Honeywell

THE POWER OF CONNECTED

KSMW-PA2502 - Companion Standard

Main Document

Revision 1.1

18.09.2018

Copyright

Disclaimer

Index

1. Introduction

1.1. Scope

This companion standard is a functional description of the 1 and 3-phase meters for the smart metering program of Kooperation Smart Meter West [KSMW].
The companion standard will define the external interfaces including the communication profiles of the smart meters as well as the used object model and necessary program specific functionalities.
It must be noted, that the companion standard is not a substitution of the metering specification published by KSMW. It has to be seen as further definition to ensure interoperability between different metering devices within the here used smart metering infrastructure.

1.2. Normative references

This companion standard is based on the following document:

- EN 62056 – 5 & EN 62056-6 [A]
- DLMS Blue Book version 1000-1 Ed. 12.2 [B]
- DLMS Green Book version 1000-2 Ed. 8.3 [C]
- IDIS Standard Package 2, Edition 2.0, 03-06-2014 [D]
- Published Specification “05_PA2502_Beschreibung_Anforderungen_IMS_V2” from 17.11.2016 [E]

The above mentioned documents are valid unless explicitly mentioned.
Mentioning DLMS/COSEM in this document refers to the above mentioned versions of the Green and Blue Book.

1.3. Document list

This companion standard references to the following documents, which are delivered together with the companion standard:

- KSMW-PA2502 Companion Standard Object Model rev 1.1.xls [1]
- KSMW-PA2502 Companion Standard G3-PLC Implementation Guide rev 1.1.pdf [2]
- KSMW-PA2502 Companion Standard P2P WAN Implementation Guide rev 2.5.pdf [3]
- KSMW-PA2502 Companion Standard Display Implementation Guide rev 1.1.pdf [4]
- KSMW-PA2502 Companion Standard M-Bus Implementation Guide rev 1.1.pdf [5]

1.4. Abbreviations

Abbreviation	Explanation
AA	Application Association
AARE	Application Association Response
AARQ	Application Association ReQuest

ACSE	Association Control Service Element
APDU	Application Protocol Data Unit
ASE	Application Service Element
A-XDR	Adapted Extended Data Representation
CII	Consