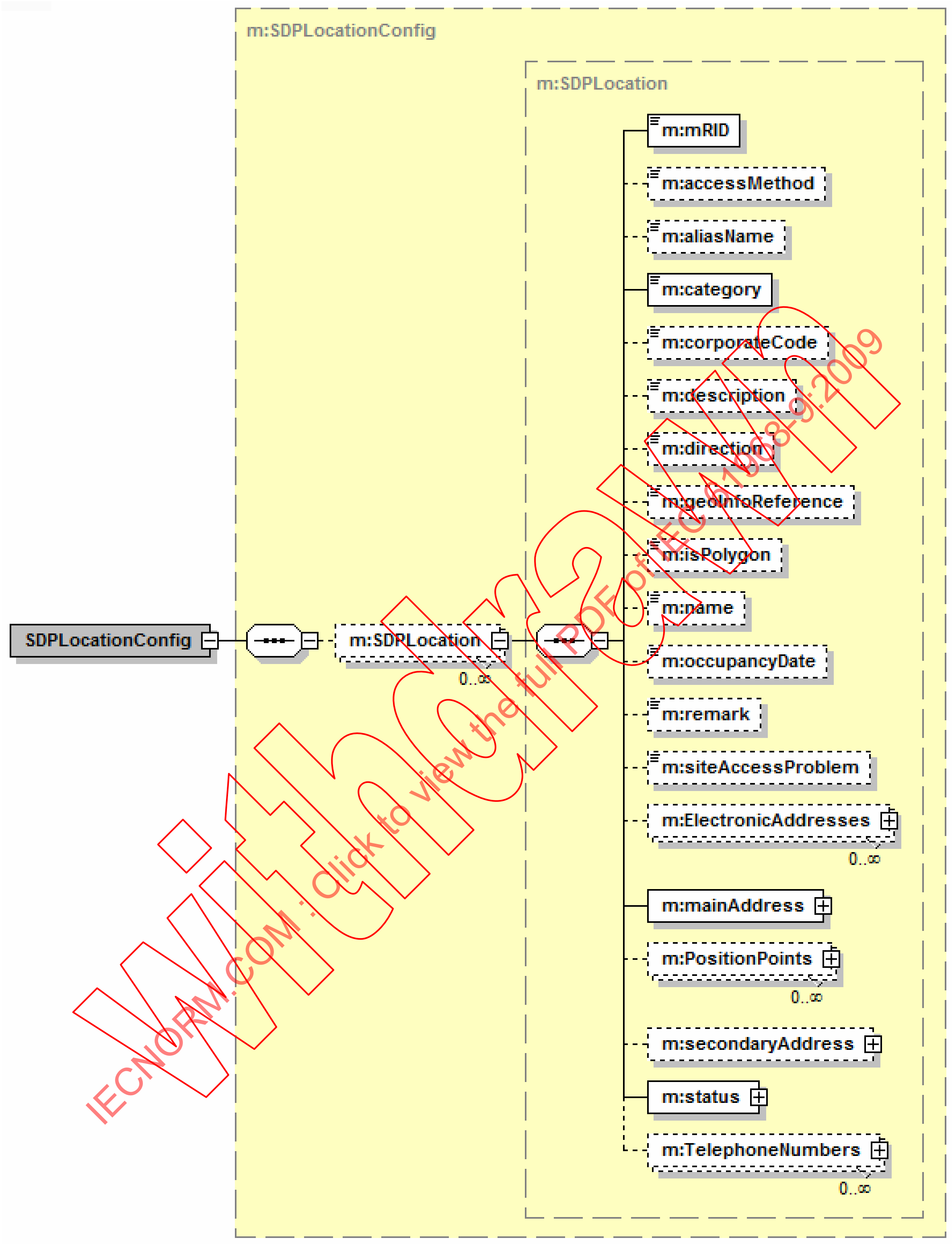


Figure 66 – Message exchange for transferring service delivery point location information



IEC 1784/09

Figure 67 – SDPLocationConfig message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.22.

5.9.2.12 Service locations

Information about the service locations may be configured into MS by means of the ServiceLocationConfig message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 68 and the message format is given in Figure 69.

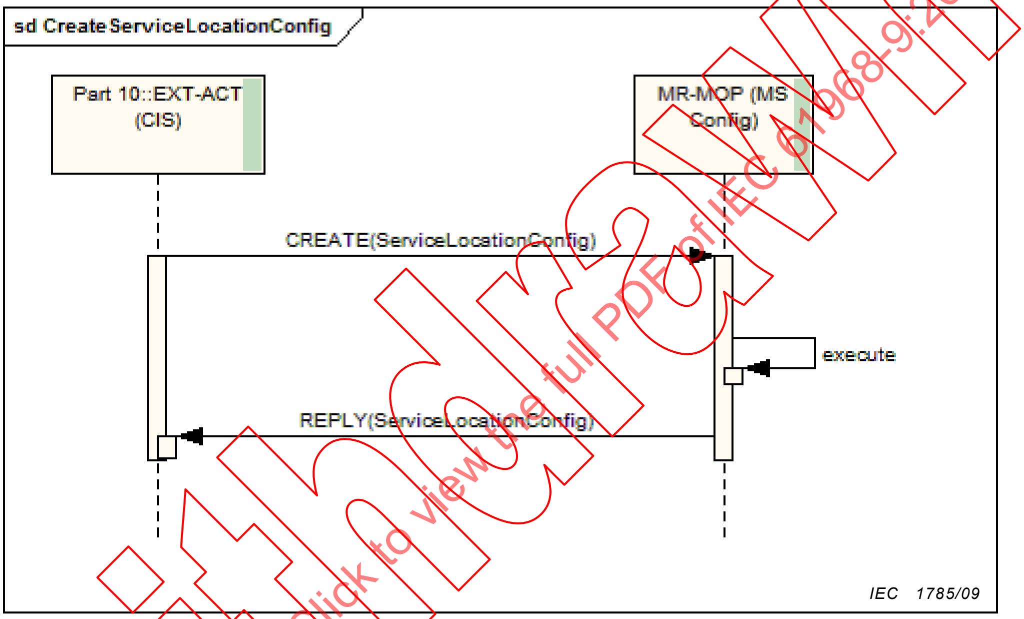

sd CreateServiceLocationConfig
Part 10::EXT-ACT (CIS)
MR-MOP (MS Config)
CREATE(ServiceLocationConfig)
execute
REPLY(ServiceLocationConfig)
IEC 1785/09

Figure 68 – Message exchange for transferring service location information

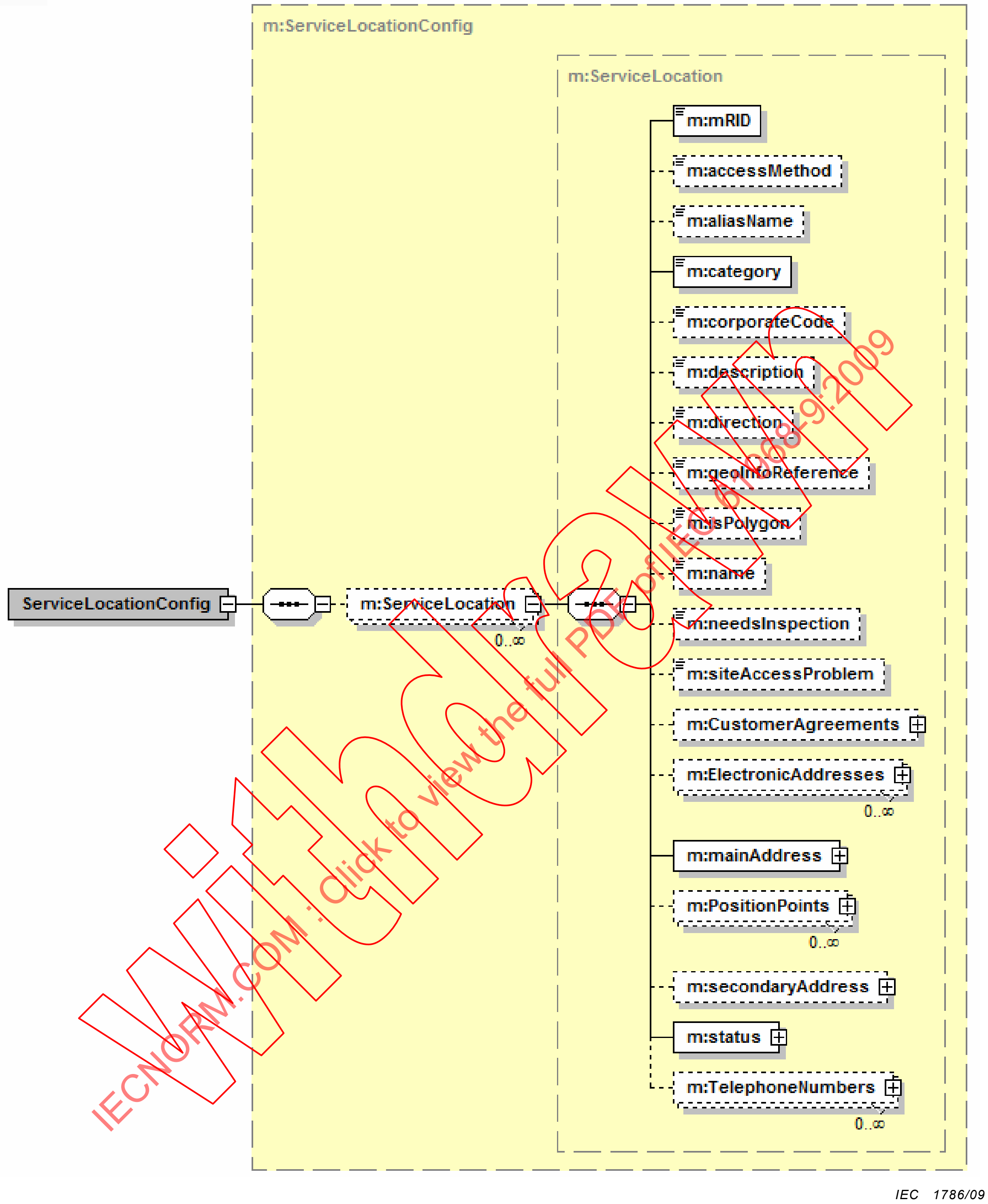


Figure 69 – ServiceLocationConfig message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.21.

5.9.3 MS financial information

5.9.3.1 General

When a customer purchases a prepaid token at a point of sale or makes a payment on a billed account, financial information is originated in MS relating to this transaction. Typically, this information is periodically transferred to the CIS and billing system for further processing.

The messages defined in 5.9.3 make provision for transferring this information to the CIS and billing system.

5.9.3.2 Receipts

When MS receives a payment, a receipt is generated and the relevant information about the payment is typically recorded as a receipt record. This information may be accessed by means of ReceiptRecord message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 70 and the message format is given in Figure 71.

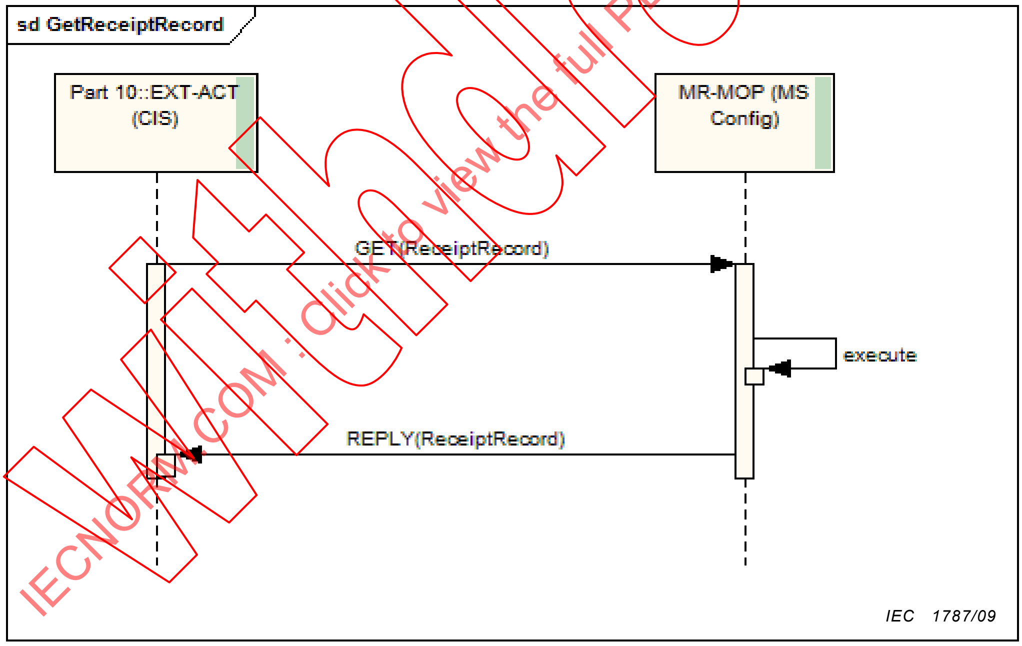
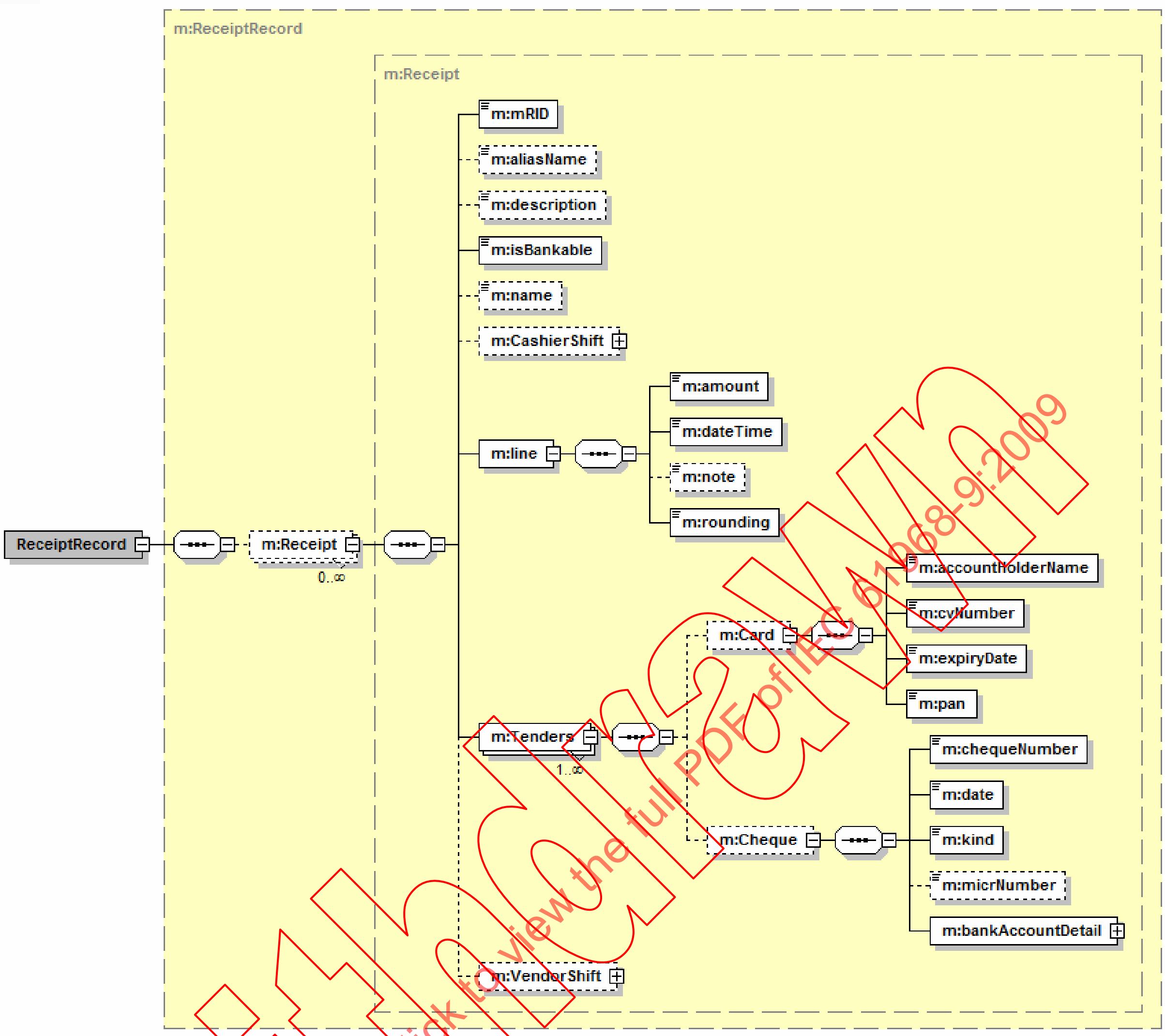


Figure 70 – Message exchange for transferring receipt information



IEC 1788/09

Figure 71 – ReceiptRecord message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.23.

5.9.3.3 Transactions

Information about the sale of a prepaid token or the receipt of an account payment is typically also recorded by MS in the form a financial transaction record, capturing the relevant details relating to the particular transaction. This information may be accessed by means of the TransactionRecord message payload.

In the case where a specified range (by date or mRID) of the message payload is to be specified in a particular contextual implementation, this may be realised by making use of the optional elements in the message header (see Figure B.2).

An example of a message exchange is given in Figure 72 and the message format is given in Figure 73.

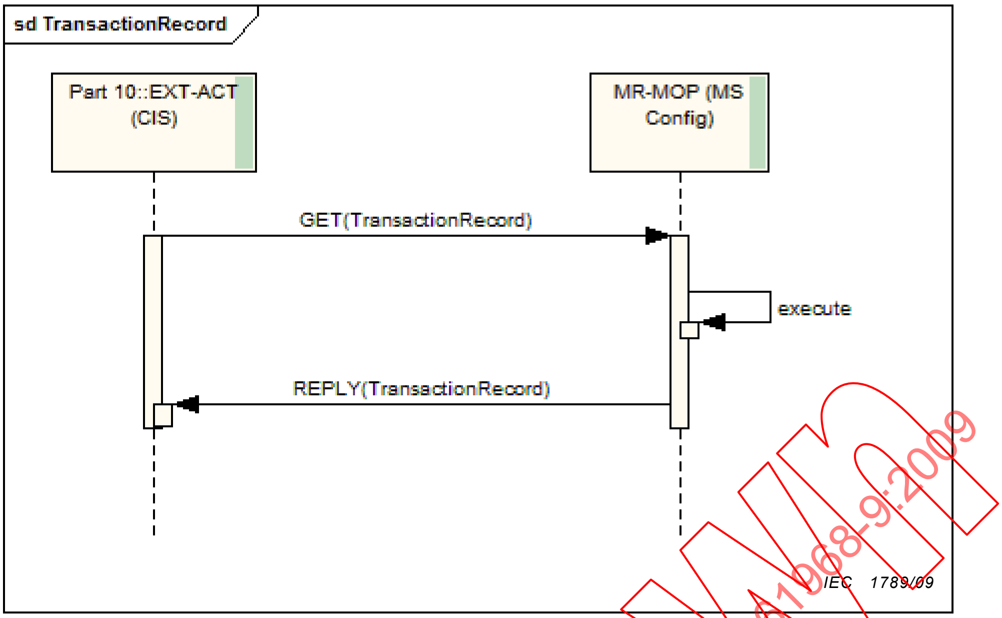
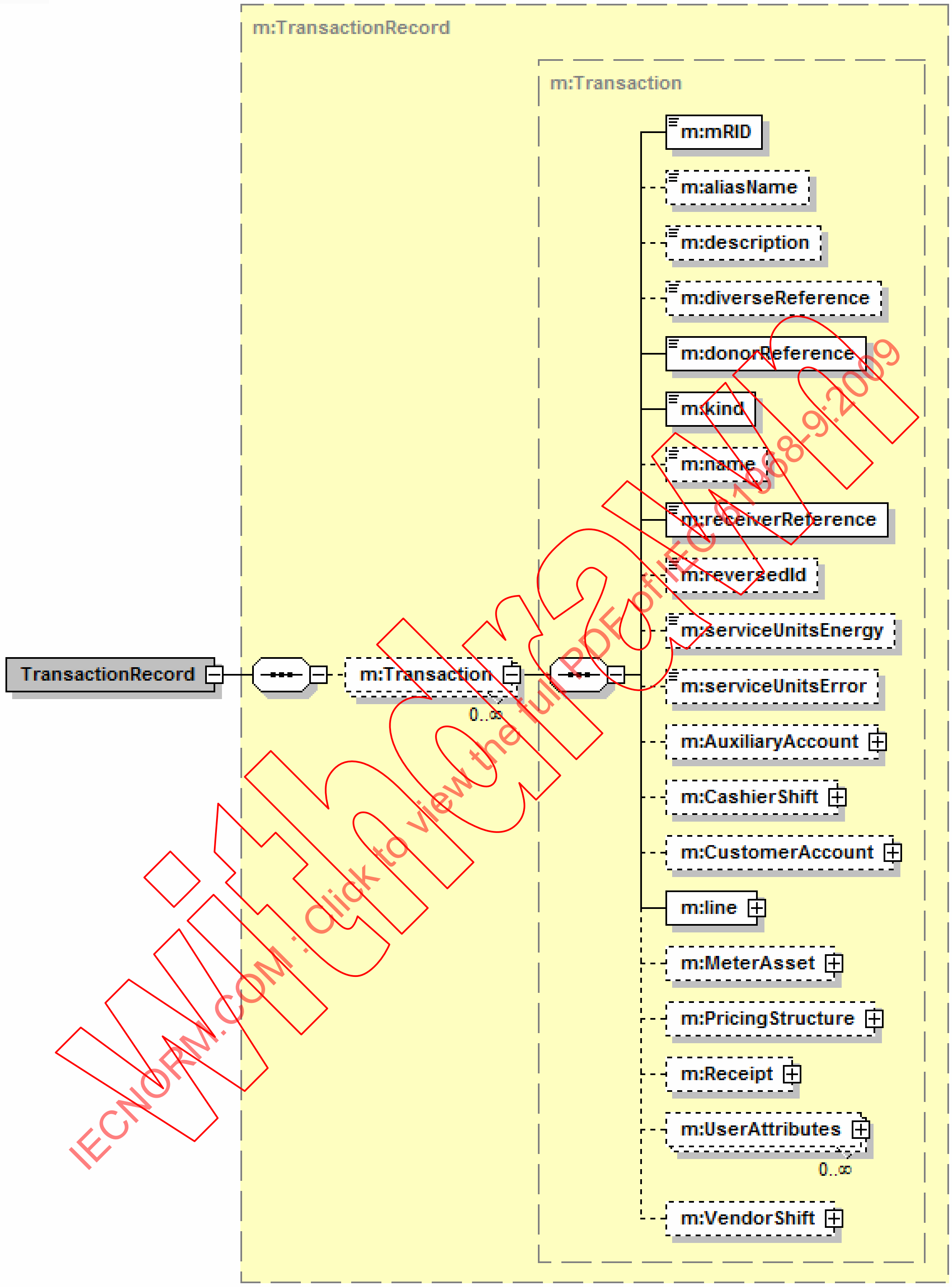


Figure 72 – Message exchange for transferring transaction information



IEC 1790/09

Figure 73 – TransactionRecord message format

The detailed description of each class and attribute are defined by Common Information Model and IEC 61970-301. The detailed, annotated XML schema is defined in Clause H.24.

6 Document conventions

6.1 UML diagrams

All UML-based sequence diagrams contained herein are to be considered as informative examples of how a message exchange could occur.

NOTE One of the strengths of the CIM is its flexibility. As technology advances, and new needs develop, new messages can be created. These new messages might involve additional systems (not pictured.) These new messages may leverage different options than the ones depicted in the example.

All UML-based communication diagrams and message flow diagrams contained herein are to be considered informative.

All UML-based class diagrams contained herein are to be considered informative. The reader is referred to IEC 61968-1 to locate the document that contains the normative definitions of the classes used in the CIM.

6.2 Message definitions

Message format diagrams contained in the body of this document are to be considered normative, with the normative XML schemas being supplied in Annex H.

Use cases and sequence diagrams presented in this document are for informative purposes only, and represent usage examples for the normative message definitions.

6.2.1 Mandatory versus optional

The messages described within this standard were derived from use cases which satisfy an underlying business need for a specific information exchange. Each use case provides a given context for the use of the CIM. Message format diagrams describe the elements which are passed. The elements depicted in dashed-line boxes are to be considered optional in a given context. The elements depicted in solid boxes are to be considered mandatory in a given context. If a diagram should depict an entire class as mandatory or optional, the reader should interpret this to mean that the use of the class is either mandatory or optional, but not that every element within the class is now mandatory or optional. The reader must refer to the normative definition of the class to determine this.

Annex A
(informative)

Description of message type verbs

Table A.1 – Commonly used verbs

Proposed verbs	Meaning	Message body
CREATE	The CREATE verb is used to submit a request to the master system to create a new document. The master system may in turn publish the new document using the verb CREATED. The master system may also use the verb REPLY to respond to the CREATE request, indicating whether the request has been processed successfully or not.	Header, Payload
UPDATE	The UPDATE verb is used to submit a request to the master system to make a change in the document based on the information in the message. The master system may in turn publish the changed document using the verb UPDATED to notify that the document has been changed since last published. The master system will use the verb REPLY to respond to the UPDATE request, indicating whether the request has been processed successfully or not.	Header, Payload
CHANGE	Synonym for UPDATE	Header, Payload
CANCEL	The CANCEL verb is used to submit a request to the master system to cancel the document. The master system may in turn publish the cancelled message using the verb CANCELED to notify that the document has been cancelled since last published. The master system will use the verb REPLY to respond to the CANCEL request, indicating whether the request has been processed successfully or not. The CANCEL verb is used when the business content of the document is no longer valid due to error(s).	Header, Request mRID(s) for objects specified using Reqest.ID
CLOSE	The CLOSE verb is used to submit a request to the master system to close the document. The master system may in turn publish the closed message using the verb CLOSED to notify that the document has been closed since last published. The master system will use the verb REPLY to respond to the CLOSE request, indicating whether the request has been processed successfully or not. The CLOSE verb is used when the business document reaches the end of its life cycle due to successful completion of a business process.	Header, Request mRID(s) for objects specified using Reqest.ID
DELETE	The DELETE verb is used to submit a request to the master system to delete the document. The master system may in turn publish the closed message using the verb DELETED to notify that the document has been deleted since last published. The master system may also use the verb REPLY to respond to the DELETE request, indicating whether the request has been processed successfully or not. The DELETE verb is used when the business document should no longer be kept in the integrated systems either due to error(s) or due to archiving needs.	Header, Request mRID(s) for objects specified using Reqest.ID
GET	The GET verb is used to submit a query request to the master system to get the current data for a given document reference code or a set of	Header, Request Variety of options to qualify request

Proposed verbs	Meaning	Message body
	documents. The master system will use the verb REPLY to respond to the GET request.	using request elements
CREATED	The CREATED verb is used to publish the creation of a document as a result of either an external request or an internal action within the master system of that document. This is the first time that data for this document reference code has been published as the result of internal or external request; in which case, it would use the same document reference as the CREATE message. This message type is usually subscribed by interested systems and could be used for mass updates. There is no need to reply to this message type.	Header, Payload
UPDATED	The UPDATED verb is used to publish events related to the change of a document as a result of either an external request or an internal action within the master system of that document. This could be a generic change in the content of the document or a specific status change such as “approved”, “issued” etc. This message type is usually subscribed by interested systems and could be used for mass updates. There is no need to reply to this message type.	Header, Payload
CHANGED	Treat as a synonym with UPDATED	Header, Payload
CLOSED	The CLOSED verb is used to publish events related to the normal closure of a document as a result of either an external request or an internal action within the master system of that document. This message type is usually subscribed by interested systems and could be used for mass updates. There is no need to reply to this message type.	Header, Payload
CANCELED	The CANCELED verb is used to publish the cancellation of a document as a result of either an external request or an internal action within the master system of that document. This message type is usually subscribed by interested systems and could be used for mass updates. There is no need to reply to this message type.	Header, Payload
DELETED	The DELETED verb is used to publish the deletion of a document as a result of either an external request or an internal action within the master system of that document. This message type is usually subscribed by interested systems and could be used for mass updates. There is no need to reply to this message type.	Header, Payload
SHOW	Synonym to REPLY, may be deprecated.	Header, Reply, Payload
REPLY	The REPLY verb is used to return the processing result of a request for a CREATE, UPDATE, DELETE, CANCEL or CLOSE.	Header, Reply, Payload (optional) Where a payload would be supplied in a response to a GET request (unless there were errors), it would not be returned for UPDATE, CANCEL, CLOSE or DELETE

Proposed verbs	Meaning	Message body
		requests.
SUBSCRIBE	The SUBSCRIBE verb is used to indicate a subscription for a type of information identified by the noun. This is realized within the integration infrastructure (e.g. JMS).	Not implemement as an application-level message
UNSUBSCRIBE	The UNSUBSCRIBE verb is used to indicate the termination of a subscription for a type of information identified by the noun. This is realized within the integration infrastructure (e.g. JMS).	Not implemement as an application-level message

It is important to note that the verbs are listed in the document are for convenience purposes and also to reflect changes that may be reflected in future versions of IEC 61968-1.

More detailed definitions of terms used in this annex are given in IEC 61968-1.

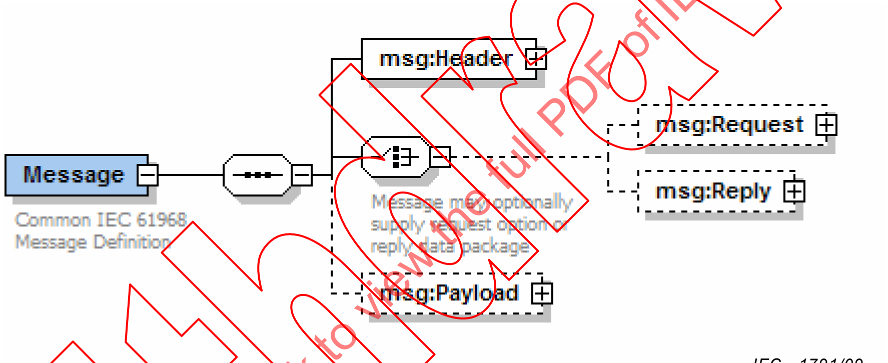
Annex B
(informative)

Recommended message structure

The purpose of this annex is to describe a recommended message structure. The message structure (see Figure B.1) has several packages for:

- header for message control information;
- payload for conveying message data;
- request for request parameters;
- reply for reply codes and errors.

The message payloads would be conveyed using the described message structure, where the message header would identify specific nouns and verbs.


msg:Header
Message
Common IEC 61968 Message Definition
msg:Request
msg:Reply
Message may optionally supply request option or reply data package
msg:Payload

IEC 1791/09

Figure B.1 – Recommended message message-format

The structure of the header itself is described by Figure B.2.

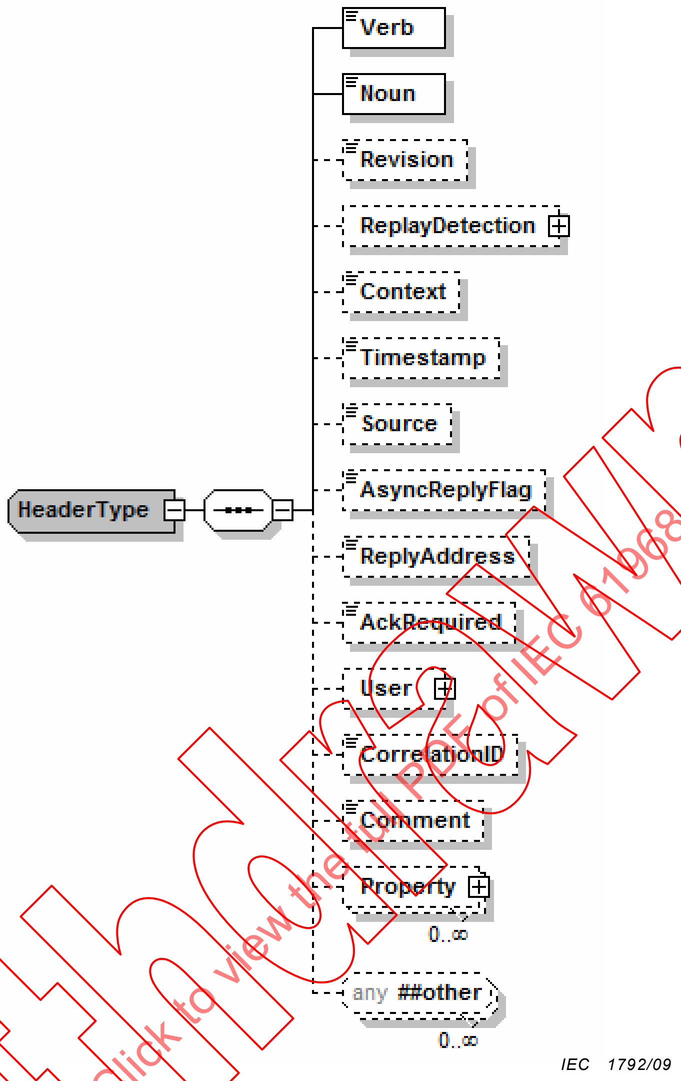


Figure B.2 – Recommended.Header message format

The message payload could be strongly typed or loosely coupled using an XML::any. The type of the payload is determined by the noun in the message header. Not all messages require payloads, as in the cases of get, close or cancel messages, as examples.

The request package of the message would provide arguments for requests. Figure B.3 contains the recommended structure of the request package:

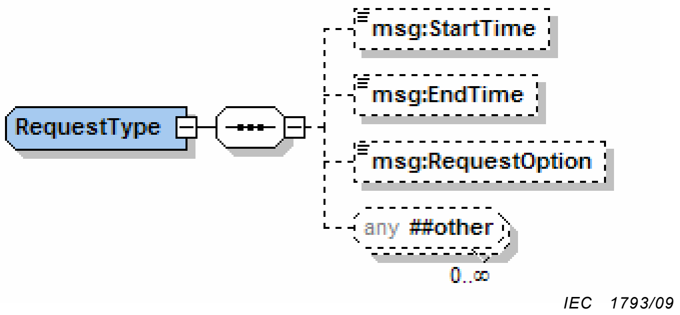


Figure B.3 – Recommended.RequestType message format

The reply package is used for returning error codes and specific errors. The recommended message format for the reply is depicted in Figure B.4:

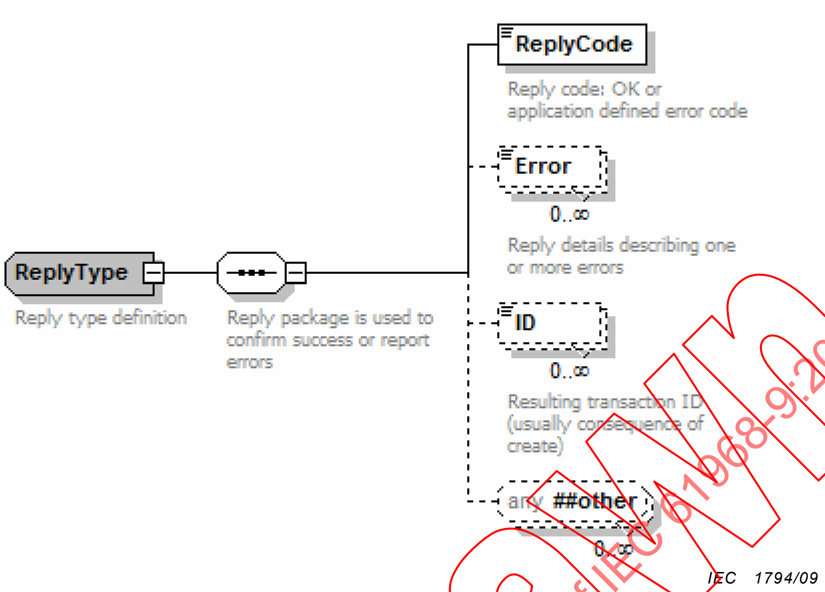

ReplyType
Reply type definition
Reply package is used to confirm success or report errors
ReplyCode
Reply code: OK or application defined error code
Error
0..∞
Reply details describing one or more errors
ID
0..∞
Resulting transaction ID (usually consequence of create)
any ##other
0..∞
IEC 1794/09

Figure B.4 – Recommended.ReplyType message format

The suggested ReplyCodes can be categorized into families as described in Table B.1:

Table B.1 – ReplyCode categories

Category	Error
1	Missing element
2	Bad value
3	Too many values
4	Request timed out
5	Application error

These categories can serve as the first three digits of a numbering scheme to form codes such as those described in Table B.2. Should additional error codes be developed over time, they can be inserted as the next value(s) within the appropriate category.

Table B.2 – ReplyCode enumerations

ReplyCode	Description
0.0	No Errors
1.1	Noun element(s) not found in CIM payload
1.2	Revision element(s) not found in CIM payload
1.3	Verb element(s) not found in CIM payload
2.1	Invalid DSI meter number(s)
2.2	Invalid feeder name
2.3	Invalid feeder number(s) and substation name combination
2.4	Invalid meter number(s)
2.5	Invalid noun value
2.6	Invalid ReadingTypeID
2.7	Invalid substation name
2.8	Invalid switch name(s) and substation name combination
2.9	Invalid verb or revision number
2.10	Unable to process the request, ReadingTypeID not supported by the system
2.11	Payload mismatch
3.1	Too many meters in request
3.2	Too many pending XML requests
3.3	Too many ReadingTypeID's in request
4.1	Request timed out
5.1	Unable to perform your request due to high system activity level
5.2	Unable to process the request. Transaction not attempted
5.3	Unable to process the request. Transaction attempted and failed

The following XML Schema describes the recommended message structures. This structure can be used by a variety of transport mechanisms including web services and JMS.

```
<?xml version="1.0" encoding="utf-8"?>
<!-- edited with XMLSpy v2007 sp1 (http://www.altova.com) by Scott Neumann (UISOL) -->
<!-- Common Message Specification for ESB Integration via IEC 61968 -->
<xs:schema xmlns="http://www.iec.ch/TC57/2008/schema/message"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" targetNamespace="http://www.iec.ch/TC57/2008/schema/message"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.0.0">
<xs:import namespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" schemaLocation="WSS200401wssecurity-secext-10.xsd"/>
<xs:import namespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd" schemaLocation="WSS200401wssecurity-utility-10.xsd"/>
<xs:complexType name="RequestType">
        <xs:sequence>
                <xs:element name="StartTime" type="xs:string" minOccurs="0">
                </xs:element>
                <xs:element name="EndTime" type="xs:string" minOccurs="0">
                </xs:element>
                <xs:element name="Option" type="xs:string" minOccurs="0">
                </xs:element>
```

```
				<xs:element name="ID" type="xs:string" minOccurs="0"
maxOccurs="unbounded">
				</xs:element>
				<xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
		</xs:sequence>
</xs:complexType>
<xs:complexType name="ReplyType">
		<xs:sequence>
		<xs:element name="ReplyCode" type="xs:string">
		</xs:element>
		<xs:element name="Error" type="xs:string" minOccurs="0" maxOccurs="unbounded">
		</xs:element>
		<xs:element name="ID" type="xs:string" minOccurs="0" maxOccurs="unbounded">
		</xs:element>
		<xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
		</xs:sequence>
</xs:complexType>
<xs:complexType name="PayloadType">
		<xs:sequence>
		<xs:choice>
				<xs:any namespace="##other" processContents="skip" minOccurs="0"
maxOccurs="unbounded">
				</xs:any>
				<xs:element name="Compressed" type="xs:string" minOccurs="0">
				</xs:element>
		</xs:choice>
		<xs:element name="format" type="xs:string" minOccurs="0">
		</xs:element>
		</xs:sequence>
</xs:complexType>
<xs:complexType name="ReplayDetectionType">
		<xs:sequence>
		<xs:element name="Nonce" type="wsse:EncodedString"/>
		<xs:element name="Created" type="wsu:AttributedDateTime"/>
		</xs:sequence>
</xs:complexType>
<xs:complexType name="UserType">
		<xs:sequence>
		<xs:element name="UserID" type="xs:string">
		</xs:element>
		<xs:element name="Organization" type="xs:string">
		</xs:element>
		</xs:sequence>
</xs:complexType>
<xs:complexType name="HeaderType">
		<xs:sequence>
		<xs:element name="Verb" type="xs:string">
		</xs:element>
		<xs:element name="Noun" type="xs:string">
		</xs:element>
		<xs:element name="Revision" type="xs:string" minOccurs="0">
		</xs:element>
		<xs:element name="ReplayDetection" type="ReplayDetectionType" minOccurs="0"/>
		<xs:element name="Context" minOccurs="0">
		<xs:simpleType>
				<xs:restriction base="xs:string">
				<xs:enumeration value="PRODUCTION"/>
				<xs:enumeration value="TESTING"/>
				<xs:enumeration value="DEVELOPMENT"/>
				<xs:enumeration value="STUDY"/>
```

```
                <xs:enumeration value="TRAINING"/>
                </xs:restriction>
        </xs:simpleType>
        </xs:element>
        <xs:element name="Timestamp" type="xs:dateTime" minOccurs="0">
        </xs:element>
        <xs:element name="Source" type="xs:string" minOccurs="0">
        </xs:element>
        <xs:element name="AsyncReplyFlag" type="xs:boolean" minOccurs="0">
        </xs:element>
        <xs:element name="ReplyAddress" type="xs:string" minOccurs="0">
        </xs:element>
        <xs:element name="AckRequired" type="xs:boolean" minOccurs="0">
        </xs:element>
        <xs:element name="User" type="UserType" minOccurs="0">
        </xs:element>
        <xs:element name="MessageID" type="xs:string" minOccurs="0">
        </xs:element>
        <xs:element name="CorrelationID" type="xs:string" minOccurs="0">
        </xs:element>
        <xs:element name="Comment" type="xs:string" minOccurs="0">
        </xs:element>
        <xs:element name="Property" type="MessageProperty" minOccurs="0"
maxOccurs="unbounded">
        </xs:element>
        <xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
</xs:complexType>
<xs:element name="Message" type="MessageType">
</xs:element>
<xs:complexType name="MessageProperty">
        <xs:sequence>
                <xs:element name="Name" type="xs:string"/>
                <xs:element name="Value" type="xs:string" minOccurs="0"/>
        </xs:sequence>
</xs:complexType>
<xs:element name="RequestMessage" type="RequestMessageType">
</xs:element>
<xs:element name="ResponseMessage" type="ResponseMessageType">
</xs:element>
<xs:element name="FaultMessage" type="FaultMessageType">
</xs:element>
<xs:element name="EventMessage" type="EventMessageType">
</xs:element>
<xs:complexType name="MessageType">
        <xs:sequence>
        <xs:element name="Header" type="HeaderType"/>
        <xs:element name="Request" type="RequestType" minOccurs="0"/>
        <xs:element name="Reply" type="ReplyType" minOccurs="0"/>
        <xs:element name="Payload" type="PayloadType" minOccurs="0"/>
        </xs:sequence>
</xs:complexType>
<xs:complexType name="RequestMessageType">
        <xs:sequence>
        <xs:element name="Header" type="HeaderType"/>
        <xs:element name="Request" type="RequestType" minOccurs="0"/>
        <xs:element name="Payload" type="PayloadType" minOccurs="0"/>
        </xs:sequence>
</xs:complexType>
<xs:complexType name="ResponseMessageType">
        <xs:sequence>
```

```
		<xs:element name="Header" type="HeaderType"/>
		<xs:element name="Reply" type="ReplyType"/>
		<xs:element name="Payload" type="PayloadType" minOccurs="0"/>
		</xs:sequence>
</xs:complexType>
<xs:complexType name="FaultMessageType">
		<xs:sequence>
		<xs:element name="Reply" type="ReplyType"/>
		</xs:sequence>
</xs:complexType>
<xs:complexType name="EventMessageType">
		<xs:sequence>
		<xs:element name="Header" type="HeaderType"/>
		<xs:element name="Request" type="RequestType" minOccurs="0"/>
		<xs:element name="Payload" type="PayloadType" minOccurs="0"/>
		</xs:sequence>
</xs:complexType>
</xs:schema>
```

Annex C
(informative)

Recommended procedure for the generation of a ReadingType

C.1 General

Metering systems collect readings from meters, manage configurations, operate remote devices, and more as new systems develop and capabilities continue to grow. AMR/AMI vendors often times must interface with a variety of meter vendors, a variety of MDM systems and/or a variety of customer information systems. Different vendors tend to use different terminology. In an effort to rise above the specific vendor names, it is common for a utility or vendor to give names to the data elements that they use. This annex describes a procedure that may be used to create data element names, and suggests that data element names may be used as the ReadingType.name. Then, having identified the attributes of a given data element, this annex also recommends a technique to populate the ReadingType.mRID attributes.

C.2 Understanding the important attributes of a data element name

C.2.1 General

Data elements can be described in terms of 11 key attributes.

C.2.2 Attribute #1, time attribute enumerations

Attribute #1 supplies an “adjective” to describe aspects of a “time attribute” with regard to the measurement. Enumerated values include the values described in Table C.1.

The user should note that for reading types which involve a period of time (such as Attribute#3=IntervalData), the specific period of time (e.g. “hourly”) is represented in the timestamp field per ISO 8601. For example, “[P1H]” represents a period of 1 h, “[P15M]” represents a period of 15 min. With the interval period broken out in a separate field, data consumers as well as data producers will need to be mindful to study the timestamp field for period information.

NOTE 1 ISO 8601 supports the ability to use the timestamp to identify the start or end of the interval by careful placement of the timestamp field relative to the period field.

NOTE 2 ISO 8601 also supports the ability to identify the number of recurring periodic intervals with the optional “R” prefix.

Attribute #1 is used to describe time attributes (of the measurement) which cannot be expressed in the extended ISO 8601 timestamp field. It can also be used to identify certain relative relational properties.

Table C.1 – Time attribute enumerations

Index	Enumeration	Comments
0		Not applicable, or the bulk quantity of a commodity (e.g. the dial reading on an electromechanical watt-hour meter)
1	RESERVED	Reserved for “10-minute”
2	RESERVED	Reserved for “15-minute”
3	RESERVED	Reserved for “1-minute”
4	RESERVED	Reserved for “24-hour”
5	RESERVED	Reserved for “30-minute”
6	RESERVED	Reserved for “5-minute”
7	RESERVED	Reserved for “60-minute”
8	Billing-Shifted	Billing period shifted (whatever the billing period is)
9	Block	A block of data
10	RESERVED	Reserved for “2-minute”
11	Daily-Shifted	Daily self-read
12	Instantaneous	Typically measured over tens of milliseconds
13	Monthly-Shifted	Monthly self-read
14	RESERVED	Reserved for “3-minute”
15	Present	Within the present monthly cycle
16	Previous	Within the previous monthly cycle and data set
17	Previous-Season	Within the previous season and data set
18	Q1-Shifted	First quarter
19	Q2-Shifted	Second quarter
20	Q3-Shifted	Third quarter
21	Q4-Shifted	Fourth quarter
22	Seasonal	E.g. “Summer, Spring, Fall, and Winter” based cycle
23	Sub-block	A portion of a block of data
24	Weekly-shifted	Weekly self read
25	Expiration	The time at which an action is scheduled to expire
26	Has	“Exists” or “is”
27	Last	Of all the previous readings, this one is the most recent chronologically
28	Next	Of any future (pending) events, this one is next
29	Requires	“Must” or “Mandates”
30	Start	The time at which an action is scheduled to start

C.2.3 Attribute #2, data qualifier

The data type can be used to describe a salient attribute of the data. Possible enumerations include those described in Table C.2.

Table C.2 – Data qualifier enumerations

Index	Enumeration	Comments
0		Not applicable
1	Absolute	
2	Average	
3	Compensated	
4	Excess	
5	HighThreshold	
6	Incremental	
7	LowThreshold	
8	Maximum	
9	Minimum	
10	Missing	
11	Nominal	
12	Normal	
13	Relative	
14	Required	
15	RMS	Root mean squared
16	SecondMaximum	
17	SecondMinimum	
18	SurveyData	Data which should serve as a representative sample of the population
19	TestData	Not real billing data
20	Uncompensated	
21	Unstable	
22	Action	

C.2.4 Attribute #3, accumulation behaviour

The “accumulation behaviour” indicates how the value is represented to accumulate over time. Enumerated values include those described in Table C.3.

One benefit to explicitly declaring the accumulation behaviour as part of the readingType in an exchange between a MS and MDM is to help the MDM ensure that appropriate validation rules are applied to the data supplied by the MS.

Table C.3 – Accumulation behaviour enumerations

Index	Enumeration	Comments
0		Not applicable, or implied by the unit of measure.
1	BulkQuantity	A value from a register which represents the bulk quantity of a commodity. This quantity is computed as the integral of the commodity usage rate. This value is used as the basis for the dial reading at the meter, and as a result, may roll over upon reaching a maximum dial value. NOTE This enumeration is usually implied by the context. Users may elect to omit phrases implied by their context from a textual description (e.g. the “alias name”), but choose to retain them in the numeric description (e.g. the “name”).
2	ContinuousCumulative	The sum of the previous billing period values and the present period value.
3	Cumulative	The sum of the previous billing period values.
4	DeltaData	The "consumption" or "usage" of a commodity between two distinct points in time. Or, alternatively, the quantity consumed over a specific period of time (with "zero" at the start of the measurement). The period over which the begin and endpoints occur may be non-recurring or irregular (such as a billing period which changes size with the number of days in a month). Further context is provided by the TimeAttribute, Flow direction, and the time-stamp(s). (See also "Interval data" for a related concept).
6	Indicating	As if a needle swung out to a value to indicate the current value.
7	IntervalData	The "delta" difference between two values at regularly spaced intervals in time. The use of "IntervalData" implies that the datum is part of a consecutive series of data sampled at uniform intervals of time.
8	RecordedData	A series of data samples accompanied by timestamp(s).
9	Summation	A form of accumulation which may be selective with respect to time. NOTE This enumeration is usually implied by the unit of measure.
10	TimeDelay	A form of computation which introduces a time delay characteristic to the data value.

It should be noted that the formulas for interval data are specific, and often qualified as “incremental” or “absolute”. (The “Data Qualifier” attribute can be used for this purpose.)

- Incremental interval data (a “delta” or “interval” difference between two samples captured at regularly spaced intervals in time.)

$$\text{Data}_I = \frac{\text{Sample}_I}{\text{Scalar}} - \frac{\text{Sample}_{I-1}}{\text{Scalar}}$$

- Relative interval data (the difference between a scaled sample and a reference value.)

$$\text{Data}_I = \frac{\text{Sample}_I}{\text{Scalar}} - \text{ReferenceValue}$$

- Absolute recorded data (a sample accompanied by a timestamp.)

$$\mathrm{Data}_I = \frac{\mathrm{Sample}_I}{\mathrm{Scalar}}$$

C.2.5 Attribute #4, direction of flow

Anything involving current might have a flow direction. Possible enumerations include those described in Table C.4.

Table C.4 – Direction of flow enumerations

Index	Enumeration	Comments
0		N/A
1	Forward	"Delivered," or "Import", see IEC 61968-2
2	Lagging	Typically used to describe that a power factor or reactive energy component is lagging the reference value.
3	Leading	Typically used to describe that a power factor or reactive energy component is leading the reference value.
4	Net	|Forward| - |Reverse|, see IEC 61968-2
5	Q1+Q2	Reactive positive quadrants
6	Q1+Q2+Q3+Q4	Sum of all quadrants
7	Q1+Q3	
8	Q1+Q4	Quadrants 1 and 4
9	Q1-Q4	Q1 minus Q4
10	Q2+Q3	Quadrants 2 and 3
11	Q2+Q4	
12	Q2-Q3	Q2 minus Q3
13	Q3+Q4	Reactive negative quadrants
14	Q3-Q2	Q3 minus Q2
15	Quadrant1	
16	Quadrant2	
17	Quadrant3	
18	Quadrant4	
19	Reverse	"Received," or "Export", see IEC 61968-2
20	Total	|Forward| + |Reverse|, see IEC 61968-2

C.2.6 Attribute #5, unit of measure category

The categorization scheme described here breaks the unit of measure into separate classes. One class collects all types of meter readings, another meter system parameters, and yet another class for event related measurements. Categories include those described in Table C.5.

Table C.5 – Unit of measure enumerations

Attribute 5.1 -- Subclass	Attribute 5.2 -- Index	Enumeration	Comments
0	0		Not applicable
0	1	Air	
0	2	ApparentPowerFactor	
0	3	Currency	
0	4	Current	
0	5	CurrentAngle	
0	6	CurrentImbalance	
0	7	Date	
0	8	Demand	
0	9	Distance	
0	10	DistortionVoltAmperes	
0	11	Energization	
0	12	Energy	
0	13	EnergizationLoadSide	
0	14	Fan	
0	15	Frequency	
0	16	Funds	
0	17	IEEE1366ASAI	
0	18	IEEE1366ASIDI	
0	19	IEEE1366ASIFI	
0	20	IEEE1366CAIDI	
0	21	IEEE1366CAIFI	
0	22	IEEE1366CEMIn	
0	23	IEEE1366CEMSMIn	
0	24	IEEE1366CTAIDI	
0	25	IEEE1366MAIFI	
0	26	IEEE1366MAIFIe	
0	27	IEEE1366SAIDI	
0	28	IEEE1366SAIFI	
0	29	InsulativeGas	
0	30	InsulativeOil	
0	31	LineLosses	
0	32	Losses	
0	33	NegativeSequence	
0	34	PhasorPowerFactor	
0	35	PhasorReactivePower	
0	36	PositiveSequence	
0	37	Power	
0	38	PowerFactor	
0	39	Price	
0	40	QuantityPower	

Attribute 5.1 -- Subclass	Attribute 5.2 -- Index	Enumeration	Comments
0	41	Sag	Or voltage dip
0	42	Swell	
0	43	SwitchPosition	
0	44	TapPosition	
0	45	TariffRate	DR
0	46	Temperature	
0	47	TotalHarmonicDistortion	
0	48	TransformerLosses	
0	49	UNIPEDEVoltageDip10to15	
0	50	UNIPEDEVoltageDip15to30	
0	51	UNIPEDEVoltageDip30to60	
0	52	UNIPEDEVoltageDip60to90	
0	53	UNIPEDEVoltageDip90to100	
0	54	Voltage	
0	55	VoltageAngle	
0	56	VoltageExcursion	
0	57	VoltageImbalance	
0	58	Volume	
0	59	ZeroFlowDuration	
0	60	ZeroSequence	
1	0	ApplicationContext	
1	1	ApTitle	
1	2	AssetNumber	
1	3	Bandwidth	
1	4	BatteryVoltage	
1	5	BroadcastAddress	
1	6	DeviceAddressType1	
1	7	DeviceAddressType2	
1	8	DeviceAddressType3	
1	9	DeviceAddressType4	
1	10	DeviceClass	
1	11	ElectronicSerialNumber	
1	12	EndDeviceID	
1	13	GroupAddressType1	
1	14	GroupAddressType2	
1	15	GroupAddressType3	
1	16	GroupAddressType4	
1	17	IpAddress	
1	18	MacAddress	
1	19	MfgAssignedConfigurationID	
1	20	MfgAssignedPhysicalSerialNumber	
1	21	MfgAssignedProductNumber	
1	22	MfgAssignedUniqueCommunicationAddress	

Attribute 5.1 -- Subclass	Attribute 5.2 -- Index	Enumeration	Comments
1	23	MultiCastAddress	
1	24	OneWayAddress	
1	25	SignalStrength	
1	26	TwoWayAddress	
2	0	Alarm	
2	1	BatteryCarryover	
2	2	DataOverflowAlarm	
2	3	DemandLimit	
2	4	DemandReset	
2	5	Diagnostic	
2	6	EmergencyLimit	
2	7	EncoderTamper	
2	8	IEEE1366MomentaryInterruption	
2	9	IEEE1366MomentaryInterruptionEvent	
2	10	IEEE1366SustainedInterruption	
2	11	InterruptionBehaviour	
2	12	InversionTamper	
2	13	LoadInterrupt	
2	14	LoadShed	
2	15	Maintenance	
2	16	PhysicalTamper	
2	17	PowerLossTamper	
2	18	PowerOutage	
2	19	PowerQuality	
2	20	PowerRestoration	
2	21	Programmed	
2	22	Pushbutton	
2	23	RelayActivation	
2	24	RelayCycle	
2	25	RemovalTamper	
2	26	ReprogrammingTamper	
2	27	ReverseRotationTamper	
2	30	SwitchArmed	
2	31	SwitchDisabled	
2	32	Tamper	
2	33	WatchdogTimeout	

C.2.7 Attribute #6, measurement category enumerations

The technique proposed here is to categorize cardinal and ordinal number values into families of codes as described in Table C.6.

Table C.6 – Measurement categories

Attribute 6.1 -- Category	Error
0	Harmonics
1	Interharmonics
2	Ordinal or cardinal numbering
3	Time of use (TOU) tier numbering
4	Currency code (per ISO 4217)

These categories provide context for the numbering scheme supplied by attribute 6.2. Attribute 6.2 is merely a number. Joined together, they form attribute 6 as described in Table C.7.

Table C.7 – MeasurementCategory enumerations

Attribute 6 - Categorized index	Enumeration	Comments
0.0		Not applicable
0.1	Harmonic1	Fundamental
0.2	Harmonic2	Second harmonic
0.3	Harmonic3	Third harmonic
0.4	Harmonic4	Fourth harmonic
0.5	Harmonic5	Fifth harmonic
0.6	Harmonic6	Sixth harmonic
0.7	Harmonic7	etc. (additional harmonics (up to 999) created as necessary in category=0)
1.1	Interharmonic1	Fundamental
1.2	Interharmonic2	Second interharmonic
1.3	Interharmonic3	Third interharmonic
1.4	Interharmonic4	Fourth interharmonic
1.5	Interharmonic5	Fifth interharmonic
1.6	Interharmonic6	Sixth interhamonic
1.7	Interharmonic7	etc. (additional interharmonics (up to 999) created as necessary in category=1)
2.0	n0	Zeroth unit (for systems that use zero based numbering schemes)
2.1	n1	First; or "1" where the measure needs an argument such as CEMI(n=1)
2.2	n2	Second; or "2" where the measure needs an argument such as CEMI(n=2)
2.3	n3	Third; or "3" where the measure needs an argument such as CEMI(n=3)
2.4	n4	Fourth; or "4" where the measure needs an argument such as CEMI(n=4)
2.5	n5	Fifth; or "5" where the measure needs an argument
2.6	n6	Sixth, or "6"
2.7	n7	Seventh, or "7"
2.8	n8	Eigth, or "8"

Attribute 6 - Categorized index	Enumeration	Comments
2.9	n9	Nineth, or "9"
2.10	n10	Tenth, or "10"
2.15	n15	"15"
2.30	n30	"30"
2.45	n45	"45"
2.60	n60	"60"
2.155	n155	"155"
2.305	n305	“305”, etc. (additional numbers up to 999 as necessary in class=2)
3.1	TouRateA	First tier
3.2	TouRateB	Second tier
3.3	TouRateC	Third tier
3.4	TouRateD	Fourth tier
3.5	TouRateE	Fifth tier
3.6	TouRateF	Sixth tier
3.7	TouRateG	etc. (additional tiers (up to 999) created as necessary in class=3)
4.392	JPY	Japanese Yen, adapted from ISO 4217 currency codes
4.710	ZAR	South African Rand
4.752	SEK	Swedish Krona
4.756	CHF	Swiss Franc
4.840	USD	US Dollar
4.978	EUR	Euro

C.2.8 Attribute #7, phase

The proposed phase enumerations are described in Table C.8.

Table C.8 – Phase enumerations

Index	Enumeration	Comments	A_1	B_1	C_1	N_1	A_2	B_2	C_2	N_2
0		Not applicable to any phase	0	0	0	0	0	0	0	0
128	Phase-A	Phase A	1							
136	Phase-AA	Phase A current relative to phase A voltage	1				1			
132	Phase-AB	Phase A to B	1					1		
225	Phase-ABC-N	ABC to neutral	1	1	1					1
193	Phase-AB-N	AB to neutral	1	1						1
129	Phase-AN	Phase A to neutral	1							1
64	Phase-B	Phase B		1						
72	Phase-BA	Phase B current relative to phase A voltage		1			1			
66	Phase-BC	Phase B to C		1					1	
97	Phase-BC-N	BC to neutral		1	1					1
65	Phase-BN	Phase B to neutral		1						1
32	Phase-C	Phase C			1					
40	Phase-CA	Phase C current relative to phase A voltage			1		1			
40	Phase-CA	Phase C to A			1		1			
41	Phase-CA-N	CA to neutral			1		1			1
33	Phase-CN	Phase C to neutral			1					1
16	Phase-N	Neutral				1				
224	Phases-ABC	Involving all phases	1	1	1					

NOTE The index values were computed by:

$$\text{Index}=A_1\times 2^7+B_1\times 2^6+C_1\times 2^5+N_1\times 2^4+A_2\times 2^3+B_2\times 2^2+C_2\times 2^1+N_2$$

When current is measured, it is often simple a single phase conductor indicated such as:

- A
- B
- C
- N

When voltage is measured, it may contain a simple description of the phase conductor, or if measured relative to something else, an indication of the measurement being “line-to-neutral” or “line-to-line” measured. Common voltage measurements include:

- A
- B
- C
- AN
- BN

- CN
- AB
- BC
- CA
- ABC

When phase angles are measured, it is usually relative to a fixed reference such as voltage phase A. The word “angle” has already been described in a previous field. This field may therefore contain an angle designation such as:

- $A_{current}A_{volts}$
- $B_{current}A_{volts}$
- $C_{current}A_{volts}$

Where the first phase listed describes the current or voltage phase being measured, and the second phase (the “A”) describes what it is being measured against (voltage phase A.) Since the fact that the measurement is an angle can be determined from the context (provided by the unit of measure), and the point of reference is always the second phase, there is no need to provide additional indications within the phase lettering. This allows the same code (such as “Phase-CA”) to be used as a line-to-line voltage descriptor, a voltage phase-angle descriptor, as well as a current phase-angle descriptor.

C.2.9 Attribute #8, metric multiplier

The CIM has developed a technique of separating the metric multiplier from the unit of measure. In an effort to avoid negative numbers as enumeration values, negative exponents are made positive by the creation of a large positive number. The proposed multiplier enumerations are described in Table C.9.

Table C.9 – Metric multiplier enumerations

Index	Enumeration	Comments
0		Not applicable or "× 1"
1	Da	Deca = × 10 = × 10^{1}
2	‘h’ or ‘%’	hecto, × 100 (or a dimensionless value expressed as a percentage) = × 10^{2}
3	K	kilo, × 1 000 = × 10^{3}
6	M	Mega, × 1 000 000 = × 10^{6}
9	G	Giga = × 1 000 000 000 = × 10^{9}
255	d	Deci = × 0,1 = × 10^{-1}
254	c	Centi = × 0,01 = × 10^{-2}
253	m	milli, × 0,001 = × 10^{-3}
250	µ	micro, × 0,000 001 = × 10^{-6}
247	n	nano, × 0,000 000 001 = × 10^{-9}

C.2.10 Attribute #9, displayable unit of measure (UOM)

The displayable unit of measure is typically indicated on the meter faceplate. Typical units of measure include those described in Table C.10.

Table C.10 – Unit of measure enumerations

Value	Quantity	Unit name	Symbol
0	N/A	none	
2	Length	meter	m
3	Mass	gram	g
4	Rotational speed	rotations per second	rev/s
5	Current	ampere	A
6	Temperature	Kelvin	K
7	Amount of substance	mole	mol
8	Luminous intensity	candela	cd
9	Plane angle	degrees	deg
10	Plane angle	radian	rad
11	Solid angle	steradian	sr
21	Absorbed dose	Gray (J/kg)	Gy
22	Radioactivity	Becquerel (l/s)	Bq
23	Relative temperature	degrees Celsius	°C
24	Dose equivalent	Sievert (J/kg)	Sv
25	Electric capacitance	Farad (C/V)	F
26	Electric charge	Coulomb (AS)	C
27	Time	second	seconds
28	Electric inductance	Henry (Wb/A)	H
29	Electric potential	Volt (W/A)	V
30	Electric resistance	Ohm (V/A)	Ω
31	Energy joule	(N m)	J
32	Force newton	(kg m/s^2)	N
33	Frequency hertz	(1/s)	Hz
34	Illuminance lux	(lm/m^2)	lx
35	Luminous flux	lumen (cd sr)	Lm
36	Magnetic flux	Weber (V s)	Wb
37	Magnetic flux density	Tesla (Wb/m^2)	T
38	Real power	Watts (I^{2}R)	W
39	Pressure	Pascal (N/m^2)	Pa
41	Area	square meter	m^2
42	Volume	cubic meter	m^3
43	Velocity	meters per second (m/s)	m/s
44	Acceleration	meters per second squared	m/s^2
45	Volumetric flow rate	cubic meters per second	m^3/s
46	Fuel efficiency	meters/cubic meter	m/m^3
47	Moment of mass	kilogram meter (kg m)	M
48	Density	kilogram/cubic meter	kg/m^3
49	Viscosity	meter squared/second	m^2/s
50	Thermal conductivity	Watt/meter Kelvin	W/m K
51	Heat capacity	Joule/Kelvin	J/K
52	Concentration	parts per million	ppm

Value	Quantity	Unit name	Symbol
53	Electric conductance	Siemens (A/V)	Siemens
54	Angular velocity	radians per second	rad/s
61	Apparent power	Volt Ampere	VA
63	Reactive power	Volt Ampere reactive (VISinθ)	VAr
64	Phase angle	degrees	θ-Deg
65	Power factor	dimensionless	Cosθ
66	Volt seconds	Volt seconds (Ws/A)	Vs
67	Volts squared	Volt squared (W^2/A^2)	V^2
68	Amp seconds	amp seconds	As
69	Amps squared	amp squared	A^2
70	Amps squared time	amp square second	A^2t
71	Apparent energy	Volt Ampere hours	VAh
72	Real energy	Watt hours	Wh
73	Reactive energy	Volt Ampere reactive hours	VArh
74	Magnetic flux	Volts per Hertz	V/Hz
75	Rate of change of frequency	hertz per second	Hz/s
76	Number of characters	characters	char
77	Data rate	characters per second	char/s
78	Turbine inertia	kg square meter	kgm^2
79	Sound pressure level	decibel	dB
80	Monetary unit	Money (NOTE Generic money unless supplemented with ISO 4217 currency codes via MeasurementCategory Enumerations).	¤
100	Quantity power	Q	Q
101	Quantity energy	Qh	Qh
102	resistivity	ρ (rho)	Ωm
103	A/m	magnetic field strength, Ampere per metre	A/m
104	volt-squared hour	Volt-squared-hours	V^2h
105	ampere-squared hour	Ampere-squared hour	A^2h
106	Ampere-hours	Ampere-hours	Ah
107	Wh/m^3	energy per volume	Wh/m^3
108	Timestamp	time and date per ISO 8601	timeStamp
109	Boolean	"1" = "true", "live", "on", "high", "set"; "0" = "false", "dead", "off", "low", "cleared"	status
110	BooleanArray	An array of boolean values describing an array or gang of switches	statuses
111	Count	counter value	Count
113	Signal Strength	receive signal strength indicator	dBm
114	ApplicationValue	encoded value	Code
115	Kh-Wh	active energy metering constant	Wh/rev
116	Kh-VArh	reactive energy metering constant	VArh/rev
117	Kh-Vah	apparent energy metering constant	VAh/rev
118	EndDeviceEvent	value to be interpreted as a EndDeviceEventCode	MeCode

C.3 Using data element attributes to define the ReadingType

C.3.1 General

The “ReadingType” may be automatically generated by judiciously populating each of the attributes and presenting them in order side-by-side. It is recommended that attributes be as explicit as possible to identify a data element, but on the other hand, not become so explicit that it overspecifies the data element.

The mRID of the ReadingTypeID is taken as authoritative by the system, and can be generated by presenting the numeric form of each attribute in side-by-side fashion separated by commas (i.e. CSV style).

<ReadingTypeID mRID> := <attribute 1>.<attribute 2>.<attribute 3>.<attribute 4>.<attribute 5>.<attribute 6>.<attribute 7>.<attribute 8>.<attribute 9>

It should be noted that two of the attributes themselves are compound fields. This result is to have a name with 11 fields:

1) TimeAttribute
2) DataQualifier
3) AccumlationBehaviour
4) FlowDirection
5) UomCategorySubclass
6) UomCategoryIndex
7) MeasurementCategory
8) Enumeration
9) Phase
10) Multiplier
11) UnitOfMeasure

This name can be regionalized as appropriate by using the regional text form for each attribute, and presenting them in order, side-by-side, separated by spaces. The example described below suggests that the metric multiplier together with the unit of measure attributes be presented within parenthesis without a space separator. This produces an effective unit of measure for the data element. In order to eliminate excessive spoken words, a practice may be established within a product of exposing a simplified aliasName or description to users. The aliasName can offer a simplified name which omits attributes which are clearly understood from the context.

<Name> := <attribute 1> <attribute 2> <attribute 3> <attribute 4> <attribute 5> <attribute 6> <attribute 7> (<attribute 8><attribute 9>)

While a good interface will eliminate ambiguity, and not publish “anonymous” data, a good interface will also not go to the other extreme and overspecify the data it publishes. The concept is similar to the practice of publishing an appropriate number of digits of resolution for a numerical value. One system may know (for example) the phase on which a voltage was measured and supply it as an attribute of the measurement. Another system might not know the phase with good authority and simply publish the voltage measured at a given meter. One system might not. The ReadingTypeIDs used by a data publisher should be appropriate for the data supplied. The ReadingTypeIDs used in a response may (therefore) differ slightly than the ReadingTypeIDs used in a request.

Examples of ReadingTypeIDs are found in Table C.11. These are merely examples, and by no means an exhaustive list of what can be constructed given the enumeration values listed above.

Table C.11 – Example ReadingTypes

Example mRID	Example Name	Example aliasName
2.6.7.1.0.12.0.0.0.3.72	15-minute Incremental IntervalData Forward Energy (kWh)	15-minute IntervalData Forward Energy (kWh)
4.6.4.0.2.8.0.0.0.0.111	24-hour Incremental DeltaData IEEE1366MomentaryInterruption (Count)	Momentary Interruption (count)
4.6.4.0.2.9.0.0.0.0.111	24-hour Incremental DeltaData IEEE1366MomentaryInterruptionEvent (Count)	Momentary Interruption Event (count)
4.6.4.0.2.10.0.0.0.0.111	24-hour Incremental DeltaData IEEE1366SustainedInterruption (Count)	Sustained Interruption (count)
4.6.4.0.2.10.0.0.0.0.27	24-hour Incremental DeltaData IEEE1366SustainedInterruption (seconds)	Sustained Interruption (seconds)
7.6.7.1.0.12.0.0.0.3.73	60-minute Incremental IntervalData Forward Energy (kVArh)	Hourly Interval Data Forward Energy (kVArh)
7.6.7.1.0.12.0.0.0.3.72	60-minute Incremental IntervalData Forward Energy (kWh)	Hourly Interval Data Forward Energy (kWh)
7.6.7.4.0.12.0.0.0.3.73	60-minute Incremental IntervalData Net Energy (kVArh)	Hourly Interval Data Net Energy (kVArh)
7.6.7.4.0.12.0.0.0.3.72	60-minute Incremental IntervalData Net Energy (kWh)	Hourly Interval Data Net Energy (kWh)
7.6.7.19.0.12.0.0.0.3.73	60-minute Incremental IntervalData Reverse Energy (kVArh)	Hourly Interval Data Reverse Energy (kVArh)
7.6.7.19.0.12.0.0.0.3.72	60-minute Incremental IntervalData Reverse Energy (kWh)	Hourly Interval Data Reverse Energy (kWh)
7.6.7.20.0.12.0.0.0.3.73	60-minute Incremental IntervalData Total Energy (kVArh)	Hourly Interval Data Total Energy (kVArh)
7.6.7.20.0.12.0.0.0.3.72	60-minute Incremental IntervalData Total Energy (kWh)	Hourly Interval Data Total Energy (kWh)
0.0.1.0.2.4.0.0.0.0.111	BulkQuantity DemandReset (Count)	DemandReset (Count)
0.0.1.0.0.11.0.0.0.0.111	BulkQuantity Energization (Count)	Energization (Count)
0.0.1.1.0.12.0.0.0.3.73	BulkQuantity Forward Energy (kVArh)	Forward Energy (kVArh)
0.0.1.1.0.12.0.0.0.3.72	BulkQuantity Forward Energy (kWh)	Forward Energy (kWh)
0.0.1.0.2.13.2.0.0.0.111	BulkQuantity LoadInterrupt n0 (Count)	LoadInterrupt (Count)
0.0.1.0.2.13.2.1.0.0.111	BulkQuantity LoadInterrupt n1 (Count)	
0.0.1.4.0.12.0.0.0.3.73	BulkQuantity Net Energy (kVArh)	Net Energy (kVArh)
0.0.1.4.0.12.0.0.0.3.72	BulkQuantity Net Energy (kWh)	Net Energy (kWh)
0.0.1.0.2.23.2.0.0.0.111	BulkQuantity RelayActivation n0 (Count)	RelayActivation (Count)
0.0.1.0.2.24.2.0.0.0.111	BulkQuantity RelayCycle n0 (Count)	
0.0.1.0.2.24.2.1.0.0.111	BulkQuantity RelayCycle n1 (Count)	
0.0.1.19.0.12.0.0.0.3.73	BulkQuantity Reverse Energy (kVArh)	Reverse Energy (kVArh)
0.0.1.19.0.12.0.0.0.3.72	BulkQuantity Reverse Energy (kWh)	Reverse Energy (kWh)
0.0.1.20.0.12.0.0.0.3.73	BulkQuantity Total Energy (kVArh)	Total Energy (kVArh)
0.0.1.20.0.12.0.0.0.3.72	BulkQuantity Total Energy (kWh)	Total Energy (kWh)
11.8.6.1.0.8.0.0.0.3.38	Daily-Shifted Maximum Indicating Forward Demand (kW)	Daily Max Demand (kW)
11.0.9.1.0.12.3.1.0.3.72	Daily-Shifted Summation Forward Energy TouRateA (kWh)	Daily Forward TOU Rate A (kWh)

Example mRID	Example Name	Example aliasName
0.0.0.0.0.011.0.0.0.0.109	Energization (status)	Energization status
0.0.6.0.0.4.0.0.0.0.5	Indicating Current (A)	Current (A)
0.0.6.0.0.4.0.0.128.0.5	Indicating Current Phase-A (A)	Current Phase A (Amps)
0.0.6.0.0.4.0.0.64.0.5	Indicating Current Phase-B (A)	Current Phase B (Amps)
0.0.6.0.0.4.0.0.32.0.5	Indicating Current Phase-C (A)	Current Phase C (Amps)
0.0.6.0.0.5.0.0.136.0.64	Indicating CurrentAngle Phase-AAv (θ-Deg)	CurrentAngle Phase-AAv (θ-Deg)
0.0.6.0.0.5.0.0.72.0.64	Indicating CurrentAngle Phase-BAv (θ-Deg)	CurrentAngle Phase-BAv (θ-Deg)
0.0.6.0.0.5.0.0.41.0.64	Indicating CurrentAngle Phase-CAv (θ-Deg)	CurrentAngle Phase-CAv (θ-Deg)
0.0.6.0.0.15.0.0.0.0.33	Indicating Frequency (Hz)	Frequency (Hz)
0.0.6.0.0.38.0.0.0.0.65	Indicating PowerFactor (Cosθ)	PowerFactor (Cosθ)
0.0.6.0.0.54.0.0.0.0.29	Indicating Voltage (V)	Voltage (V)
0.0.6.0.0.54.0.0.128.0.29	Indicating Voltage Phase-A (V)	Voltage Phase A (V)
0.0.6.0.0.54.0.0.64.0.29	Indicating Voltage Phase-B (V)	Voltage Phase B (V)
0.0.6.0.0.54.0.0.32.0.29	Indicating Voltage Phase-C (V)	Voltage Phase C (V)
0.0.6.0.0.55.0.0.72.0.64	Indicating VoltageAngle Phase-BAv (θ-Deg)	VoltageAngle Phase-BAv (θ-Deg)
0.0.6.0.0.55.0.0.41.0.64	Indicating VoltageAngle Phase-CAv (θ-Deg)	VoltageAngle Phase-CAv (θ-Deg)
12.0.6.1.0.8.0.0.0.3.38	Instantaneous Indicating Forward Demand (kW)	Instantaneous Demand (kW)
15.0.3.1.0.8.0.0.0.3.38	Present Cumulative Forward Demand (kW)	Present Cumulative Demand (kW)
15.8.6.1.0.8.0.0.0.3.38	Present Maximum Indicating Forward Demand (kW)	Max Demand (kW)
16.8.6.1.0.8.0.0.0.3.38	Previous Maximum Indicating Forward Demand (kW)	Max Demand (kW)
16.16.6.1.0.8.0.0.0.3.38	Previous SecondMaximum Indicating Forward Demand (kW)	Second Max Demand (kW)
0.0.9.20.0.12.3.1.0.3.73	Summation Total Energy TouRateA (kVArh)	Total Energy TOU Rate A (kVArh)
0.0.9.20.0.12.3.1.0.3.72	Summation Total Energy TouRateA (kWh)	Total Energy TOU Rate A (kWh)
0.0.9.20.0.12.3.2.0.3.73	Summation Total Energy TouRateB (kVArh)	Total Energy TOU Rate B (kVArh)
0.0.9.20.0.12.3.2.0.3.72	Summation Total Energy TouRateB (kWh)	Total Energy TOU Rate B (kWh)
0.0.9.20.0.12.3.3.0.3.73	Summation Total Energy TouRateC (kVArh)	Total Energy TOU Rate C (kVArh)
0.0.9.20.0.12.3.3.0.3.72	Summation Total Energy TouRateC (kWh)	Total Energy TOU Rate C (kWh)
0.0.0.0.0.43.0.0.0.0.118	SwitchPosition (MeCode)	

Annex D
(informative)

Recommended quality code enumerations

D.1 General

Quality codes may be returned with a Reading.value to indicate that there is something remarkable about the data. This annex describes proposed enumerations for quality codes to be used in this context.

The presence of a quality code is optional. When a value of zero is supplied, it is taken to mean that a system is explicitly declaring that the data is valid. When an end device which produces the data (a meter) declares data to be valid, it is saying that no remarkable events occurred during the collection of the data. When a meter data management system says that the data is "valid", it implies that the value has satisfied all of the appropriate validation rules.

The quality code values defined by this clause are conveyed within a message using the ReadingQuality.quality element.

D.2 Identification of the system offering commentary on the data

Conceivably, any piece of equipment which touches the data could introduce error, or offer remarks regarding the veracity of the data. For example, a meter might record interval data, and offer this data to the data collection system, but in doing so also indicate that something remarkable (such as a power outage or clock adjustment) occurred during some of the intervals. A meter data collection system might obtain readings from the field, but perform a sanity check on the values to find that some of them are unreasonable. A meter data management system, armed with a knowledge of the account history and usage patterns might scrutinize the data to find that certain values are suspect or possibly suggest that tampering may have occurred. All of these systems need an opportunity to comment on the data without (necessarily) changing it. When a system does change the data, this too can be noted in the quality codes.

This annex proposes using a three digit code as a prefix to the actual quality code. This is described in Table D.1.

Table D.1 – Example System IdentifiersValue

Example System IdentifiersValue	Issuer of code
0	Not applicable, or unidentified
1	End device
2	Metering system (data collection) network
3	Meter data management system

D.3 Quality code categorization

Quality codes can be categorized. Table D.2 describes the categories proposed by this annex.

Table D.2 – Example quality code categories

Category	Description
1	Diagnostics related
2	Power quality related issues at the data collection point
3	Tamper/Revenue protection related
4	Data collection related
5	Failed reasonability testing
6	Failed validation testing
7	Edited
8	Estimated
9	Oscillatory
10	Questionable

D.4 Quality code index

When these categories are applied to a particular index of codes, it results in examples such as those in Table D.3.

Table D.3 – Example categorized codes

Categorized code	Enumeration	Comments
Null	AssumedValid	Missing or empty quality code tags
0.0	DataValid	The system which declares the data as "valid" is indicated by the origin code.
1.0	DiagnosticsFlag	Look for related EndDeviceEvent(s)
1.1	BatteryLow	
1.3	SensorFailure	
1.4	WatchdogFlag	
2.0	PowerQualityFlag	Look for related MeterEvent(s)
2.1	ExcessiveOutageCount	Power outage threshold count exceeded
2.2	PqCounter	Power quality threshold count exceeded
2.3	ServiceDisconnectSwitching	Service disconnect operation during data collection interval
2.32	PowerFail	Power fail occurred during or at the start of this interval. According to ANSI C12.19, if the power is out for the entire interval, the data value presented should be zero.
3.0	RevenueProtection	Tamper/Revenue protection suspect flag Look for related EndDeviceEvent(s)
3.1	CoverOpened	
3.2	LogicalDisconnect	
3.3	RevenueProtectionSuspect	
3.4	ReverseRotation	
3.5	StaticDataFlag	
4.0	AlarmFlag	
4.1	OverflowConditionDetected	Numeric overflow condition detected in the meter

Categorized code	Enumeration	Comments
4.2	PartialInterval	Partial interval recorded in the meter due to clock change, power outage, or some similar event
4.3	LongInterval	Long interval recorded in the meter due to a clock change or some similar event
4.4	SkippedInterval	Skipped interval by the meter due to a clock change or similar event.
4.5	TestData	
4.6	ConfigurationChanged	Configuration changed during this interval
4.7	NotRecording	Load profile recording stopped
4.16	DstInEffect	Daylight saving time in effect during or at start of this interval
4.64	ClockSetForward	Clock set forward during or at the start of this interval. The interval may be short.
4.128	ClockSetBackward	Clock set backward during or at the start of this interval. The interval may be long.
4.259	DstChangeOccurred	Data has been adversely affected by a change in daylight saving time. Timestamps and/or data values may be incorrect. NOTE Compared to the ANSI C12.19 approach in which the DST flag is set for the entire Summer, this flag is only set for the affected data when the DST change occurs in the Spring and Fall.
5.256	DataOutsideExpectedRange	Presented data was identified as "invalid" by the originating system.
5.257	ErrorCode	Data was identified as "invalid" by the originating system and replaced with a special coded value to indicate an error.
5.258	Suspect	Data should be scrutinized and considered suspect
5.259	KnownMissingRead	
6.0	Failed generic validation	
6.1	Failed validation rule #1	It is recommended that the validation rule numbers be assigned regional prefixes as necessary. For example, USCAISO1, AUS1, etc.
6.2	Failed validation rule #2	
6.3	Failed validation rule #3	
6.4	Failed validation rule #4	
6.5	Failed validation rule #5	
6.6	Failed validation rule #6	
6.7	Failed validation rule #7	
6.8	Failed validation rule #8	
6.9	Failed validation rule #9	
6.10	Failed validation rule #10	Etc.
7.0	Manually edited	Generic
7.1	Edited 1	Edited using approach #1
7.2	Edited 2	Edited using approach #2
7.3	Edited 3	Etc.
8.0	Generic Estimation	
8.1	Estimated 1	Estimated value using approach #1
8.2	Estimated 2	Estimated value using approach #2
8.3	Estimated 3	Etc.

D.5 Categorized quality codes with originator identifier

When the concepts described above are combined, it can result in a code that fully describes the originating system, along with the categorized quality code.

ReadingQuality.quality := <Originating System ID>.<Code Category>.<Code Index>

Examples of the resulting quality code are described in Table D.4.

Table D.4 – Example quality codes

QualityCode	Origin		Code	
Null	N/A	0	AssumedValid	Null
1.0.0	EndDevice	1	DataValid	0.0
2.0.0	MsNetwork	2	DataValid	0.0
3.0.0	MDM	3	DataValid	0.0
3.8.0	MDM	3	Generic estimation	8.0
3.6.1	MDM	3	Failed validation rule #1	6.1
1.4.5	EndDevice	1	TestData	4.5
1.5.257	EndDevice	1	ErrorCode	5.257
2.5.256	MsNetwork	2	DataOutsideExpectedRange	5.256
1.4.5	EndDevice	1	TestData	4.5
2.3.4	MsNetwork	2	ReverseRotation	3.4
1.1.4	EndDevice	1	WatchdogFlag	1.4
3.3.5	MDM	3	StaticDataFlag	3.5
2.3.5	MsNetwork	2	StaticDataFlag	3.5
1.5.257	EndDevice	1	ErrorCode	5.257
2.4.259	MsNetwork	2	DstChangeOccurred	4.259
1.4.16	EndDevice	1	DstInEffect	4.16
1.4.1	EndDevice	1	OverflowConditionDetected	4.1
1.4.2	EndDevice	1	PartialInterval	4.2
1.4.3	EndDevice	1	LongInterval	4.3
1.4.4	EndDevice	1	SkippedInterval	4.4
3.8.1	MDM	3	Estimated value using approach #1	8.1
3.7.0	MDM	3	Manually edited (generic)	7.0
3.7.2	MDM	3	Edited using approach #2	7.2
3.3.3	MDM	3	RevenueProtectionSuspect	3.3

Annex E
(informative)

Recommended EndDeviceEvent category enumerations

E.1 General

Metering systems collect not only readings from meters, but must also report exceptions. This annex describes recommended codes to be used for this function. The reader should note that IEC 61968-9 supports the cross-pollenation of ReadingTypeID, EndDeviceEvent category, and EndDeviceControl type codes. A MeterReading may report an EndDeviceEvent by identifying in the unit of measure that the value should be interpreted as an EndDeviceEvent category. Likewise, there are many similarities between reported meter events and commands. An EndDeviceEvent which reports “Switch Open” can be issued as a command to “Open Switch.”

NOTE A code of zero is recommended as a placeholder to indicate that “no event” has happened.

Codes for EndDeviceEvent.category values, as proposed by this annex, are categorized in a manner that divides the code into 4 parts:

EndDeviceEvent.category := <domain>.<domain part>.<type>.<index>

E.2 EndDeviceEvent domains

At the highest level, it is helpful to categorize event codes by the domain that they occur in.

Example domain codes are described in Table E.1.

Table E.1 – Example EndDeviceEvent domain codes

Domain value	Enumeration
1	Communications
2	Device asset
3	Electric meter
4	Gas meter
5	Gateway
6	Grid power
7	Security
8	Transformer
9	Turbine

E.3 EndDeviceEvent domain parts

Domains can be sub-categorized into parts. Example domain parts are described in Table E.2.

Table E.2 – Example EndDeviceEvent domain part codes

Domain part	Enumeration
1	Access
2	Battery
3	Cartridge
4	Circuit
5	Control
6	Current
7	Customer interface
8	Demand
9	DER switch
10	Enclosure
11	Firmware
12	Frequency
13	Gas metrology
14	Gateway device
15	Load control
16	Load profile
17	Logs
18	Memory
19	Messaging
20	Meter asset
21	Metrology
22	Mode
23	Network interface
24	Password
25	Phase
26	Power
27	Power factor
28	Power quality
29	Pressure
30	Protective device
31	RCD switch
32	Security key
33	Tamper
34	Tariff
35	Temperature
36	Time
37	Transformer device
38	Voltage

E.4 EndDeviceEvent types

EndDeviceEvent codes can be further subdivided into “types”. Example types are described in Table E.3.

Table E.3 – Example EndDeviceEvent type codes

Type value	Enumeration
1	Alarm
2	Alarm management
3	Attribute
4	Calibration
5	Check status
6	Command
7	Configuration
8	Credit
9	End alarm
10	Identity
11	Maint mode
12	Metering mode
13	Outage
14	Quality flag
15	Read type
16	Setting
17	Status
18	Status check
19	Test mode
20	Requires
21	Has feature

E.5 EndDeviceEvent index

The most specific part of the EndDeviceEvent number is its index value. Example EndDeviceEvent indexes are presented in Table E.4.

Table E.4 – Example EndDeviceEvent indexes

Index	Enumeration
1	Abort
2	Access attempt
3	Acknowledged
4	Activated
5	Adjusted
6	Alarm cleared
7	Allocation error
8	Almost full
9	Alter option
10	Alternate phone
11	Armed for closure
12	Armed for open
13	Association failure

Index	Enumeration
14	Auto registration
15	Auto registration confirm
16	Battery charged
17	Battery failure
18	Blank
19	BTU alarm
20	Cable failed
21	Calculated
22	Call-in failure
23	Certificate expired
24	Changed
25	Changed by condition
26	Changed ID
27	Change-out
28	Cleared
29	Clock failed
30	Code changed
31	Cold start
32	Comm failed
33	Comm reestablished
34	Comm terminated
35	Command accepted
36	Command error
37	Command format error
38	Commanded change
39	Concentration
40	Configuration error
41	Connect failed
42	Connected
43	Corrupt data
44	Cover removed
45	Cross-phase
46	Cross-phase cleared
47	Current loss
48	Customer alarm
49	Customer configuration
50	Daily rate change
51	DAP failed
52	Data error
53	Date changed
54	Day change error
55	Day limit
56	Daylight time
57	Decreased

Index	Enumeration
58	Default
59	Demand limiting disabled, formula #
60	Demand limiting enabled, formula #
61	Demand reset
62	Demand reset fail
63	DER connected
64	DER disconnected
65	Deviated
66	Disabled
67	Disconnect failed
68	Disconnected
69	Display error
70	Door closed
71	Door open
72	Download status
73	DSP failed
74	Duplicate
75	Emergency increase
76	Enabled
77	Encoder
78	EPROM failure
79	Error
80	Estimated
81	Event log cleared
82	Event log updated
83	Event log uploaded
84	EWM failure
85	Failed
86	Firmware failure
87	FPV unreasonable
88	Frozen
89	GC failure
90	Health OK
91	High distortion
92	High distortion cleared
93	High limit
94	High neutral current
95	History cleared
96	History updated
97	Holiday
98	Imbalance
99	Imbalance cleared
100	Inactive
101	Inactive cleared

Index	Enumeration
102	Increased
103	Initial read
104	Input error
105	Installed
106	Invalid
107	Invalid attribute
108	Invalid blank name
109	Invalid command
110	Invalid date
111	Invalid delete
112	Invalid duplicate name
113	Invalid gateway ID
114	Invalid key
115	Invalid null
116	Invalid numeric
117	Invalid parameter
118	Invalid priority
119	Invalid return
120	Invalid search
121	Invalid serial number
122	Invalid sigmatic
123	Invalid time
124	Invalid time zone
125	Invalid timeout
126	Invalid type
127	Invalid value
128	Invalid version
129	Invalid XML
130	IO mismatch
131	Key disabled
132	Key enabled
133	Key unlocked
134	KYZ failure
135	Last read
136	Last read updated
137	Leading kvarh
138	Leading kvarh cleared
139	Limit exceeded
140	List cleared
141	List pointers reset
142	List pointers updated
143	Load energized
144	Load not-energized
145	Local access

Index	Enumeration
146	Log error
147	Login failed
148	Login failed
149	Low battery detected
150	Low limit
151	Low modem battery
152	Magnetic
153	Measured
154	Memory error
155	Message error
156	Message failed
157	Meter comm failed
158	Minutes
159	Mismatch
160	Missing
161	Modem change-out
162	Modem failed
163	Modem intialzed
164	MOL out of range
165	Momentary
166	Momentary events
167	New time
168	No modem
169	Node failed
170	Node link failed
171	Not initialized
172	Not armed
173	NVRAM failure
174	Old time
175	On-demand
176	Option board error
177	Overflow
178	Overload
179	Parameter change
180	Partial success
181	Pending table activation
182	Pending table clear
183	Phase restored
184	Physical
185	Power out
186	PQ event ended
187	PQ event occurred
188	PQ event started
189	Prepayment change

Index	Enumeration
190	Preset
191	Procedure invoked
192	Program changed
193	Program error
194	Pulse failure
195	Pushbutton pressed
196	Questionable
197	Queue
198	Radio failed
199	RAM failure
200	RAM full
201	Rate change
202	Read access
203	Read missing
204	Read secure register
205	Read secure table
206	Recoder access
207	Recording disabled
208	Recording enabled
209	Register error
210	Register failed
211	Remote access
212	Removed
213	Reprogrammed
214	Reset
215	Reset error
216	Restored
217	Results pending
218	Returned within limit
219	Reverse rotation
220	ROM failure
221	RTP activated
222	RTP deactivated
223	Sag
224	Sag cleared
225	Scheduled
226	Scheduled change
227	Sealed
228	Season changed
229	Seconds
230	Security event
231	Self read
232	Self-test error
233	Sequence error

Index	Enumeration
234	Session interrupted
235	Session too long
236	Set
237	Setup disabled
238	Setup enabled
239	Special schedule
240	Specific gravity
241	Standard time
242	Started
243	Stopped
244	Success
245	Survey failed
246	Sustained
247	Sustained events
248	Swell
249	Swell cleared
250	Synch disabled
251	Synch enabled
252	Synch failed
253	Synched
254	Synchronized
255	Table full
256	Table written
257	Tamper detected
258	Test button pushed
259	Test call
260	Test mode
261	Threshold
262	Tier changed
263	Tilt
264	Time changed
265	Total events
266	TOU mismatch
267	Turbine pulse error
268	Unprogrammed
269	Unsealed
270	Unstable
271	Unsupported
272	Upgrade pending
273	Voltage A inactive
274	Voltage B inactive
275	Voltage C inactive
276	Voltage loss
277	Volume calc error

Index	Enumeration
278	Warm start
279	Watchdog failed
280	Watchdog reset
281	Week rate change
282	Write access
283	Demand limit evaluation time-period, formula #
284	Demand limit exceeded, formula #
285	Demand limit off-time, formula #
286	Demand limit threshold, formula #
287	Demand limit warning, formula #
288	Power limit
289	Power limit expiration time-perioid

E.6 Numbered values

Certain attributes require further specification of a particular channel number, alarm number, relay number, port number, formula number, etc.

These can be provided in the code as appropriate in a separate field.

E.7 Example EndDeviceEvent codes

When the domain, domain part, type, and index fields are combined and presented as a single string, yet consists of four comma separated codes. Examples are presented in Table E.5.

Table E.5 – Example EndDeviceEvent codes

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
7.1.1.2.	Security	Access	Alarm	Access attempt	Unauthorized access attempted
3.11.2.3.1	Electric meter	Firmware	Alarm management	Acknowledged	Alarm acknowledged, alarm # 1
3.11.2.4.2	Electric meter	Firmware	Alarm management	Activated	Alarm activated, alarm #2
3.11.2.4.	Electric meter	Firmware	Alarm management	Activated	Area alarm
3.15.1.1.	Electric meter	Load control	Alarm	Abort	Switch schedule aborted
2.2.1.17.	Device asset	Battery	Alarm	Battery failure	Battery failed
5.7.9.3.1	Gateway	Customer interface	End alarm	Acknowledged	Customer alarm #1 end
3.16.1.8.	Electric meter	Load profile	Alarm	Almost full	Load profile data area almost full
3.18.1.7.	Electric meter	Memory	Alarm	Allocation error	Memory allocation error
3.11.7.9.	Electric meter	Firmware	Configuration	Alter option	Alter option command executed
3.18.1.8.	Electric meter	Memory	Alarm	Almost full	Disk almost full
3.11.7.10.	Electric meter	Firmware	Configuration	Alternate phone	Alternate phone number used

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
3.21.1.6.	Electric meter	Metrology	Alarm	Alarm cleared	kWh received – alarm cleared
1.23.1.13.	Communications	Network interface	Alarm	Association failure	Network association failure
2.2.18.16.	Device asset	Battery	Status check	Battery charged	Battery OK
3.3.7.27.	Electric meter	Cartridge	Configuration	Change-out	Cartridge changed
1.19.3.18.	Communications	Messaging	Attribute	Blank	Attribute blank
1.23.1.20.	Communications	Network interface	Alarm	Cable failed	Cut cable
1.23.1.22.	Communications	Network interface	Alarm	Call-in failure	Call-in failure
3.34.8.5.	Electric meter	Tariff	Credit	Adjusted	Adjusted the remaining credit
3.11.1.36.	Electric meter	Firmware	Alarm	Command error	Illegal command
3.11.1.36.	Electric meter	Firmware	Alarm	Command error	Delayed command failure
3.31.6.11.	Electric meter	RCD switch	Command	Armed for closure	Arm remote connect/disconnect switch
3.21.4.24.	Electric meter	Metrology	Calibration	Changed	Calibration change
3.21.1.28.1	Electric meter	Metrology	Alarm	Cleared	Meter return within limits, channel #1
2.20.18.14.	Device asset	Meter asset	Status check	Auto registration	Auto registration status
2.20.7.26.	Device asset	Meter asset	Configuration	Changed ID	Meter hard ID changed
2.20.18.15.	Device asset	Meter asset	Status Check	Auto registration confirm	Auto registration confirmation
2.20.7.27.	Device asset	Meter asset	Configuration	Change-out	Meter change-out
7.24.1.24.	Security	Password	Alarm	Changed	Password changed
1.23.1.32.	Communications	Network interface	Alarm	Comm failed	Communications failure
4.13.1.39.	Gas meter	Gas metrology	Alarm	Concentration	Concentration alarm
1.19.1.37.	Communications	Messaging	Alarm	Command format error	Command format error
3.21.14.21.	Electric meter	Metrology	Quality Flag	Calculated	Measurement calculated
6.6.1.47.	Grid power	Current	Alarm	Current loss	Loss of current
3.11.7.40.	Electric meter	Firmware	Configuration	Configuration error	Configuration error
3.15.18.25.	Electric meter	Load control	Status check	Changed by condition	Control point changed by condition
5.7.1.48.1	Gateway	Customer interface	Alarm	Customer alarm	Customer alarm #1
7.24.1.30.	Security	Password	Alarm	Code changed	Station code changed
3.11.18.31.	Electric meter	Firmware	Status check	Cold start	Cold start
3.16.1.43.	Electric meter	Load profile	Alarm	Corrupt data	Load profile data corrupt
7.32.1.23.	Security	Security key	Alarm	Certificate expired	Certificate expiration alert
3.11.1.52.	Electric meter	Firmware	Alarm	Data error	Unknown in analysis
2.36.1.29.	Device asset	Time	Alarm	Clock failed	Clock malfunction
6.25.9.28.	Grid power	Phase	End alarm	Cleared	Phase alert cleared
3.8.1.61.	Electric meter	Demand	Alarm	Demand reset	Demand reset detected
4.13.1.55.	Gas meter	Gas metrology	Alarm	Day limit	Gas day high limit exceeded

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
1.23.17.33.	Communications	Network interface	Status	Comm reestablished	Communications re-established
3.8.1.62.	Electric meter	Demand	Alarm	Demand reset fail	Unable to reset demand
3.15.18.38.	Electric meter	Load control	Status check	Commanded change	Control point changed by command
3.18.1.52.	Electric meter	Memory	Alarm	Data error	Data corruption detection
3.8.6.57.	Electric meter	Demand	Command	Decreased	Demand reduction signal
1.23.17.34.	Communications	Network Interface	Status	Comm terminated	Communication terminated normally
5.14.7.49.	Gateway	Gateway device	Configuration	Customer configuration	Request customer configuration
1.23.1.51.	Communications	Network interface	Alarm	DAP failed	DAP error
3.31.1.41.	Electric meter	RCD switch	Alarm	Connect failed	Connect failed
3.31.18.24.	Electric meter	RCD switch	Status check	Changed	Status input change
6.25.1.45.	Grid power	Phase	Alarm	Cross-phase	Cross phase condition
3.8.6.61.	Electric meter	Demand	Command	Demand reset	Reset demand
3.33.1.44.	Electric meter	Tamper	Alarm	Cover removed	Cover removal detected
3.11.2.66.5	Electric meter	Firmware	Alarm management	Disabled	Alarm disabled, alarm #5
3.31.6.42.	Electric meter	RCD switch	Command	Connected	Close remote Connect/disconnect switch
2.10.1.70.	Device asset	Enclosure	Alarm	Door closed	Door closed
2.10.1.71.	Device asset	Enclosure	Alarm	Door open	Door opened
3.11.1.69.	Electric meter	Firmware	Alarm	Display error	Display failure
3.3.1.79.	Electric meter	Cartridge	Alarm	Error	Cartridge error
6.25.9.46.	Grid power	Phase	End alarm	Cross-phase cleared	Cross phase cleared
2.9.18.63.	Device asset	DER switch	Status check	DER connected	Optional input closure, generator
3.11.2.76.2	Electric meter	Firmware	Alarm management	Enabled	Alarm enabled, alarm #2
5.14.7.66.	Gateway	Gateway device	Configuration	Disabled	Gateway not enabled
2.9.18.64.	Device Asset	DER switch	Status check	DER disconnected	Optional input opening, generator
2.36.1.54.	Device Asset	Time	Alarm	Day change error	Day change error
3.21.4.65.1	Electric meter	Metrology	Calibration	Deviated	Calibration over-deviation, #1
3.31.18.42.	Electric meter	RCD switch	Status check	Connected	Service turn-on
3.3.1.88.	Electric meter	Cartridge	Alarm	Frozen	Cartridge frozen
3.21.14.58.	Electric meter	Metrology	Quality Flag	Default	Measurement default
1.23.1.73.	Communications	Network interface	Alarm	DSP failed	DSP error
3.34.1.61.	Electric meter	Tariff	Alarm	Demand reset	Unscheduled billing reset
3.31.6.59.1	Electric meter	RCD switch	Command	Demand Limiting disabled, formula #1	Disable demand limiting for RCD switch
3.18.1.78.	Electric meter	Memory	Alarm	EPROM failure	EPROM failure

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
3.31.6.60.1	Electric meter	RCD switch	Command	Demand Limiting enabled, formula #1	Enable demand limiting for RCD switch
3.31.1.67.	Electric meter	RCD switch	Alarm	Disconnect failed	Disconnect failed
3.34.8.57.	Electric meter	Tariff	Credit	Decreased	Subtracted from remaining credit
4.13.1.84.	Gas meter	Gas metrology	Alarm	EWM failure	EWM potential failure
3.34.16.50.	Electric meter	Tariff	Setting	Daily rate change	Daily tariff changed
1.19.10.74.	Communications	Messaging	Identity	Duplicate	Duplicate name
3.11.18.72.	Electric meter	Firmware	Status check	Download status	Firmware download status
3.8.1.93.	Electric meter	Demand	Alarm	High limit	Demand threshold caution
4.13.1.87.	Gas meter	Gas metrology	Alarm	FPV unreasonable	FPV calculated unreasonably
2.20.10.74.	Device asset	Meter asset	Identity	Duplicate	Duplicate device name
2.20.10.74.	Device asset	Meter asset	Identity	Duplicate	Multiple devices with serial number
2.20.18.66.	Device asset	Meter asset	Status check	Disabled	Device not enabled
2.20.18.66.	Device asset	Meter asset	Status check	Disabled	Parent device not enabled
3.31.6.66.	Electric meter	RCD switch	Command	Disabled	Disable RCD switch
2.36.16.53.	Device asset	Time	Setting	Date changed	Date setting changed
4.13.1.89.	Gas meter	Gas metrology	Alarm	GC failure	GC analyzer failure
6.6.1.94.	Grid power	Current	Alarm	High neutral current	High neutral current
1.23.10.74.	Communications	Network interface	Identity	Duplicate	Duplicate network interface ID
3.18.1.86.	Electric meter	Memory	Alarm	Firmware failure	Firmware image corrupted
3.31.6.68.	Electric meter	RCD switch	Command	Disconnected	Open remote connect/disconnect switch
2.36.16.56.	Device Asset	Time	Setting	Daylight time	Change to spring daylight savings time
3.33.1.77.	Electric meter	Tamper	Alarm	Encoder	Encoder tamper
3.34.18.61.	Electric meter	Tariff	Status check	Demand reset	Billing reset
3.31.6.76.	Electric meter	RCD switch	Command	Enabled	Enable RCD switch
6.25.1.85.	Grid power	Phase	Alarm	Failed	Loss of voltage transducer on one phase
6.25.1.85.	Grid power	Phase	Alarm	Failed	Loss of voltage per phase
6.25.1.85.	Grid power	Phase	Alarm	Failed	Phase drop out
2.17.18.81.	Device asset	Logs	Status check	Event log cleared	Event log cleared
3.21.1.93.	Electric meter	Metrology	Alarm	High limit	High limit alarm
3.21.14.80.	Electric meter	Metrology	Quality flag	Estimated	Measurement estimated
2.17.18.82.	Device Asset	Logs	Status check	Event log updated	Event log last read entry updated
2.17.18.83.	Device Asset	Logs	Status check	Event log uploaded	Event log uploaded
3.31.18.68.	Electric meter	RCD switch	Status check	Disconnected	Service turn-off

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
3.34.8.75.	Electric meter	Tariff	Credit	Emergency increase	Pre-payment emergency credit
3.11.18.90.	Electric meter	Firmware	Status check	Health OK	Meter health checked
6.28.1.91.	Grid power	Power quality	Alarm	High distortion	High distortion detected
4.29.1.93.	Gas meter	Pressure	Alarm	High limit	Sensed pressure equals P max
2.35.1.93.	Device asset	Temperature	Alarm	High limit	Sensed temperature equals T max
2.17.18.95.	Device asset	Logs	Status check	History cleared	History log cleared
2.17.18.96.	Device asset	Logs	Status check	History updated	History log last read entry updated
6.28.9.92.	Grid power	Power quality	End alarm	High distortion cleared	High distortion cleared
6.38.1.93.	Grid power	Voltage	Alarm	High limit	High voltage
2.36.16.97.	Device asset	Time	Setting	Holiday	Holiday changed
6.38.9.99.	Grid power	Voltage	End alarm	Imbalance cleared	Voltage imbalance cleared
6.38.17.98.	Grid power	Voltage	Status	Imbalance	Voltage imbalance condition
3.11.1.104.	Electric meter	Firmware	Alarm	Input error	Sensor malfunction
3.11.1.106.	Electric meter	Firmware	Alarm	Invalid	Invalid down limit
3.21.1.104.1	Electric meter	Metrology	Alarm	Input error	Form C error, input #1
3.21.1.106.	Electric meter	Metrology	Alarm	Invalid	Measurement failed
3.21.1.106.	Electric meter	Metrology	Alarm	Invalid	Measurement error detected
1.19.1.111.	Communications	Messaging	Alarm	Invalid delete	Invalid delete data points
6.25.1.100.	Grid power	Phase	Alarm	Inactive	Inactive phase condition
5.14.1.113.	Gateway	Gateway device	Alarm	Invalid gateway ID	Invalid gateway ID
7.24.1.106.	Security	Password	Alarm	Invalid	Invalid password
1.19.1.118.	Communications	Messaging	Alarm	Invalid priority	Invalid priority not supported
4.13.1.122.	Gas meter	Gas metrology	Alarm	Invalid sigmatic	Invalid sigmatic message
2.36.1.110.	Device asset	Time	Alarm	Invalid date	Invalid installation date
2.36.1.110.	Device asset	Time	Alarm	Invalid date	Invalid execution date and time
2.36.1.110.	Device asset	Time	Alarm	Invalid date	Invalid complete date and time range
1.19.1.129.	Communications	Messaging	Alarm	Invalid XML	Invalid XML format
2.2.1.149.	Device asset	Battery	Alarm	Low battery detected	Battery low condition detected
7.32.1.114.	Security	Security key	Alarm	Invalid key	Invalid security key
1.23.1.130.	Communications	Network interface	Alarm	IO mismatch	I/O board definition mismatch
7.1.1.147.	Security	Access	Alarm	Login failed	User login failure
3.21.1.134.	Electric meter	Metrology	Alarm	KYZ failure	KYZ alarm
2.36.1.124.	Device asset	Time	Alarm	Invalid time zone	Invalid time zone ID

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
2.36.1.124.	Device asset	Time	Alarm	Invalid time zone	Daylight savings or standard time variance error
6.6.1.150.	Grid power	Current	Alarm	Low limit	Low current limit exceeded
2.36.1.125.	Device Asset	Time	Alarm	Invalid timeout	Invalid timeout date and time
3.21.1.139.	Electric meter	Metrology	Alarm	Limit exceeded	Meter over/under threshold
3.21.1.139.	Electric meter	Metrology	Alarm	Limit exceeded	Limit alarm reported
2.17.1.146.	Device asset	Logs	Alarm	Log error	Event log error
3.11.1.154.	Electric meter	Firmware	Alarm	Memory error	End of data event table reached
3.11.1.155.	Electric meter	Firmware	Alarm	Message error	Unintelligible message received
3.11.1.155.	Electric meter	Firmware	Alarm	Message error	Parameter error
3.11.1.155.	Electric meter	Firmware	Alarm	Message error	Un-programmed caution
3.8.1.159.	Electric meter	Demand	Alarm	Mismatch	Demand definitions mismatch
6.25.1.139.	Grid power	Phase	Alarm	Limit exceeded	Phase angle limit exceeded
6.27.1.137.	Grid power	Power factor	Alarm	Leading kvarh	Leading kvarh caution
4.29.1.139.	Gas meter	Pressure	Alarm	Limit exceeded	Pressure is out of range
6.27.1.139.	Grid power	Power factor	Alarm	Limit exceeded	Power factor exceeded limits
3.21.1.150.	Electric meter	Metrology	Alarm	Low limit	Low limit alarm
1.23.1.151.	Communications	Network interface	Alarm	Low modem battery	Power outage modem battery low
1.19.1.156.	Communications	Messaging	Alarm	Message failed	Message failure
2.35.1.139.	Device asset	Temperature	Alarm	Limit exceeded	Temperature is out of range
2.36.1.139.	Device asset	Time	Alarm	Limit exceeded	Time change > 10 s
2.36.1.139.	Device asset	Time	Alarm	Limit exceeded	Time variance exceeded limits even after module tries to correct it
2.36.1.139.	Device asset	Time	Alarm	Limit exceeded	Time variance exceeded limits without attempt to correct automatically
1.19.1.159.	Communications	Messaging	Alarm	Mismatch	Communications mismatch
3.18.1.159.	Electric meter	Memory	Alarm	Mismatch	Family definitions mismatch
3.18.1.159.	Electric meter	Memory	Alarm	Mismatch	Mass memory mismatch
1.23.1.157.	Communications	Network interface	Alarm	Meter comm failed	Communications between modules within a meter lost
4.13.1.164.	Gas meter	Gas metrology	Alarm	MOL out of range	Total MOL % out of range
4.29.1.150.	Gas meter	Pressure	Alarm	Low limit	Sensed pressure equals P min
6.38.1.139.	Grid power	Voltage	Alarm	Limit exceeded	Voltage exception
1.23.1.162.	Communications	Network interface	Alarm	Modem failed	Modem failure detected

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
2.35.1.150.	Device asset	Temperature	Alarm	Low limit	Sensed temperature equals T min
3.33.1.152.	Electric meter	Tamper	Alarm	Magnetic	Magnetic switch alarm
3.8.1.178.	Electric meter	Demand	Alarm	Overload	Demand overload threshold
3.11.1.176.	Electric meter	Firmware	Alarm	Option board error	Option board error
1.23.1.169.	Communications	Network interface	Alarm	Node failed	Node failure
1.23.1.170.	Communications	Network interface	Alarm	Node link failed	Node link failure
3.18.1.173.	Electric meter	Memory	Alarm	NVRAM failure	NVRAM failure
6.38.1.150.	Grid power	Voltage	Alarm	Low limit	Low voltage
3.18.1.177.	Electric meter	Memory	Alarm	Overflow	Buffer overflow/overrun error
3.11.1.193.	Electric meter	Firmware	Alarm	Program error	Program malfunction
3.11.1.193.	Electric meter	Firmware	Alarm	Program error	Illegal command processor state
6.20.1.185.	Grid power	Meter asset	Alarm	Power out	Power outage
3.26.1.185.	Electric meter	Power	Alarm	Power out	Undetected power outage
3.3.1.212.	Electric meter	Cartridge	Alarm	Removed	Cartridge removed
3.21.1.194.	Electric meter	Metrology	Alarm	Pulse failure	Pulse initiator
3.18.1.199.	Electric meter	Memory	Alarm	RAM failure	RAM error
3.33.1.184.	Electric meter	Tamper	Alarm	Physical	Physical tamper
1.23.1.197.	Communications	Network interface	Alarm	Queue	Queue alarm
3.18.1.200.	Electric meter	Memory	Alarm	RAM full	RAM memory full
6.28.1.187.	Grid power	Power quality	Alarm	PQ event occurred	Generic power quality event
1.23.1.198.	Communications	Network interface	Alarm	Radio failed	Radio faults
6.28.1.188.	Grid power	Power quality	Alarm	PQ event started	Power quality event start
3.11.1.209.	Electric meter	Firmware	Alarm	Register error	Register processor failure
3.11.1.209.	Electric meter	Firmware	Alarm	Register error	Meter bus alarm
3.21.1.203.	Electric meter	Metrology	Alarm	Read Missing	Expected read missed
3.11.1.214.	Electric meter	Firmware	Alarm	Reset	Metering equipment reset
3.11.1.214.	Electric meter	Firmware	Alarm	Reset	Processor reset
6.37.1.185.	Grid power	Transformer device	Alarm	Power out	Transformer outage
3.11.1.215.	Electric meter	Firmware	Alarm	Reset error	Unable to reset
3.31.1.195.	Electric meter	RCD switch	Alarm	Pushbutton pressed	RCD Switch pushbutton pressed
3.21.1.209.	Electric meter	Metrology	Alarm	Register error	Register definitions mismatch
3.21.1.209.	Electric meter	Metrology	Alarm	Register error	Register full
3.21.1.209.1	Electric meter	Metrology	Alarm	Register error	Pulse overflow, channel #1
3.21.1.209.	Electric meter	Metrology	Alarm	Register error	Register error
1.23.1.210.	Communications	Network interface	Alarm	Register failed	Intelligent register communication failure
1.23.1.210.	Communications	Network interface	Alarm	Register failed	Encoder register communication failure

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
1.23.1.214.	Communications	Network interface	Alarm	Reset	Reset alarm
7.1.1.230.	Security	Access	Alarm	Security event	Security event
3.18.1.220.	Electric meter	Memory	Alarm	ROM failure	ROM error
6.20.1.216.	Grid power	Meter asset	Alarm	Restored	Meter restoration
3.11.1.232.	Electric meter	Firmware	Alarm	Self-test error	Diagnostic failure
2.36.1.215.	Device asset	Time	Alarm	Reset error	Unable to reset time
1.19.1.234.	Communications	Messaging	Alarm	Session interrupted	Communication session interrupted
1.19.1.235.	Communications	Messaging	Alarm	Session too long	Communication session too long
3.33.1.219.	Electric meter	Tamper	Alarm	Reverse rotation	Reverse flow detected
4.13.1.240.	Gas meter	Gas metrology	Alarm	Specific gravity	Specific gravity alarm
6.37.1.216.	Grid power	Transformer device	Alarm	Restored	Transformer restoration
3.16.1.245.	Electric meter	Load profile	Alarm	Survey failed	Load survey readings failed
6.25.1.233.	Grid power	Phase	Alarm	Sequence error	Phase sequence error
3.18.1.255.	Electric meter	Memory	Alarm	Table full	Data table full
9.5.1.267.	Turbine	Control	Alarm	Turbine pulse error	Turbine pulse error
6.12.1.270.	Grid power	Frequency	Alarm	Unstable	Line frequency unstable
2.36.1.252.	Device asset	Time	Alarm	Synch failed	Time synchronization error
3.33.1.257.	Electric meter	Tamper	Alarm	Tamper detected	Tamper indication
4.13.1.277.	Gas meter	Gas metrology	Alarm	Volume calc error	Corrected volume calculation error
3.33.1.263.	Electric meter	Tamper	Alarm	Tilt	Tilt detected
1.23.1.279.	Communications	Network interface	Alarm	Watchdog failed	Watch dog or keep-alive failure detected
3.34.1.266.	Electric meter	Tariff	Alarm	TOU mismatch	TOU start mismatch
3.34.1.266.	Electric meter	Tariff	Alarm	TOU mismatch	TOU mismatch
3.34.1.266.	Electric meter	Tariff	Alarm	TOU mismatch	TOU years mismatch
6.38.1.276.	Grid power	Voltage	Alarm	Voltage loss	Loss of voltage transducer VT
6.38.1.276.	Grid power	Voltage	Alarm	Voltage loss	Low loss potential detected
3.11.2.236.2	Electric meter	Firmware	Alarm management	Set	Alarm value set, alarm #2
1.19.3.108.	Communications	Messaging	Attribute	Invalid blank name	Invalid attribute defined value name blank
1.19.3.112.	Communications	Messaging	Attribute	Invalid duplicate name	Invalid attribute defined value duplicate name
1.19.3.115.	Communications	Messaging	Attribute	Invalid null	Invalid null parameter
1.19.3.116.	Communications	Messaging	Attribute	Invalid numeric	Invalid numeric attribute
1.19.3.126.	Communications	Messaging	Attribute	Invalid type	Invalid attribute type
3.21.4.236.3	Electric meter	Metrology	Calibration	Set	Calibration value set, #3
6.4.5.185.	Grid power	Circuit	Check status	Power out	Verify feeder power
6.20.5.185.	Grid power	Meter asset	Check status	Power out	Verify meter power
6.30.5.185.	Grid power	Protective device	Check status	Power out	Verify section power

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
6.37.5.185.	Grid power	Transformer device	Check status	Power out	Verify transformer power
3.34.6.201.	Electric meter	Tariff	Command	Rate change	Price signal
3.15.6.242.	Electric meter	Load control	Command	Started	Load control initiate
3.15.6.243.	Electric meter	Load control	Command	Stopped	Load control restore
3.3.7.105.	Electric meter	Cartridge	Configuration	Installed	Cartridge installed
5.14.7.105.	Gateway	Gateway device	Configuration	Installed	Gateway installed
2.20.7.105.	Device asset	Meter asset	Configuration	Installed	Meter installed
3.11.7.117.	Electric meter	Firmware	Configuration	Invalid parameter	Constants invalid
3.11.7.117.	Electric meter	Firmware	Configuration	Invalid parameter	Invalid re-programming data
3.11.7.128.	Electric meter	Firmware	Configuration	Invalid version	Internal device ID mismatch
3.11.7.128.	Electric meter	Firmware	Configuration	Invalid version	Invalid meter software version
3.11.7.159.	Electric meter	Firmware	Configuration	Mismatch	Miscellaneous configuration mismatch
3.11.7.159.	Electric meter	Firmware	Configuration	Mismatch	Invalid device type, e.g. not TOU
3.11.7.171.	Electric meter	Firmware	Configuration	Not initialized	Meter not programmed
3.11.7.213.	Electric meter	Firmware	Configuration	Reprogrammed	Configuration written/changed
3.11.7.213.	Electric meter	Firmware	Configuration	Reprogrammed	Reprogrammed
3.11.7.213.	Electric meter	Firmware	Configuration	Reprogrammed	Meter reconfigured
5.14.7.212.	Gateway	Gateway device	Configuration	Removed	Gateway removed
5.14.7.213.	Gateway	Gateway device	Configuration	Reprogrammed	Gateway reprogrammed
2.20.7.212.	Device asset	Meter asset	Configuration	Removed	Meter removed
2.20.7.227.	Device asset	Meter asset	Configuration	Sealed	End device sealed
3.11.7.237.	Electric meter	Firmware	Configuration	Setup disabled	Remote setup disabled
3.11.7.238.	Electric meter	Firmware	Configuration	Setup enabled	Remote setup enabled
3.11.7.250.	Electric meter	Firmware	Configuration	Synch disabled	Remote synch disabled
3.11.7.251.	Electric meter	Firmware	Configuration	Synch enabled	Remote synch enabled
3.11.7.253.	Electric meter	Firmware	Configuration	Synched	Device has been synchronized
3.11.7.272.	Electric meter	Firmware	Configuration	Upgrade pending	Firmware update pending
2.20.7.269.	Device asset	Meter asset	Configuration	Unsealed	End device unsealed
3.11.7.280.	Electric meter	Firmware	Configuration	Watchdog Reset	Last reset by watchdog
3.34.8.102.	Electric meter	Tariff	Credit	Increased	Added to remaining credit
6.25.9.101.	Grid power	Phase	End alarm	Inactive cleared	Inactive phase cleared
6.27.9.138.	Grid power	Power factor	End alarm	Leading kvarh cleared	Leading kvarh cleared
6.25.9.183.	Grid power	Phase	End alarm	Phase restored	Phase restoration
6.28.9.186.	Grid power	Power quality	End alarm	PQ event ended	Power quality event end
3.26.9.216.	Electric meter	Power	End alarm	Restored	Power on/restored
2.35.9.218.	Device asset	Temperature	End alarm	Returned within limit	Temperature returns within range

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
6.38.9.224.	Grid power	Voltage	End alarm	Sag cleared	Voltage sag cleared
6.38.9.249.	Grid power	Voltage	End alarm	Swell cleared	Voltage swell cleared
1.19.10.107.	Communications	Messaging	Identity	Invalid attribute	Invalid attribute ID
2.20.10.106.	Device asset	Meter asset	Identity	Invalid	Invalid device ID
2.20.10.106.	Device asset	Meter asset	Identity	Invalid	Invalid serial number
1.19.10.109.	Communications	Messaging	Identity	Invalid command	Invalid command ID
1.19.10.119.	Communications	Messaging	Identity	Invalid return	Invalid information return type ID
1.19.10.120.	Communications	Messaging	Identity	Invalid search	Invalid search ID
1.23.10.121.	Communications	Network interface	Identity	Invalid serial number	Invalid serial number for network interface
1.19.10.127.	Communications	Messaging	Identity	Invalid value	Invalid defined value ID
8.37.10.106.	Transformer	Transformer device	Identity	Invalid	Invalid transformer ID
2.20.10.160.	Device asset	Meter asset	Identity	Missing	Missing node
2.20.10.271.	Device asset	Meter asset	Identity	Unsupported	Unsupported device type ID
3.22.11.242.	Electric meter	Mode	Maint mode	Started	Maintenance mode
3.22.11.242.	Electric meter	Mode	Maint mode	Started	Meter shop mode started
3.22.11.243.	Electric meter	Mode	Maint mode	Stopped	Meter shop mode stopped
3.22.12.242.	Electric meter	Mode	Metering mode	Started	Begin recording meter readings
3.22.12.243.	Electric meter	Mode	Metering mode	Stopped	Metering mode stopped
3.26.13.158.	Electric meter	Power	Outage	Minutes	Outage minutes
3.26.13.165.	Electric meter	Power	Outage	Momentary	Momentary outage <5 min
3.26.13.166.	Electric meter	Power	Outage	Momentary events	Number of momentary outages
3.26.13.229.	Electric meter	Power	Outage	Seconds	Outage seconds
3.26.13.246.	Electric meter	Power	Outage	Sustained	Outage > 5 min
3.26.13.247.	Electric meter	Power	Outage	Sustained events	Number of sustained outages
3.26.13.265.	Electric meter	Power	Outage	Total events	Total outages
3.21.14.103.	Electric meter	Metrology	Quality flag	Initial read	Measurement initial
3.21.14.135.	Electric meter	Metrology	Quality flag	Last read	Measurement last
3.21.14.153.	Electric meter	Metrology	Quality flag	Measured	Measurement measured
3.21.14.190.	Electric meter	Metrology	Quality flag	Preset	Pre-set register values
3.21.14.196.0	Electric meter	Metrology	Quality flag	Questionable	No flow, channel #0
3.21.14.260.	Electric meter	Metrology	Quality flag	Test mode	Test mode data
3.21.15.175.	Electric meter	Metrology	Read type	On-request	Meter readings passed
3.21.15.175.	Electric meter	Metrology	Read type	On-request	Verify read
3.21.15.225.	Electric meter	Metrology	Read type	Scheduled	Scheduled call-in
3.21.15.231.	Electric meter	Metrology	Read type	Self read	Self meter read
2.36.16.167.	Device asset	Time	Setting	New time	Time setting changed to new time
2.36.16.167.	Device asset	Time	Setting	New time	DST change to new time

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
2.36.16.174.	Device asset	Time	Setting	Old time	Time setting reset to previous time
2.36.16.174.	Device asset	Time	Setting	Old time	Time setting changed (old time)
2.36.16.174.	Device asset	Time	Setting	Old time	Working register time set to previous time
2.36.16.174.	Device asset	Time	Setting	Old time	DST change to previous time
3.34.16.192.	Electric meter	Tariff	Setting	Program changed	Tariff program changed
3.34.16.201.	Electric meter	Tariff	Setting	Rate change	Rate change
6.28.16.207.	Grid power	Power quality	Setting	Recording disabled	Recording disabled
6.28.16.208.	Grid power	Power quality	Setting	Recording enabled	Recording enabled
2.36.16.228.	Device asset	Time	Setting	Season changed	Season changed
2.36.16.241.	Device asset	Time	Setting	Standard time	Change to fall standard time
2.36.16.254.	Device asset	Time	Setting	Synchronized	Time reset
6.25.16.261.	Grid power	Phase	Setting	Threshold	Phase threshold
3.34.16.262.	Electric meter	Tariff	Setting	Tier changed	Tier changed
2.36.16.264.	Device asset	Time	Setting	Time changed	Time setting changed
3.34.16.281.	Electric meter	Tariff	Setting	Week rate change	Week tariff changed
7.1.17.145.	Security	Access	Status	Local access	Unit accessed
7.32.17.131.	Security	Security key	Status	Key disabled	Local key disable
7.32.17.132.	Security	Security key	Status	Key enabled	Local key enable
7.32.17.133.	Security	Security key	Status	Key unlocked	Local key unlock
7.32.17.133.	Security	Security key	Status	Key unlocked	Local key enable/disable
1.23.17.161.	Communications	Network interface	Status	Modem change-out	Modem change-out
1.23.17.163.	Communications	Network interface	Status	Modem initialized	Modem initialized
1.23.17.168.	Communications	Network interface	Status	No modem	No modem
1.19.17.180.	Communications	Messaging	Status	Partial success	Partial success
7.1.17.202.	Security	Access	Status	Read access	Unit accessed by reader
7.1.17.204.	Security	Access	Status	Read secure register	Read secured register
7.1.17.205.	Security	Access	Status	Read secure table	Read secured table
7.1.17.206.	Security	Access	Status	Recoder access	Unit accessed by recorder
7.1.17.211.	Security	Access	Status	Remote access	Unit accessed remotely
1.19.17.217.	Communications	Messaging	Status	Results pending	Results pending
1.19.17.244.	Communications	Messaging	Status	Success	Succeeded
6.38.17.223.	Grid power	Voltage	Status	Sag	Voltage sag
7.1.17.282.	Security	Access	Status	Write access	End device accessed for write
6.38.17.248.	Grid power	Voltage	Status	Swell	Voltage swell condition

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
6.38.17.273.	Grid power	Voltage	Status	Voltage A inactive	Potential A inactive
6.38.17.274.	Grid power	Voltage	Status	Voltage B inactive	Potential B inactive
6.38.17.275.	Grid power	Voltage	Status	Voltage C inactive	Potential C inactive
3.21.18.136.	Electric meter	Metrology	Status check	Last read updated	Last read entry updated
3.21.18.140.	Electric meter	Metrology	Status check	List cleared	List cleared
3.21.18.141.	Electric meter	Metrology	Status check	List pointers reset	Reset list pointers
3.21.18.142.	Electric meter	Metrology	Status check	List pointers updated	Update list pointers
3.15.18.165.	Electric meter	Load control	Status check	Momentary	Load control relay momentary closure, relay #
3.15.18.165.	Electric meter	Load control	Status check	Momentary	Output control relay toggled, relay #
3.18.18.179.	Electric meter	Memory	Status check	Parameter change	EPROM read
3.18.18.179.	Electric meter	Memory	Status check	Parameter change	EPROM written
3.18.18.179.	Electric meter	Memory	Status check	Parameter change	Storage medium changed
3.18.18.179.	Electric meter	Memory	Status check	Parameter change	Constants changed
3.18.18.179.	Electric meter	Memory	Status check	Parameter change	Primary constant changed
3.18.18.179.	Electric meter	Memory	Status check	Parameter change	Secondary constant changed
3.18.18.179.	Electric meter	Memory	Status check	Parameter change	Output constant changed
3.11.18.191.	Electric meter	Firmware	Status check	Procedure invoked	Meter procedure invoked
3.21.18.181.	Electric meter	Metrology	Status check	Pending table activation	Pending table activation
3.21.18.182.	Electric meter	Metrology	Status check	Pending table clear	Pending table clear
3.15.18.189.	Electric meter	Load control	Status check	Prepayment change	Control point changed due to prepayment
3.15.18.225.	Electric meter	Load control	Status check	Scheduled	Schedule loaded
3.15.18.226.	Electric meter	Load control	Status check	Scheduled change	Control point changed by schedule
3.34.18.221.	Electric meter	Tariff	Status check	RTP activated	RTP activation
3.34.18.222.	Electric meter	Tariff	Status check	RTP deactivated	RTP deactivation
3.15.18.242.	Electric meter	Load control	Status check	Started	Load control initiate
3.15.18.243.	Electric meter	Load control	Status check	Stopped	Load control restore
3.11.18.259.	Electric meter	Firmware	Status check	Test call	Test call-in
3.34.18.239.	Electric meter	Tariff	Status check	Special schedule	Special schedule activation
2.20.18.258.	Device asset	Meter asset	Status check	Test button pushed	Test button pushed
3.21.18.256.	Electric meter	Metrology	Status check	Table written	Table written to

End device Event code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
3.11.18.278.	Electric meter	Firmware	Status check	Warm start	Warm start
3.22.19.242.	Electric meter	Mode	Test mode	Started	Test mode started
3.22.19.243.	Electric meter	Mode	Test mode	Stopped	Test mode ended

E.8 Example EndDeviceEvent request codes

Just as the EndDeviceEvent codes offer an enumerated code that can be used in a publication or reply, IEC 61968-9 also supports the use of EndDeviceEvent codes as part of a request to solicit EndDeviceEvent code data in the response. For a request-type query, the header should employ a suitable verb such as “GET”. Example codes are provided in Table E.6.

Table E.6 – Example EndDeviceEvent codes

EndDeviceEvent code	Domain or equipment	DomainPart	Type or issue	Attribute	Meter/Headend event
6.4.5.185.	Grid power	Circuit	Status check	Power out	Verify feeder power
6.20.5.185.	Grid power	Meter asset	Status check	Power out	Verify meter power
6.30.5.185.	Grid power	Protective device	Status check	Power out	Verify section power
6.37.5.185.	Grid power	Transformer device	Status check	Power out	Verify transformer power
3.31.6.11.	Electric meter	RCD switch	Status check	Armed for closure	Is RCD armed?
3.31.21.11.	Electric meter	RCD switch	Has feature	Armed for closure	Does the RCD support the “arm for closure” feature?

Annex F
(informative)

Recommended EndDeviceControl code enumerations

F.1 General

Many metering systems have the capability to issue a command to equipment in the field that takes an action. The action might be to perform a demand-reset, open or close a switch, or a number of other activities. This annex describes example codes that may be used for such applications.

The recommended approach is to create EndDeviceEvent codes that are of the “command” type. The code for the response of the same condition is likely to be of an “alarm” type. For example, compare the codes for “demand reset” in Table F.1, as well as the EndDeviceEvent command codes in Table E.6.

Table F.1 – Comparison of example demand reset codes

Domain	DomainPart	Type	Attribute	Comment
Electric meter	Demand	Alarm	Demand reset	Demand reset detected
Electric meter	Demand	Command	Demand reset	Reset demand

In addition to changing the “type” of code to a “command”, the requestor must also use a suitable verb in the CIM header such as “CREATE”. Within an XML message, the code is specified using the EndDeviceControl.type element.

F.2 Example EndDeviceControl codes

Example codes, chosen from selected EndDeviceEvent status codes, are listed in Table F.2.

Table F.2 – Example EndDevice control codes

EndDeviceEvent code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
3.8.6.57.	Electric meter	Demand	Command	Decreased	Demand reduction signal
3.8.6.61.	Electric meter	Demand	Command	Demand reset	Reset demand
3.15.6.242.0	Electric meter	Load control	Command	Started	Load control initiate, port 0
3.15.6.243.1	Electric meter	Load control	Command	Stopped	Load control restore, port 1
3.31.6.11.	Electric meter	RCD switch	Command	Armed for closure	Arm remote connect/disconnect switch
3.31.6.42.	Electric meter	RCD switch	Command	Connected	Close remote connect/disconnect switch
3.31.6.59.1	Electric meter	RCD switch	Command	Demand limiting disabled, formula #	Disable demand limiting formula #1 for RCD switch

EndDeviceEvent code	Domain or equipment	Domain part	Type or issue	Attribute	Meter/Headend event
3.31.6.60.2	Electric meter	RCD switch	Command	Demand limiting enabled, formula #	Enable demand limiting formula #2 for RCD switch
3.31.6.66.	Electric meter	RCD switch	Command	Disabled	Disable RCD Switch
3.31.6.68.	Electric meter	RCD switch	Command	Disconnected	Open remote connect/disconnect switch
3.31.6.76.	Electric meter	RCD switch	Command	Enabled	Enable RCD switch
3.34.6.201.	Electric meter	Tariff	Command	Rate change	Price signal

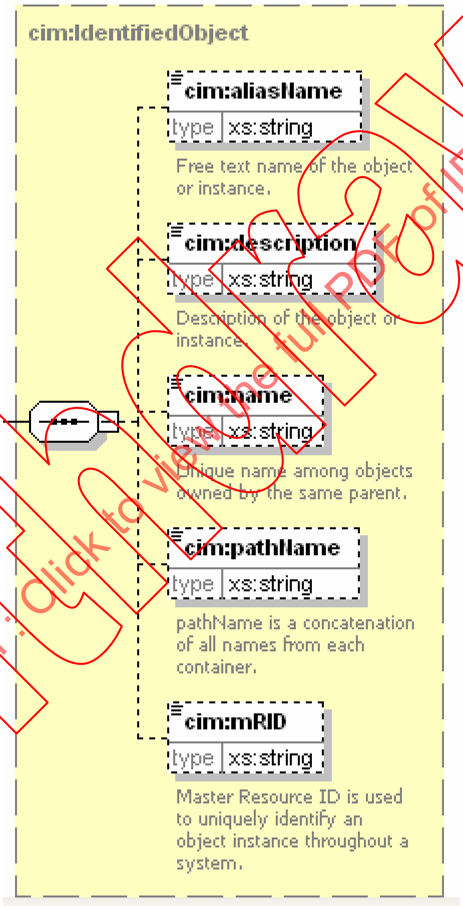
Annex G
(informative)

Recommended procedure for maintaining relationships between objects

G.1 General

This annex describes an example of the use of IEC 61968 based messages to maintain synchronization between an MDM system and a MS.

G.2 Object identifiers


cim:IdentifiedObject
cim:aliasName
type xs:string
Free text name of the object or instance.
cim:description
type xs:string
Description of the object or instance.
cim:name
type xs:string
Unique name among objects owned by the same parent.
cim:pathName
type xs:string
pathName is a concatenation of all names from each container.
cim:mRID
type xs:string
Master Resource ID is used to uniquely identify an object instance throughout a system.

IEC 1795/09

Figure G.1 – Typical CIM naming object

Objects should be given a "name" by the system of record by using one of the naming elements. Figure G.1 shows a typical naming object which appears by extension in nearly every class. In some cases, the utility must identify which application they want to have serve as the system of record for a given class. If a neutral ID service is available to broker a globally unique ID for the object, it enables use of the "mRID" attribute as the authoritative name. If such a service is not available, the "name" attribute will be used as the authoritative name, however it may be difficult to maintain context in some cases. For uniqueness when using the "name" attribute, it may be necessary to also populate the pathName attribute with the list of classes from the top to the named object (e.g. the xpath). The tradition is to name each class from the top to the bottom of the hierarchy, by listing them left to right, separating them by a slash "/".

G.3 CIM based message exchanges

G.3.1 Initialization

The CIM can be used in very flexible ways. At some installations, it may be the practice to send information from one application to another in piecemeal fashion to reflect the current development (in near-real-time) for every aspect of a deployment. Each time a message is transmitted[1] the header can contain a timestamp which indicates when the update took place. Alternatively, some applications (especially MDM systems) tend to have a policy of packaging all of the related data together and sending it all at once to the MS. (The advantage to this approach is that the data consumer can easily check to see if all of the required data are present, and reject the data if it sees any problems.) Regardless of the approach used, it is important that data flow through the applications in the appropriate order. This way, the system of record can be allowed to name the object before it is disseminated.

The CIM header allows one verb to be applied to one noun. This simple arrangement works well to keep the interface understandable, and depending on the noun used, may cause each of the classes of data to be published in separate messages (e.g. meterAsset data is separated from customer data, which is separated from networkDataSets, etc.). The CustomerMeterDataSet message on the other hand allows all of the salient meter installation data to be presented in a single schema. When published, the data producer would say that the item has been (verb=) CREATED.

G.3.2 Installation

If data is presented to the MS in piecemeal fashion prior to installation, then all that is needed upon installation is a message that ties the pieces together. The CustomerMeterDataSet message is good at this because it captures all of the salient information in a single schema. The CustomerMeterDataSet message can be populated with the name identifiers to indicate the relationship between the particular instances of a meter, customer, premises, etc. Alternatively, if the MDM has a policy of presenting all of the data at once, the CustomerMeterDataSet can be fully populated and presented as a full message.

G.3.3 Updates

G.3.3.1 General

When a data producer publishes an update, it would say that a particular noun has been CHANGED. Only the items that have changed should be included in the message. The data consumer can treat the naming identifier as an immutable key. All other data presented in the message should be considered “CHANGED”, and the relationships between named identifiers are also changeable.

G.3.3.2 Change of tenancy

When a customer moves out of a premises, and a new customer has yet to move in, it is the practice of many utilities to either assign usage to an idle account (typically something like “99999”), or to a new non-existing account that will be assigned to the next tenant. In these cases, the CHANGED verb would be an ideal choice. Some may prefer to use the DELETE verb. Such a choice could have ramifications within the system depending on its data retention policies. Some vendors may elect to not support the DELETE verb in certain contexts, and if they do support it, do not actually delete the data, but at most break the relationships between objects. Care should be taken in the use of the DELETE verb.

[1] See Annex B for a recommended message structure.

G.4 Example

Table G.1 describes a collection of information that might need to be passed from the MDM to the MS as a result of a meter installation. Certain pieces may be passed early, such as when a meter communication module is installed in a meter, the relation between the two has been established, and the initial configuration is known. This information can be passed through the systems prior to deployment of the new meter, or after installation – depending on the policy of the systems. Likewise, when a meter is removed from service and taken to the meter shop, the meter's location should change. If the meter is disassembled, and the ComModule removed, the relationship between these two can be broken as well, and the assets assigned a new status and new locations.

Table G.1 – Example meter relationships

Description of item	Value	CIM location	Type
This communication module	MCM-ID	EndDeviceAsset.DeviceFunction.ComFunction.amrAddress	name/mRID
Is servicing this end device	End-device ID	MeterAsset.name	name/mRID
(Which is let's say an electricity meter)	End-device type	ServiceDeliveryPoint.serviceType	Attribute
Using this connection	Port Number	EndDeviceAsset.DeviceFunction.ComPort	Attribute
	Upstream CommAsset	EndDeviceAsset.DeviceFunction.	name/mRID
At this location	ServiceDeliveryPoint	ServiceDeliveryPoint.name	name/mRID
For this account	Account ID	CustomerAccount.name	name/mRID
And this customer	Customer ID	CustomerAccount.Customer.name	name/mRID
In this utility	Business Unit		
Commencing at this date	Installation date	ActivityRecord.statusDateTime	Attribute
Because	"In-Service"	ActivityRecord.status	Attribute

Note that names (mRID's) serve as immutable keys – ordinary attributes do not.

Annex H
(normative)

XML Schemas for message payloads

H.1 General

The purpose of this annex is to provide XML schemas for message payloads to augment the descriptions provided earlier in this document. These XML schemas were defined using profile definitions within CIMTool. These schemas may be extended as needed for specific implementation needs.

H.2 CustomerMeterDataSet

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/CustomerMeterDataSet#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/CustomerMeterDataSet#">
<xs:element name="CustomerMeterDataSet" type="m:CustomerMeterDataSet"/>
<xs:complexType name="CustomerMeterDataSet">
<xs:sequence>
<xs:element name="Customer" type="m:Customer" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="CustomerAccount" type="m:CustomerAccount" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="CustomerAgreement" type="m:CustomerAgreement" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="EndDeviceAsset" type="m:EndDeviceAsset" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="SDPLocation" type="m:SDPLocation" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ServiceCategory" type="m:ServiceCategory" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ServiceDeliveryPoint" type="m:ServiceDeliveryPoint" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ServiceLocation" type="m:ServiceLocation" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="Customer" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#Customer">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.aliasName"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.description"/>
<xs:element name="kind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CustomerKind">
<xs:restriction base="xs:string">
<xs:enumeration value="commercialIndustrial"/>
<xs:enumeration value="energyServiceScheduler"/>
```

```
<xs:enumeration value="energyServiceSupplier"/>
<xs:enumeration value="internalUse"/>
<xs:enumeration value="other"/>
<xs:enumeration value="pumpingLoad"/>
<xs:enumeration value="residential"/>
<xs:enumeration value="residentialAndCommercial"/>
<xs:enumeration value="residentialAndStreetlight"/>
<xs:enumeration value="residentialFarmService"/>
<xs:enumeration value="residentialStreetlightOthers"/>
<xs:enumeration value="windMachine"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
<xs:element name="pucNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.pucNumber"/>
<xs:element name="specialNeed" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.specialNeed"/>
<xs:element name="vip" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.vip"/>
<xs:element name="CustomerAgreements" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.CustomerAgreements">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="CustomerAccount"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CustomerAccount">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="billingCycle" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CustomerAccount.billingCycle"/>
<xs:element name="budgetBill" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CustomerAccount.budgetBill"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
<xs:element name="CustomerAgreements" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#CustomerAccount.CustomerAgreements">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="CustomerAgreement"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CustomerAgreement">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
<xs:element name="ServiceDeliveryPoints" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#CustomerAgreement.ServiceDeliveryPoints">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
```

```
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceAsset" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="amrSystem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.amrSystem"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="SDPLocation" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#SDPLocation">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="accessMethod" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#SDPLocation.accessMethod"/>
<xs:element name="occupancyDate" minOccurs="0" maxOccurs="1" type="m:AbsoluteDate"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#SDPLocation.occupancyDate"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#SDPLocation.remark"/>
<xs:element name="siteAccessProblem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#SDPLocation.siteAccessProblem"/>
<xs:element name="mainAddress" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Location.mainAddress">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress">
<xs:sequence>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="streetDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.streetDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail">
<xs:sequence>
<xs:element name="addressGeneral" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.addressGeneral"/>
<xs:element name="buildingName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.buildingName"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.code"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.name"/>
<xs:element name="number" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.number"/>
<xs:element name="prefix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.prefix"/>
<xs:element name="suffix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.suffix"/>
```

```
<xs:element name="suiteNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.suiteNumber"/>
<xs:element name="type" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.type"/>
<xs:element name="withinTownLimits" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.withinTownLimits"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="townDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.townDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail">
<xs:sequence>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.code"/>
<xs:element name="country" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.country"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.name"/>
<xs:element name="section" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.section"/>
<xs:element name="stateOrProvince" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.stateOrProvince"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ServiceCategory" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceCategory">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.description"/>
<xs:element name="kind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceCategory.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceKind">
<xs:restriction base="xs:string">
<xs:enumeration value="electricty"/>
<xs:enumeration value="gas"/>
<xs:enumeration value="heat"/>
<xs:enumeration value="internet"/>
<xs:enumeration value="other"/>
<xs:enumeration value="rates"/>
<xs:enumeration value="refuse"/>
<xs:enumeration value="sewerage"/>
<xs:enumeration value="time"/>
<xs:enumeration value="tvLicence"/>
<xs:enumeration value="water"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ServiceDeliveryPoint"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint">
```

```
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="billingCycle" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.billingCycle"/>
<xs:element name="budgetBill" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.budgetBill"/>
<xs:element name="checkBilling" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.checkBilling"/>
<xs:element name="consumptionRealEnergy" minOccurs="0" maxOccurs="1"
type="m:RealEnergy" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.consumptionRealEnergy"/>
<xs:element name="ctptReference" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.ctptReference"/>
<xs:element name="estimatedLoad" minOccurs="0" maxOccurs="1" type="m:CurrentFlow"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.estimatedLoad"/>
<xs:element name="grounded" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.grounded"/>
<xs:element name="loadMgmt" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.loadMgmt"/>
<xs:element name="nominalServiceVoltage" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.nominalServiceVoltage"/>
<xs:element name="phaseConfig" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.phaseConfig">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#PhaseConfigurationKind">
<xs:restriction base="xs:string">
<xs:enumeration value="onePhaseThreeWire"/>
<xs:enumeration value="onePhaseTwoWire"/>
<xs:enumeration value="other"/>
<xs:enumeration value="threePhaseFourWire"/>
<xs:enumeration value="threePhaseThreeWire"/>
<xs:enumeration value="threePhaseTwoWire"/>
<xs:enumeration value="twoPhaseThreeWire"/>
<xs:enumeration value="twoPhaseTwoWire"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="ratedCurrent" minOccurs="0" maxOccurs="1" type="m:CurrentFlow"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.ratedCurrent"/>
<xs:element name="ratedPower" minOccurs="0" maxOccurs="1" type="m:ActivePower"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.ratedPower"/>
<xs:element name="serviceDeliveryRemark" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.serviceDeliveryRemark"/>
<xs:element name="servicePriority" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.servicePriority"/>
<xs:element name="EndDeviceAssets" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.EndDeviceAssets">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
<xs:element name="SDPLocations" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.SDPLocations">
```

```
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
<xs:element name="ServiceCategory" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.ServiceCategory">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
<xs:element name="ServiceLocation" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.ServiceLocation">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ServiceLocation" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceLocation">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="accessMethod" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceLocation.accessMethod"/>
<xs:element name="needsInspection" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceLocation.needsInspection"/>
<xs:element name="siteAccessProblem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceLocation.siteAccessProblem"/>
<xs:element name="mainAddress" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Location.mainAddress">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress">
<xs:sequence>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="streetDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.streetDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail">
<xs:sequence>
<xs:element name="addressGeneral" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.addressGeneral"/>
<xs:element name="buildingName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.buildingName"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.code"/>
```

```
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.name"/>
<xs:element name="number" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.number"/>
<xs:element name="prefix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.prefix"/>
<xs:element name="suffix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.suffix"/>
<xs:element name="suiteNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.suiteNumber"/>
<xs:element name="type" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.type"/>
<xs:element name="withinTownLimits" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.withinTownLimits"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="townDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.townDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail">
<xs:sequence>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.code"/>
<xs:element name="country" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.country"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.name"/>
<xs:element name="section" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.section"/>
<xs:element name="stateOrProvince" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.stateOrProvince"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="AbsoluteDate" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#AbsoluteDate">
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="CurrentFlow" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#CurrentFlow">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="RealEnergy" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RealEnergy">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="ActivePower" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ActivePower">
<xs:restriction base="xs:float"/>
</xs:simpleType>
</xs:schema>
```

H.3 MeterAssetReading

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/MeterAssetReading#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/MeterAssetReading#">
<xs:element name="MeterAssetReading" type="m:MeterAssetReading"/>
<xs:complexType name="MeterAssetReading">
<xs:sequence>
<xs:element name="MeterServiceWork" type="m:MeterServiceWork" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ServiceLocation" type="m:ServiceLocation" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="MeterServiceWork"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterServiceWork">
<xs:sequence>
<xs:element name="kind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Work.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#WorkKind">
<xs:restriction base="xs:string">
<xs:enumeration value="construction"/>
<xs:enumeration value="disconnect"/>
<xs:enumeration value="inspection"/>
<xs:enumeration value="maintenance"/>
<xs:enumeration value="meter"/>
<xs:enumeration value="other"/>
<xs:enumeration value="reconnect"/>
<xs:enumeration value="service"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="priority" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Work.priority"/>
<xs:element name="requestDateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Work.requestDateTime"/>
<xs:element name="MeterAsset" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterServiceWork.MeterAsset">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="serialNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.serialNumber"/>
<xs:element name="MeterReadings" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset.MeterReadings">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterReading">
<xs:sequence>
<xs:element name="Readings" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterReading.Readings">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading">
<xs:sequence>
<xs:element name="timeStamp" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeasurementValue.timeStamp"/>
<xs:element name="value" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.value"/>
<xs:element name="ReadingQualities" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.ReadingQualities">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality">
<xs:sequence>
```

```
<xs:element name="quality" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality.quality"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ReadingType" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.ReadingType">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Document.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ServiceLocation" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceLocation">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="accessMethod" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceLocation.accessMethod"/>
<xs:element name="needsInspection" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceLocation.needsInspection"/>
<xs:element name="siteAccessProblem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceLocation.siteAccessProblem"/>
<xs:element name="mainAddress" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Location.mainAddress">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress">
<xs:sequence>
<xs:element name="streetDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.streetDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail">
<xs:sequence>
```

```
<xs:element name="addressGeneral" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.addressGeneral"/>
<xs:element name="buildingName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.buildingName"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.code"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.name"/>
<xs:element name="number" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.number"/>
<xs:element name="prefix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.prefix"/>
<xs:element name="suffix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.suffix"/>
<xs:element name="suiteNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.suiteNumber"/>
<xs:element name="type" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.type"/>
<xs:element name="withinTownLimits" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.withinTownLimits"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="townDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.townDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail">
<xs:sequence>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.code"/>
<xs:element name="country" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.country"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.name"/>
<xs:element name="section" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.section"/>
<xs:element name="stateOrProvince" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.stateOrProvince"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:schema>
```

H.4 EndDeviceControls

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/EndDeviceControls#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/EndDeviceControls#">
<xs:element name="EndDeviceControls" type="m:EndDeviceControls"/>
<xs:complexType name="EndDeviceControls">
<xs:sequence>
```

```
<xs:element name="EndDeviceControl" type="m:EndDeviceControl" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceControl"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceControl">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.description"/>
<xs:element name="drProgramLevel" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceControl.drProgramLevel"/>
<xs:element name="drProgramMandatory" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceControl.drProgramMandatory"/>
<xs:element name="priceSignal" minOccurs="0" maxOccurs="1" type="m:FloatQuantity"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceControl.priceSignal"/>
<xs:element name="type" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceControl.type"/>
<xs:element name="DemandResponseProgram" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceControl.DemandResponseProgram">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#DemandResponseProgram">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
<xs:element name="type" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DemandResponseProgram.type"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="EndDeviceAsset" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceControl.EndDeviceAsset">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="EndDeviceGroup" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceControl.EndDeviceGroup">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceGroup">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="groupAddress" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceGroup.groupAddress"/>
</xs:sequence>
</xs:complexType>
</xs:element>
```

```
<xs:element name="scheduledInterval" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceControl.scheduledInterval">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#DateTimeInterval">
<xs:sequence>
<xs:element name="end" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DateTimeInterval.end"/>
<xs:element name="start" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DateTimeInterval.start"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="FloatQuantity" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#FloatQuantity">
<xs:restriction base="xs:float"/>
</xs:simpleType>
</xs:schema>
```

H.5 EndDeviceEvents

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/EndDeviceEvents#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/EndDeviceEvents#">
<xs:element name="EndDeviceEvents" type="m:EndDeviceEvents"/>
<xs:complexType name="EndDeviceEvents">
<xs:sequence>
<xs:element name="EndDeviceEvent" type="m:EndDeviceEvent" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceEvent" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceEvent">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.category"/>
<xs:element name="createdDateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.createdDateTime"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.description"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.reason"/>
<xs:element name="severity" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.severity"/>
<xs:element name="userID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceEvent.userID"/>
<xs:element name="Assets" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.Assets">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
```

```
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:schema>
```

H.6 MeterReadings

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/MeterReadings#" elementFormDefault="qualified"
attributeFormDefault="unqualified" xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/MeterReadings#">
<xs:element name="MeterReadings" type="m:MeterReadings"/>
<xs:complexType name="MeterReadings">
<xs:sequence>
<xs:element name="MeterReading" type="m:MeterReading" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ReadingType" type="m:ReadingType" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceEvent" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceEvent">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.category"/>
<xs:element name="createdDateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.createdDateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.reason"/>
<xs:element name="severity" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.severity"/>
<xs:element name="userID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceEvent.userID"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="MeterAsset" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
```

```
<xs:element name="status" minOccurs="0" maxOccurs="1" type="m:Status"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.status"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="MeterReading" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterReading">
<xs:sequence>
<xs:element name="CustomerAgreement" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterReading.CustomerAgreement">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#CustomerAgreement">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="EndDeviceEvents" minOccurs="0" maxOccurs="unbounded"
type="m:EndDeviceEvent" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterReading.EndDeviceEvents"/>
<xs:element name="IntervalBlocks" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterReading.IntervalBlocks">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IntervalBlock">
<xs:sequence>
<xs:element name="IntervalReadings" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IntervalBlock.IntervalReadings">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IntervalReading">
<xs:sequence>
<xs:element name="timeStamp" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeasurementValue.timeStamp"/>
<xs:element name="value" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IntervalReading.value"/>
<xs:element name="ReadingQualities" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IntervalReading.ReadingQualities">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality">
<xs:sequence>
<xs:element name="quality" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality.quality"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Pending" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IntervalBlock.Pending">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending">
<xs:sequence>
<xs:element name="multiplyBeforeAdd" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.multiplyBeforeAdd"/>
<xs:element name="offset" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.offset"/>
<xs:element name="scalarDenominator" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.scalarDenominator"/>
<xs:element name="scalarFloat" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.scalarFloat"/>
<xs:element name="scalarNumerator" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.scalarNumerator"/>
<xs:element name="ReadingType" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.ReadingType">
<xs:complexType sawsdl:modelReference="">
```

```
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ReadingType" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IntervalBlock.ReadingType">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="MeterAsset" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterReading.MeterAsset">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Readings" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterReading.Readings">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading">
<xs:sequence>
<xs:element name="timeStamp" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeasurementValue.timeStamp"/>
<xs:element name="value" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.value"/>
<xs:element name="ReadingQualities" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.ReadingQualities">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality">
<xs:sequence>
<xs:element name="quality" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality.quality"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ReadingType" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.ReadingType">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ServiceDeliveryPoint" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterReading.ServiceDeliveryPoint">
```

```
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="Pending" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#Pending">
<xs:sequence>
<xs:element name="multiplyBeforeAdd" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.multiplyBeforeAdd"/>
<xs:element name="offset" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.offset"/>
<xs:element name="scalarDenominator" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.scalarDenominator"/>
<xs:element name="scalarFloat" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.scalarFloat"/>
<xs:element name="scalarNumerator" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.scalarNumerator"/>
<xs:element name="ReadingType" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Pending.ReadingType">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ReadingQuality" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingQuality">
<xs:sequence>
<xs:element name="quality" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality.quality"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ReadingType" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingType">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.aliasName"/>
<xs:element name="channelNumber" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.channelNumber"/>
<xs:element name="defaultQuality" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.defaultQuality"/>
<xs:element name="defaultValueDataType" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingType.defaultValueDataType"/>
<xs:element name="dynamicConfiguration" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingType.dynamicConfiguration"/>
<xs:element name="intervalLength" minOccurs="0" maxOccurs="1" type="m:Seconds"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.intervalLength"/>
<xs:element name="kind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingKind">
<xs:restriction base="xs:string">
<xs:enumeration value="current"/>
```

```
<xs:enumeration value="currentAngle"/>
<xs:enumeration value="date"/>
<xs:enumeration value="demand"/>
<xs:enumeration value="energy"/>
<xs:enumeration value="other"/>
<xs:enumeration value="phaseAngle"/>
<xs:enumeration value="power"/>
<xs:enumeration value="powerFactor"/>
<xs:enumeration value="pressure"/>
<xs:enumeration value="time"/>
<xs:enumeration value="voltage"/>
<xs:enumeration value="voltageAngle"/>
<xs:enumeration value="volume"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="multiplier" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.multiplier">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#UnitMultiplier">
<xs:restriction base="xs:string">
<xs:enumeration value="G"/>
<xs:enumeration value="M"/>
<xs:enumeration value="T"/>
<xs:enumeration value="c"/>
<xs:enumeration value="d"/>
<xs:enumeration value="k"/>
<xs:enumeration value="m"/>
<xs:enumeration value="micro"/>
<xs:enumeration value="n"/>
<xs:enumeration value="none"/>
<xs:enumeration value="p"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
<xs:element name="reverseChronology" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.reverseChronology"/>
<xs:element name="unit" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.unit">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#UnitSymbol">
<xs:restriction base="xs:string">
<xs:enumeration value="A"/>
<xs:enumeration value="F"/>
<xs:enumeration value="H"/>
<xs:enumeration value="Hz"/>
<xs:enumeration value="Hz-1"/>
<xs:enumeration value="J"/>
<xs:enumeration value="J/s"/>
<xs:enumeration value="N"/>
<xs:enumeration value="Pa"/>
<xs:enumeration value="S"/>
<xs:enumeration value="V"/>
<xs:enumeration value="V/VAr"/>
<xs:enumeration value="VA"/>
<xs:enumeration value="VAh"/>
<xs:enumeration value="VAr"/>
<xs:enumeration value="VArh"/>
<xs:enumeration value="W"/>
<xs:enumeration value="W/Hz"/>
<xs:enumeration value="W/s"/>
<xs:enumeration value="Wh"/>
```

```
<xs:enumeration value="deg"/>
<xs:enumeration value="g"/>
<xs:enumeration value="h"/>
<xs:enumeration value="kg/J"/>
<xs:enumeration value="m"/>
<xs:enumeration value="m2"/>
<xs:enumeration value="m3"/>
<xs:enumeration value="min"/>
<xs:enumeration value="none"/>
<xs:enumeration value="ohm"/>
<xs:enumeration value="rad"/>
<xs:enumeration value="s"/>
<xs:enumeration value="s-1"/>
<xs:enumeration value="Â°C"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="Status" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="Seconds" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#Seconds">
<xs:restriction base="xs:float"/>
</xs:simpleType>
</xs:schema>
```

H.7 MeterReadSchedule

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/MeterReadSchedule#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/MeterReadSchedule#">
<xs:element name="MeterReadSchedule" type="m:MeterReadSchedule"/>
<xs:complexType name="MeterReadSchedule">
<xs:sequence>
<xs:element name="CustomerAccount" type="m:CustomerAccount" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="CustomerAgreement" type="m:CustomerAgreement" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="EndDeviceAsset" type="m:EndDeviceAsset" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="EndDeviceGroup" type="m:EndDeviceGroup" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ReadingType" type="m:ReadingType" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="TimeSchedule" type="m:TimeSchedule" minOccurs="0"
maxOccurs="unbounded"/>
```

```
</xs:sequence>
</xs:complexType>
<xs:complexType name="CustomerAccount"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CustomerAccount">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="CustomerAgreement"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CustomerAgreement">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceAsset" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceGroup" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceGroup">
<xs:sequence>
<xs:element name="groupAddress" minOccurs="1" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceGroup.groupAddress"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ReadingType" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingType">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="TimeSchedule" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#TimeSchedule">
<xs:sequence>
<xs:element name="disabled" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimeSchedule.disabled"/>
<xs:element name="offset" minOccurs="0" maxOccurs="1" type="m:Seconds"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimeSchedule.offset"/>
<xs:element name="recurrencePattern" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimeSchedule.recurrencePattern"/>
<xs:element name="recurrencePeriod" minOccurs="0" maxOccurs="1" type="m:Seconds"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimeSchedule.recurrencePeriod"/>
<xs:element name="scheduleInterval" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimeSchedule.scheduleInterval">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#DateTimeInterval">
<xs:sequence>
<xs:element name="end" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DateTimeInterval.end"/>
<xs:element name="start" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DateTimeInterval.start"/>
</xs:sequence>
</xs:complexType>
</xs:element>
```

```
<xs:element name="TimePoints" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimeSchedule.TimePoints">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimePoint">
<xs:sequence>
<xs:element name="absoluteTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimePoint.absoluteTime"/>
<xs:element name="relativeTimeInterval" minOccurs="0" maxOccurs="1" type="m:Seconds"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimePoint.relativeTimeInterval"/>
<xs:element name="sequenceNumber" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimePoint.sequenceNumber"/>
<xs:element name="window" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TimePoint.window">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#DateTimeInterval">
<xs:sequence>
<xs:element name="end" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DateTimeInterval.end"/>
<xs:element name="start" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DateTimeInterval.start"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="Seconds" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#Seconds">
<xs:restriction base="xs:float"/>
</xs:simpleType>
</xs:schema>
```

H.8 MeterServiceRequest

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/MeterServiceRequests#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/MeterServiceRequests#">
<xs:element name="MeterServiceRequests" type="m:MeterServiceRequests"/>
<xs:complexType name="MeterServiceRequests">
<xs:sequence>
<xs:element name="MeterServiceWork" type="m:MeterServiceWork" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ReadingType" type="m:ReadingType" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="MeterAsset" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterAsset">
<xs:sequence>
<xs:element name="amrSystem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.amrSystem"/>
<xs:element name="formNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset.formNumber"/>
<xs:element name="kH" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset.kH"/>
```

```
<xs:element name="kR" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset.kR"/>
<xs:element name="serialNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.serialNumber"/>
<xs:element name="MeterReadings" minOccurs="0" maxOccurs="unbounded"
type="m:MeterReading" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterAsset.MeterReadings"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="MeterReading" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterReading">
<xs:sequence>
<xs:element name="Readings" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterReading.Readings">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading">
<xs:sequence>
<xs:element name="timeStamp" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeasurementValue.timeStamp"/>
<xs:element name="value" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.value"/>
<xs:element name="ReadingQualities" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.ReadingQualities">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality">
<xs:sequence>
<xs:element name="quality" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingQuality.quality"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ReadingType" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reading.ReadingType">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="MeterServiceWork"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterServiceWork">
<xs:sequence>
<xs:element name="category" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Document.category"/>
<xs:element name="kind" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Work.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#WorkKind">
<xs:restriction base="xs:string">
<xs:enumeration value="construction"/>
<xs:enumeration value="disconnect"/>
<xs:enumeration value="inspection"/>
<xs:enumeration value="maintenance"/>
<xs:enumeration value="meter"/>
<xs:enumeration value="other"/>
<xs:enumeration value="reconnect"/>
<xs:enumeration value="service"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
```

```
<xs:element name="lastModifiedDateTime" minOccurs="0" maxOccurs="1"
type="xs:dateTime" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#Document.lastModifiedDateTime"/>
<xs:element name="priority" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Work.priority"/>
<xs:element name="requestDateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Work.requestDateTime"/>
<xs:element name="ActivityRecords" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Document.ActivityRecords">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord">
<xs:sequence>
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.category"/>
<xs:element name="createdDateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.createdDateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.reason"/>
<xs:element name="severity" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.severity"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Customers" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Work.Customers">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer">
<xs:sequence>
<xs:element name="kind" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CustomerKind">
<xs:restriction base="xs:string">
<xs:enumeration value="commercialIndustrial"/>
<xs:enumeration value="energyServiceScheduler"/>
<xs:enumeration value="energyServiceSupplier"/>
<xs:enumeration value="internalUse"/>
<xs:enumeration value="other"/>
<xs:enumeration value="pumpingLoad"/>
<xs:enumeration value="residential"/>
<xs:enumeration value="residentialAndCommercial"/>
<xs:enumeration value="residentialAndStreetlight"/>
<xs:enumeration value="residentialFarmService"/>
<xs:enumeration value="residentialStreetlightOthers"/>
<xs:enumeration value="windMachine"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="pucNumber" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.pucNumber"/>
<xs:element name="specialNeed" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.specialNeed"/>
<xs:element name="vip" minOccurs="1" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.vip"/>
<xs:element name="CustomerAgreements" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.CustomerAgreements">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#CustomerAgreement">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
```

```
<xs:element name="ServiceDeliveryPoints" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#CustomerAgreement.ServiceDeliveryPoints">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint">
<xs:sequence>
<xs:element name="billingCycle" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.billingCycle"/>
<xs:element name="budgetBill" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.budgetBill"/>
<xs:element name="checkBilling" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.checkBilling"/>
<xs:element name="consumptionRealEnergy" minOccurs="0" maxOccurs="1"
type="m:RealEnergy" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.consumptionRealEnergy"/>
<xs:element name="ctptReference" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.ctptReference"/>
<xs:element name="estimatedLoad" minOccurs="0" maxOccurs="1" type="m:CurrentFlow"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.estimatedLoad"/>
<xs:element name="grounded" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.grounded"/>
<xs:element name="loadMgmt" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.loadMgmt"/>
<xs:element name="nominalServiceVoltage" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.nominalServiceVoltage"/>
<xs:element name="phaseConfig" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.phaseConfig">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#PhaseConfigurationKind">
<xs:restriction base="xs:string">
<xs:enumeration value="onePhaseThreeWire"/>
<xs:enumeration value="onePhaseTwoWire"/>
<xs:enumeration value="other"/>
<xs:enumeration value="threePhaseFourWire"/>
<xs:enumeration value="threePhaseThreeWire"/>
<xs:enumeration value="threePhaseTwoWire"/>
<xs:enumeration value="twoPhaseThreeWire"/>
<xs:enumeration value="twoPhaseTwoWire"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="ratedCurrent" minOccurs="0" maxOccurs="1" type="m:CurrentFlow"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.ratedCurrent"/>
<xs:element name="ratedPower" minOccurs="0" maxOccurs="1" type="m:ActivePower"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceDeliveryPoint.ratedPower"/>
<xs:element name="serviceDeliveryRemark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.serviceDeliveryRemark"/>
<xs:element name="servicePriority" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.servicePriority"/>
<xs:element name="ServiceLocation" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceDeliveryPoint.ServiceLocation">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceLocation">
<xs:sequence>
<xs:element name="accessMethod" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceLocation.accessMethod"/>
```

```
<xs:element name="needsInspection" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ServiceLocation.needsInspection"/>
<xs:element name="siteAccessProblem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ServiceLocation.siteAccessProblem"/>
<xs:element name="mainAddress" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Location.mainAddress">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress">
<xs:sequence>
<xs:element name="streetDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.streetDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail">
<xs:sequence>
<xs:element name="addressGeneral" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.addressGeneral"/>
<xs:element name="buildingName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.buildingName"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.code"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.name"/>
<xs:element name="number" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.number"/>
<xs:element name="prefix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.prefix"/>
<xs:element name="suffix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.suffix"/>
<xs:element name="suiteNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.suiteNumber"/>
<xs:element name="type" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.type"/>
<xs:element name="withinTownLimits" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetDetail.withinTownLimits"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="townDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#StreetAddress.townDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail">
<xs:sequence>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.code"/>
<xs:element name="country" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.country"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.name"/>
<xs:element name="section" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.section"/>
<xs:element name="stateOrProvince" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#TownDetail.stateOrProvince"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
```

```
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Customer.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="MeterAsset" minOccurs="1" maxOccurs="1" type="m:MeterAsset"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterServiceWork.MeterAsset"/>
<xs:element name="OldMeterAsset" minOccurs="0" maxOccurs="1" type="m:MeterAsset"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#MeterServiceWork.OldMeterAsset"/>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Document.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ReadingType" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingType">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
<xs:element name="unit" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.unit">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#UnitSymbol">
<xs:restriction base="xs:string">
<xs:enumeration value="A"/>
<xs:enumeration value="F"/>
<xs:enumeration value="H"/>
<xs:enumeration value="Hz"/>
<xs:enumeration value="Hz-1"/>
<xs:enumeration value="J"/>
<xs:enumeration value="J/s"/>
<xs:enumeration value="N"/>
```

```
<xs:enumeration value="Pa"/>
<xs:enumeration value="S"/>
<xs:enumeration value="V"/>
<xs:enumeration value="V/VAr"/>
<xs:enumeration value="VA"/>
<xs:enumeration value="VAh"/>
<xs:enumeration value="VAr"/>
<xs:enumeration value="VArh"/>
<xs:enumeration value="W"/>
<xs:enumeration value="W/Hz"/>
<xs:enumeration value="W/s"/>
<xs:enumeration value="Wh"/>
<xs:enumeration value="deg"/>
<xs:enumeration value="g"/>
<xs:enumeration value="h"/>
<xs:enumeration value="kg/J"/>
<xs:enumeration value="m"/>
<xs:enumeration value="m2"/>
<xs:enumeration value="m3"/>
<xs:enumeration value="min"/>
<xs:enumeration value="none"/>
<xs:enumeration value="ohm"/>
<xs:enumeration value="rad"/>
<xs:enumeration value="s"/>
<xs:enumeration value="s-1"/>
<xs:enumeration value="Â°C"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="CurrentFlow" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#CurrentFlow">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="RealEnergy" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RealEnergy">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="ActivePower" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ActivePower">
<xs:restriction base="xs:float"/>
</xs:simpleType>
</xs:schema>
```

H.9 MeterSystemEvent

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/MeterSystemEvents#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/MeterSystemEvents#">
<xs:element name="MeterSystemEvents" type="m:MeterSystemEvents"/>
<xs:complexType name="MeterSystemEvents">
<xs:sequence>
<xs:element name="ComMediaAsset" type="m:ComMediaAsset" minOccurs="0"
maxOccurs="unbounded"/>
```

```
<xs:element name="EndDeviceAsset" type="m:EndDeviceAsset" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="EndDeviceEvent" type="m:EndDeviceEvent" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ComMediaAsset" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ComMediaAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="serialNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.serialNumber"/>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceAsset" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="amrSystem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.amrSystem"/>
<xs:element name="serialNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.serialNumber"/>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceEvent" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceEvent">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
```

```
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.category"/>
<xs:element name="createdDateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.createdDateTime"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.description"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.reason"/>
<xs:element name="severity" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.severity"/>
<xs:element name="userID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceEvent.userID"/>
<xs:element name="Assets" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.Assets">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="serialNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.serialNumber"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="DeviceFunction" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceEvent.DeviceFunction">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DeviceFunction">
<xs:sequence>
<xs:element name="configID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.configID"/>
<xs:element name="disabled" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DeviceFunction.disabled"/>
<xs:element name="firmwareID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.firmwareID"/>
<xs:element name="hardwareID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.hardwareID"/>
<xs:element name="programID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.programID"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:schema>
```

H.10 EndDeviceAssets

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/EndDeviceAssets#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/EndDeviceAssets#">
<xs:element name="EndDeviceAssets" type="m:EndDeviceAssets"/>
<xs:complexType name="EndDeviceAssets">
<xs:sequence>
<xs:element name="ComFunction" type="m:ComFunction" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ConnectDisconnectFunction" type="m:ConnectDisconnectFunction"
minOccurs="0" maxOccurs="unbounded"/>
```

```
<xs:element name="ElectricMeteringFunction" type="m:ElectricMeteringFunction"
minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="EndDeviceAsset" type="m:EndDeviceAsset" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="ReadingType" type="m:ReadingType" minOccurs="0"
maxOccurs="unbounded"/>
<xs:element name="Register" type="m:Register" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ComFunction" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ComFunction">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="amrAddress" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ComFunction.amrAddress"/>
<xs:element name="amrRouter" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ComFunction.amrRouter"/>
<xs:element name="configID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.configID"/>
<xs:element name="firmwareID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.firmwareID"/>
<xs:element name="hardwareID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.hardwareID"/>
<xs:element name="password" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.password"/>
<xs:element name="programID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.programID"/>
<xs:element name="twoWay" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ComFunction.twoWay"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ConnectDisconnectFunction"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ConnectDisconnectFunction">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="disabled" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DeviceFunction.disabled"/>
<xs:element name="eventCount" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction.eventCount"/>
<xs:element name="isConnected" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction.isConnected"/>
<xs:element name="isDelayedDiscon" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction.isDelayedDiscon"/>
<xs:element name="isLocalAutoDisconOp" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction.isLocalAutoDisconOp"/>
<xs:element name="isLocalAutoReconOp" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction.isLocalAutoReconOp"/>
<xs:element name="isRemoteAutoDisconOp" minOccurs="0" maxOccurs="1"
type="xs:boolean" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction.isRemoteAutoDisconOp"/>
<xs:element name="isRemoteAutoReconOp" minOccurs="0" maxOccurs="1"
type="xs:boolean" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction.isRemoteAutoReconOp"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
```

```
<xs:element name="rcdInfo" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction.rcdInfo">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo">
<xs:sequence>
<xs:element name="armedTimeout" minOccurs="0" maxOccurs="1" type="m:Seconds"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.armedTimeout"/>
<xs:element name="customerVoltageLimit" minOccurs="0" maxOccurs="1" type="m:Voltage"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.customerVoltageLimit"/>
<xs:element name="energyLimit" minOccurs="0" maxOccurs="1" type="m:RealEnergy"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.energyLimit"/>
<xs:element name="energyUsageStartDateTime" minOccurs="1" maxOccurs="1"
type="xs:dateTime" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.energyUsageStartDateTime"/>
<xs:element name="energyUsageWarning" minOccurs="0" maxOccurs="1"
type="m:RealEnergy" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.energyUsageWarning"/>
<xs:element name="isArmConnect" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.isArmConnect"/>
<xs:element name="isArmDisconnect" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.isArmDisconnect"/>
<xs:element name="isEnergyLimiting" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.isEnergyLimiting"/>
<xs:element name="needsPowerLimitCheck" minOccurs="0" maxOccurs="1"
type="xs:boolean" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.needsPowerLimitCheck"/>
<xs:element name="needsVoltageLimitCheck" minOccurs="0" maxOccurs="1"
type="xs:boolean" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.needsVoltageLimitCheck"/>
<xs:element name="powerLimit" minOccurs="0" maxOccurs="1" type="m:ActivePower"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.powerLimit"/>
<xs:element name="usePushbutton" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#RemoteConnectDisconnectInfo.usePushbutton"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Registers" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DeviceFunction.Registers">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ElectricMeteringFunction"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricMeteringFunction">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="billingMultiplier" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.billingMultiplier"/>
```

```
<xs:element name="billingMultiplierApplied" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.billingMultiplierApplied"/>
<xs:element name="configID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.configID"/>
<xs:element name="currentRating" minOccurs="0" maxOccurs="1" type="m:CurrentFlow"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.currentRating"/>
<xs:element name="demandMultiplier" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.demandMultiplier"/>
<xs:element name="demandMultiplierApplied" minOccurs="0" maxOccurs="1"
type="xs:boolean" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.demandMultiplierApplied"/>
<xs:element name="disabled" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DeviceFunction.disabled"/>
<xs:element name="firmwareID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.firmwareID"/>
<xs:element name="hardwareID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.hardwareID"/>
<xs:element name="kWhMultiplier" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.kWhMultiplier"/>
<xs:element name="kWMultiplier" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.kWMultiplier"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
<xs:element name="password" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.password"/>
<xs:element name="programID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.programID"/>
<xs:element name="transformerCTRatio" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.transformerCTRatio"/>
<xs:element name="transformerRatiosApplied" minOccurs="0" maxOccurs="1"
type="xs:boolean" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.transformerRatiosApplied"/>
<xs:element name="transformerVTRatio" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.transformerVTRatio"/>
<xs:element name="voltageRating" minOccurs="0" maxOccurs="1" type="m:Voltage"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction.voltageRating"/>
<xs:element name="Registers" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DeviceFunction.Registers">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceAsset" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="amrSystem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.amrSystem"/>
<xs:element name="category" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.category"/>
```

```
<xs:element name="demandResponse" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset.demandResponse"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.description"/>
<xs:element name="disconnect" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.disconnect"/>
<xs:element name="dstEnabled" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.dstEnabled"/>
<xs:element name="formNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset.formNumber"/>
<xs:element name="installationDate" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.installationDate"/>
<xs:element name="kH" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset.kH"/>
<xs:element name="kR" minOccurs="0" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#MeterAsset.kR"/>
<xs:element name="loadControl" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.loadControl"/>
<xs:element name="lotNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.lotNumber"/>
<xs:element name="manufacturedDate" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.manufacturedDate"/>
<xs:element name="metrology" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.metrology"/>
<xs:element name="outageReport" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.outageReport"/>
<xs:element name="purchasePrice" minOccurs="0" maxOccurs="1" type="m:Money"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.purchasePrice"/>
<xs:element name="readRequest" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.readRequest"/>
<xs:element name="relayCapable" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.relayCapable"/>
<xs:element name="reverseFlowHandling" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset.reverseFlowHandling"/>
<xs:element name="serialNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.serialNumber"/>
<xs:element name="timeZoneOffset" minOccurs="0" maxOccurs="1" type="m:Minutes"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.timeZoneOffset"/>
<xs:element name="ActivityRecords" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.ActivityRecords">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord">
<xs:sequence>
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.category"/>
<xs:element name="createdDateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.createdDateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.reason"/>
<xs:element name="severity" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.severity"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Customer" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.Customer">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
```

```
<xs:choice minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.DeviceFunctions">
<xs:element name="ComFunction" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ComFunction">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
<xs:element name="ConnectDisconnectFunction" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ConnectDisconnectFunction">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
<xs:element name="ElectricMeteringFunction" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricMeteringFunction">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:choice>
<xs:element name="ElectricalInfos" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.ElectricalInfos">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo">
<xs:sequence>
<xs:element name="mRID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="b" minOccurs="0" maxOccurs="1" type="m:Susceptance"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.b"/>
<xs:element name="b0" minOccurs="0" maxOccurs="1" type="m:Susceptance"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.b0"/>
<xs:element name="bil" minOccurs="0" maxOccurs="1" type="m:Voltage"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.bil"/>
<xs:element name="frequency" minOccurs="0" maxOccurs="1" type="m:Frequency"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.frequency"/>
<xs:element name="g" minOccurs="0" maxOccurs="1" type="m:Conductance"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.g"/>
<xs:element name="g0" minOccurs="0" maxOccurs="1" type="m:Conductance"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.g0"/>
<xs:element name="phaseCount" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.phaseCount"/>
<xs:element name="r" minOccurs="0" maxOccurs="1" type="m:Resistance"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.r"/>
<xs:element name="r0" minOccurs="0" maxOccurs="1" type="m:Resistance"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.r0"/>
<xs:element name="ratedApparentPower" minOccurs="0" maxOccurs="1"
type="m:ApparentPower" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectricalInfo.ratedApparentPower"/>
<xs:element name="ratedCurrent" minOccurs="0" maxOccurs="1" type="m:CurrentFlow"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.ratedCurrent"/>
<xs:element name="ratedVoltage" minOccurs="0" maxOccurs="1" type="m:Voltage"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.ratedVoltage"/>
<xs:element name="wireCount" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.wireCount"/>
<xs:element name="x" minOccurs="0" maxOccurs="1" type="m:Reactance"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.x"/>
<xs:element name="x0" minOccurs="0" maxOccurs="1" type="m:Reactance"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectricalInfo.x0"/>
</xs:sequence>
</xs:complexType>
```

```
</xs:element>
<xs:element name="ElectronicAddresses" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.ElectronicAddresses">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ElectronicAddress">
<xs:sequence>
<xs:element name="lan" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectronicAddress.lan"/>
<xs:element name="password" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectronicAddress.password"/>
<xs:element name="radio" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectronicAddress.radio"/>
<xs:element name="userID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ElectronicAddress.userID"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="EndDeviceGroups" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset.EndDeviceGroups">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceGroup">
<xs:sequence>
<xs:element name="groupAddress" minOccurs="1" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceGroup.groupAddress"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="EndDeviceModel" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.EndDeviceModel">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceModel">
<xs:sequence>
<xs:element name="corporateStandardKind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#AssetModel.corporateStandardKind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#CorporateStandardKind">
<xs:restriction base="xs:string">
<xs:enumeration value="experimental"/>
<xs:enumeration value="other"/>
<xs:enumeration value="standard"/>
<xs:enumeration value="underEvaluation"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="modelNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetModel.modelNumber"/>
<xs:element name="modelVersion" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetModel.modelVersion"/>
<xs:element name="usageKind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetModel.usageKind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#AssetModelUsageKind">
<xs:restriction base="xs:string">
<xs:enumeration value="customerSubstation"/>
<xs:enumeration value="distributionOverhead"/>
<xs:enumeration value="distributionUnderground"/>
<xs:enumeration value="other"/>
<xs:enumeration value="streetlight"/>
<xs:enumeration value="substation"/>
<xs:enumeration value="transmission"/>
```

```
<xs:enumeration value="unknown"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Seals" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetContainer.Seals">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Seal">
<xs:sequence>
<xs:element name="appliedDateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Seal.appliedDateTime"/>
<xs:element name="condition" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Seal.condition">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#SealConditionKind">
<xs:restriction base="xs:string">
<xs:enumeration value="broken"/>
<xs:enumeration value="locked"/>
<xs:enumeration value="missing"/>
<xs:enumeration value="open"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="kind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Seal.kind">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
<xs:element name="sealNumber" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Seal.sealNumber"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ServiceLocation" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.ServiceLocation">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ReadingType" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingType">
<xs:sequence>
```

```
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.aliasName"/>
<xs:element name="channelNumber" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.channelNumber"/>
<xs:element name="defaultQuality" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.defaultQuality"/>
<xs:element name="defaultValueDataType" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingType.defaultValueDataType"/>
<xs:element name="dynamicConfiguration" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#ReadingType.dynamicConfiguration"/>
<xs:element name="intervalLength" minOccurs="0" maxOccurs="1" type="m:Seconds"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.intervalLength"/>
<xs:element name="kind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingKind">
<xs:restriction base="xs:string">
<xs:enumeration value="current"/>
<xs:enumeration value="currentAngle"/>
<xs:enumeration value="date"/>
<xs:enumeration value="demand"/>
<xs:enumeration value="energy"/>
<xs:enumeration value="other"/>
<xs:enumeration value="phaseAngle"/>
<xs:enumeration value="power"/>
<xs:enumeration value="powerFactor"/>
<xs:enumeration value="pressure"/>
<xs:enumeration value="time"/>
<xs:enumeration value="voltage"/>
<xs:enumeration value="voltageAngle"/>
<xs:enumeration value="volume"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="multiplier" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.multiplier">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#UnitMultiplier">
<xs:restriction base="xs:string">
<xs:enumeration value="G"/>
<xs:enumeration value="M"/>
<xs:enumeration value="T"/>
<xs:enumeration value="c"/>
<xs:enumeration value="d"/>
<xs:enumeration value="k"/>
<xs:enumeration value="m"/>
<xs:enumeration value="micro"/>
<xs:enumeration value="n"/>
<xs:enumeration value="none"/>
<xs:enumeration value="p"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.name"/>
<xs:element name="reverseChronology" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.reverseChronology"/>
<xs:element name="unit" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ReadingType.unit">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#UnitSymbol">
```

```
<xs:restriction base="xs:string">
<xs:enumeration value="A"/>
<xs:enumeration value="F"/>
<xs:enumeration value="H"/>
<xs:enumeration value="Hz"/>
<xs:enumeration value="Hz-1"/>
<xs:enumeration value="J"/>
<xs:enumeration value="J/s"/>
<xs:enumeration value="N"/>
<xs:enumeration value="Pa"/>
<xs:enumeration value="S"/>
<xs:enumeration value="V"/>
<xs:enumeration value="V/VAr"/>
<xs:enumeration value="VA"/>
<xs:enumeration value="VAh"/>
<xs:enumeration value="VAr"/>
<xs:enumeration value="VArh"/>
<xs:enumeration value="W"/>
<xs:enumeration value="W/Hz"/>
<xs:enumeration value="W/s"/>
<xs:enumeration value="Wh"/>
<xs:enumeration value="deg"/>
<xs:enumeration value="g"/>
<xs:enumeration value="h"/>
<xs:enumeration value="kg/J"/>
<xs:enumeration value="m"/>
<xs:enumeration value="m2"/>
<xs:enumeration value="m3"/>
<xs:enumeration value="min"/>
<xs:enumeration value="none"/>
<xs:enumeration value="ohm"/>
<xs:enumeration value="rad"/>
<xs:enumeration value="s"/>
<xs:enumeration value="s-1"/>
<xs:enumeration value="Â°C"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="Register" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#Register">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="leftDigitCount" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Register.leftDigitCount"/>
<xs:element name="rightDigitCount" minOccurs="0" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Register.rightDigitCount"/>
<xs:element name="ReadingType" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Register.ReadingType">
<xs:complexType sawsdl:modelReference="">
<xs:attribute name="ref" type="xs:string"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="Conductance" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#Conductance">
<xs:restriction base="xs:float"/>
</xs:simpleType>
```

```
<xs:simpleType name="ApparentPower" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ApparentPower">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="Money" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Money">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="Reactance" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Reactance">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="Frequency" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Frequency">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="Susceptance" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Susceptance">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="ActivePower" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivePower">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="Voltage" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Voltage">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="CurrentFlow" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#CurrentFlow">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="Resistance" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Resistance">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="Seconds" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Seconds">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="Minutes" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Minutes">
<xs:restriction base="xs:float"/>
</xs:simpleType>
<xs:simpleType name="RealEnergy" sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#RealEnergy">
<xs:restriction base="xs:float"/>
</xs:simpleType>
</xs:schema>
```

H.11 EndDeviceFirmware

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/EndDeviceFirmware#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/EndDeviceFirmware#">
<xs:element name="EndDeviceFirmware" type="m:EndDeviceFirmware"/>
```

```
<xs:complexType name="EndDeviceFirmware">
<xs:sequence>
<xs:element name="EndDeviceAsset" type="m:EndDeviceAsset" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="EndDeviceAsset" sawsdl:modelReference="http://iec.ch/TC57/CIM-
generic#EndDeviceAsset">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="amrSystem" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.amrSystem"/>
<xs:element name="category" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.category"/>
<xs:element name="serialNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.serialNumber"/>
<xs:element name="ActivityRecords" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.ActivityRecords">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord">
<xs:sequence>
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.category"/>
<xs:element name="createdDateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.createdDateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.reason"/>
<xs:element name="severity" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.severity"/>
<xs:element name="status" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#ActivityRecord.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="0" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="DeviceFunctions" minOccurs="1" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#EndDeviceAsset.DeviceFunctions">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DeviceFunction">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#IdentifiedObject.mRID"/>
<xs:element name="application" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.application"/>
<xs:element name="category" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#Asset.category"/>
<xs:element name="configID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.configID"/>
<xs:element name="disabled" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#DeviceFunction.disabled"/>
```

```
<xs:element name="firmwareID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.firmwareID"/>
<xs:element name="hardwareID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.hardwareID"/>
<xs:element name="password" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.password"/>
<xs:element name="programID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/CIM-generic#AssetFunction.programID"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:schema>
```

H.12 SupplierConfig

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/SupplierConfig#" elementFormDefault="qualified"
attributeFormDefault="unqualified" xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/SupplierConfig#">
<xs:element name="SupplierConfig" type="m:SupplierConfig"/>
<xs:complexType name="SupplierConfig">
<xs:sequence>
<xs:element name="ServiceSupplier" type="m:ServiceSupplier" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ServiceSupplier"
sawsdl:modelReference="http://iec.ch/TC57/2009#ServiceSupplier">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="issuerIdentificationNumber" minOccurs="0" maxOccurs="1"
type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ServiceSupplier.issuerIdentificationNumber
"/>
<xs:element name="kind" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#ServiceSupplier.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#SupplierKind">
<xs:restriction base="xs:string">
<xs:enumeration value="other"/>
<xs:enumeration value="retailer"/>
<xs:enumeration value="utility"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="ElectronicAddresses" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#Organisation.ElectronicAddresses">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress">
<xs:sequence>
```

```
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="email" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.email"/>
<xs:element name="lan" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.lan"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="password" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.password"/>
<xs:element name="radio" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.radio"/>
<xs:element name="userID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.userID"/>
<xs:element name="web" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.web"/>
<xs:element name="status" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="postalAddress" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Organisation.postalAddress">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress">
<xs:sequence>
<xs:element name="poBox" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress.poBox"/>
<xs:element name="postalCode" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress.postalCode"/>
<xs:element name="streetDetail" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress.streetDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail">
<xs:sequence>
<xs:element name="addressGeneral" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.addressGeneral"/>
<xs:element name="buildingName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.buildingName"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.code"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.name"/>
<xs:element name="number" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.number"/>
<xs:element name="prefix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.prefix"/>
```

```
<xs:element name="suffix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.suffix"/>
<xs:element name="suiteNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.suiteNumber"/>
<xs:element name="type" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.type"/>
<xs:element name="withinTownLimits" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.withinTownLimits"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="townDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress.townDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail">
<xs:sequence>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.code"/>
<xs:element name="country" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.country"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.name"/>
<xs:element name="section" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.section"/>
<xs:element name="stateOrProvince" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.stateOrProvince"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="streetAddress" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Organisation.streetAddress">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#StreetAddress">
<xs:sequence>
<xs:element name="status" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetAddress.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="streetDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetAddress.streetDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail">
<xs:sequence>
<xs:element name="addressGeneral" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.addressGeneral"/>
<xs:element name="buildingName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.buildingName"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.code"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.name"/>
```

```
<xs:element name="number" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.number"/>
<xs:element name="prefix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.prefix"/>
<xs:element name="suffix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.suffix"/>
<xs:element name="suiteNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.suiteNumber"/>
<xs:element name="type" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.type"/>
<xs:element name="withinTownLimits" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.withinTownLimits"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="townDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetAddress.townDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail">
<xs:sequence>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.code"/>
<xs:element name="country" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.country"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.name"/>
<xs:element name="section" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.section"/>
<xs:element name="stateOrProvince" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.stateOrProvince"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TelephoneNumbers" minOccurs="1" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#Organisation.TelephoneNumbers">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="areaCode" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.areaCode"/>
<xs:element name="cityCode" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.cityCode"/>
<xs:element name="countryCode" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.countryCode"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="extension" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.extension"/>
<xs:element name="localNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.localNumber"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
```

```
</xs:schema>
```

H.13 CustomerConfig

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/CustomerConfig#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/CustomerConfig#">
<xs:element name="CustomerConfig" type="m:CustomerConfig"/>
<xs:complexType name="CustomerConfig">
<xs:sequence>
<xs:element name="Customer" type="m:Customer" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="Customer"
sawsdl:modelReference="http://iec.ch/TC57/2009#Customer">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="kind" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Customer.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#CustomerKind">
<xs:restriction base="xs:string">
<xs:enumeration value="commercialIndustrial"/>
<xs:enumeration value="energyServiceScheduler"/>
<xs:enumeration value="energyServiceSupplier"/>
<xs:enumeration value="internalUse"/>
<xs:enumeration value="other"/>
<xs:enumeration value="pumpingLoad"/>
<xs:enumeration value="residential"/>
<xs:enumeration value="residentialAndCommercial"/>
<xs:enumeration value="residentialAndStreetlight"/>
<xs:enumeration value="residentialFarmService"/>
<xs:enumeration value="residentialStreetlightOthers"/>
<xs:enumeration value="windMachine"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="pucNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Customer.pucNumber"/>
<xs:element name="specialNeed" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Customer.specialNeed"/>
<xs:element name="vip" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/2009#Customer.vip"/>
<xs:element name="ElectronicAddresses" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#Organisation.ElectronicAddresses">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
```

```
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="email" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.email"/>
<xs:element name="lan" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.lan"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="password" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.password"/>
<xs:element name="radio" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.radio"/>
<xs:element name="userID" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.userID"/>
<xs:element name="web" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.web"/>
<xs:element name="status" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#ElectronicAddress.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="postalAddress" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Organisation.postalAddress">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress">
<xs:sequence>
<xs:element name="poBox" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress.poBox"/>
<xs:element name="postalCode" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress.postalCode"/>
<xs:element name="streetDetail" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress.streetDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail">
<xs:sequence>
<xs:element name="addressGeneral" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.addressGeneral"/>
<xs:element name="buildingName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.buildingName"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.code"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.name"/>
<xs:element name="number" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.number"/>
<xs:element name="prefix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.prefix"/>
<xs:element name="suffix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.suffix"/>
```

```
<xs:element name="suiteNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.suiteNumber"/>
<xs:element name="type" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.type"/>
<xs:element name="withinTownLimits" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.withinTownLimits"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="townDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#PostalAddress.townDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail">
<xs:sequence>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.code"/>
<xs:element name="country" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.country"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.name"/>
<xs:element name="section" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.section"/>
<xs:element name="stateOrProvince" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.stateOrProvince"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="status" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Customer.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="streetAddress" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Organisation.streetAddress">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#StreetAddress">
<xs:sequence>
<xs:element name="status" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetAddress.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
```

```
</xs:element>
<xs:element name="streetDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetAddress.streetDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail">
<xs:sequence>
<xs:element name="addressGeneral" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.addressGeneral"/>
<xs:element name="buildingName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.buildingName"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.code"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.name"/>
<xs:element name="number" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.number"/>
<xs:element name="prefix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.prefix"/>
<xs:element name="suffix" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.suffix"/>
<xs:element name="suiteNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.suiteNumber"/>
<xs:element name="type" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.type"/>
<xs:element name="withinTownLimits" minOccurs="0" maxOccurs="1" type="xs:boolean"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetDetail.withinTownLimits"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="townDetail" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#StreetAddress.townDetail">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail">
<xs:sequence>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.code"/>
<xs:element name="country" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.country"/>
<xs:element name="name" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.name"/>
<xs:element name="section" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.section"/>
<xs:element name="stateOrProvince" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TownDetail.stateOrProvince"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TelephoneNumbers" minOccurs="1" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#Organisation.TelephoneNumbers">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="areaCode" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.areaCode"/>
<xs:element name="cityCode" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.cityCode"/>
<xs:element name="countryCode" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.countryCode"/>
```

```
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="extension" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.extension"/>
<xs:element name="localNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TelephoneNumber.localNumber"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:schema>
```

H.14 ServiceCategoryConfig

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/ServiceCategoryConfig#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/ServiceCategoryConfig#">
<xs:element name="ServiceCategoryConfig" type="m:ServiceCategoryConfig"/>
<xs:complexType name="ServiceCategoryConfig">
<xs:sequence>
<xs:element name="ServiceCategory" type="m:ServiceCategory" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ServiceCategory"
sawsdl:modelReference="http://iec.ch/TC57/2009#ServiceCategory">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="kind" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#ServiceCategory.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#ServiceKind">
<xs:restriction base="xs:string">
<xs:enumeration value="electricty"/>
<xs:enumeration value="gas"/>
<xs:enumeration value="heat"/>
<xs:enumeration value="internet"/>
<xs:enumeration value="other"/>
<xs:enumeration value="rates"/>
<xs:enumeration value="refuse"/>
<xs:enumeration value="sewerage"/>
<xs:enumeration value="time"/>
<xs:enumeration value="tvLicence"/>
<xs:enumeration value="water"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
```

```
</xs:sequence>
</xs:complexType>
</xs:schema>
```

H.15 PricingStructureConfig

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
targetNamespace="http://iec.ch/TC57/2009/PricingStructureConfig#"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://iec.ch/TC57/2009/PricingStructureConfig#">
<xs:element name="PricingStructureConfig" type="m:PricingStructureConfig"/>
<xs:complexType name="PricingStructureConfig">
<xs:sequence>
<xs:element name="PricingStructure" type="m:PricingStructure" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="PricingStructure"
sawsdl:modelReference="http://iec.ch/TC57/2009#PricingStructure">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.category"/>
<xs:element name="code" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#PricingStructure.code"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="revenueKind" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#PricingStructure.revenueKind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#RevenueKind">
<xs:restriction base="xs:string">
<xs:enumeration value="commercial"/>
<xs:enumeration value="industrial"/>
<xs:enumeration value="irrigation"/>
<xs:enumeration value="nonResidential"/>
<xs:enumeration value="other"/>
<xs:enumeration value="residential"/>
<xs:enumeration value="streetLight"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="revisionNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.revisionNumber"/>
<xs:element name="title" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.title"/>
<xs:element name="docStatus" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.docStatus">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
```

```
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ServiceCategory" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#PricingStructure.ServiceCategory">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#ServiceCategory">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="status" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Tariffs" minOccurs="1" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#PricingStructure.Tariffs">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Tariff">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.category"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="endDate" minOccurs="1" maxOccurs="1" type="m:AbsoluteDate"
sawsdl:modelReference="http://iec.ch/TC57/2009#Tariff.endDate"/>
<xs:element name="lastModifiedDateTime" minOccurs="1" maxOccurs="1"
type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.lastModifiedDateTime"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="revisionNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.revisionNumber"/>
<xs:element name="startDate" minOccurs="1" maxOccurs="1" type="m:AbsoluteDate"
sawsdl:modelReference="http://iec.ch/TC57/2009#Tariff.startDate"/>
<xs:element name="title" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.title"/>
```

```
<xs:element name="docStatus" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.docStatus">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="status" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TariffProfiles" minOccurs="1" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#Tariff.TariffProfiles">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#TariffProfile">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="category" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.category"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="lastModifiedDateTime" minOccurs="1" maxOccurs="1"
type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.lastModifiedDateTime"/>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="revisionNumber" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.revisionNumber"/>
<xs:element name="tariffCycle" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#TariffProfile.tariffCycle"/>
<xs:element name="title" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.title"/>
<xs:element name="ConsumptionTariffIntervals" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#TariffProfile.ConsumptionTariffIntervals">
<xs:complexType
sawsdl:modelReference="http://iec.ch/TC57/2009#ConsumptionTariffInterval">
<xs:sequence>
<xs:element name="sequenceNumber" minOccurs="1" maxOccurs="1" type="xs:integer"
sawsdl:modelReference="http://iec.ch/TC57/2009#ConsumptionTariffInterval.sequenceNumbe
r"/>
```

```
<xs:element name="startValue" minOccurs="1" maxOccurs="1" type="m:RealEnergy"
sawsdl:modelReference="http://iec.ch/TC57/2009#ConsumptionTariffInterval.startValue"/>
<xs:element name="Charges" minOccurs="1" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#ConsumptionTariffInterval.Charges">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Charge">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="kind" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Charge.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#ChargeKind">
<xs:restriction base="xs:string">
<xs:enumeration value="auxiliaryCharge"/>
<xs:enumeration value="consumptionCharge"/>
<xs:enumeration value="demandCharge"/>
<xs:enumeration value="other"/>
<xs:enumeration value="taxCharge"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="variablePortion" minOccurs="1" maxOccurs="1" type="m:PerCent"
sawsdl:modelReference="http://iec.ch/TC57/2009#Charge.variablePortion"/>
<xs:element name="ChildCharges" minOccurs="0" maxOccurs="unbounded"
sawsdl:modelReference="http://iec.ch/TC57/2009#Charge.ChildCharges">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Charge">
<xs:sequence>
<xs:element name="mRID" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.mRID"/>
<xs:element name="aliasName" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.aliasName"/>
<xs:element name="description" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.description"/>
<xs:element name="kind" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Charge.kind">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#ChargeKind">
<xs:restriction base="xs:string">
<xs:enumeration value="auxiliaryCharge"/>
<xs:enumeration value="consumptionCharge"/>
<xs:enumeration value="demandCharge"/>
<xs:enumeration value="other"/>
<xs:enumeration value="taxCharge"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="name" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#IdentifiedObject.name"/>
<xs:element name="variablePortion" minOccurs="1" maxOccurs="1" type="m:PerCent"
sawsdl:modelReference="http://iec.ch/TC57/2009#Charge.variablePortion"/>
<xs:element name="fixedPortion" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Charge.fixedPortion">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit">
<xs:sequence>
<xs:element name="energyUnit" minOccurs="0" maxOccurs="1" type="m:RealEnergy"
sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit.energyUnit"/>
<xs:element name="monetaryUnit" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit.monetaryUnit">
```

```
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#Currency">
<xs:restriction base="xs:string">
<xs:enumeration value="AUD"/>
<xs:enumeration value="CAD"/>
<xs:enumeration value="CHF"/>
<xs:enumeration value="CNY"/>
<xs:enumeration value="DKK"/>
<xs:enumeration value="EUR"/>
<xs:enumeration value="GBP"/>
<xs:enumeration value="INR"/>
<xs:enumeration value="JPY"/>
<xs:enumeration value="NOK"/>
<xs:enumeration value="RUR"/>
<xs:enumeration value="SEK"/>
<xs:enumeration value="USD"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="multiplier" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit.multiplier">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#UnitMultiplier">
<xs:restriction base="xs:string">
<xs:enumeration value="G"/>
<xs:enumeration value="M"/>
<xs:enumeration value="T"/>
<xs:enumeration value="c"/>
<xs:enumeration value="d"/>
<xs:enumeration value="k"/>
<xs:enumeration value="m"/>
<xs:enumeration value="micro"/>
<xs:enumeration value="n"/>
<xs:enumeration value="none"/>
<xs:enumeration value="p"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="fixedPortion" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Charge.fixedPortion">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit">
<xs:sequence>
<xs:element name="energyUnit" minOccurs="0" maxOccurs="1" type="m:RealEnergy"
sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit.energyUnit"/>
<xs:element name="monetaryUnit" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit.monetaryUnit">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#Currency">
<xs:restriction base="xs:string">
<xs:enumeration value="AUD"/>
<xs:enumeration value="CAD"/>
<xs:enumeration value="CHF"/>
<xs:enumeration value="CNY"/>
<xs:enumeration value="DKK"/>
<xs:enumeration value="EUR"/>
<xs:enumeration value="GBP"/>
```

```
<xs:enumeration value="INR"/>
<xs:enumeration value="JPY"/>
<xs:enumeration value="NOK"/>
<xs:enumeration value="RUR"/>
<xs:enumeration value="SEK"/>
<xs:enumeration value="USD"/>
<xs:enumeration value="other"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="multiplier" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit.multiplier">
<xs:simpleType sawsdl:modelReference="http://iec.ch/TC57/2009#UnitMultiplier">
<xs:restriction base="xs:string">
<xs:enumeration value="G"/>
<xs:enumeration value="M"/>
<xs:enumeration value="T"/>
<xs:enumeration value="c"/>
<xs:enumeration value="d"/>
<xs:enumeration value="k"/>
<xs:enumeration value="m"/>
<xs:enumeration value="micro"/>
<xs:enumeration value="n"/>
<xs:enumeration value="none"/>
<xs:enumeration value="p"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:float"
sawsdl:modelReference="http://iec.ch/TC57/2009#AccountingUnit.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="docStatus" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.docStatus">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
<xs:element name="remark" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
<xs:element name="value" minOccurs="1" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="status" minOccurs="1" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Document.status">
<xs:complexType sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
<xs:sequence>
<xs:element name="dateTime" minOccurs="1" maxOccurs="1" type="xs:dateTime"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
<xs:element name="reason" minOccurs="0" maxOccurs="1" type="xs:string"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
```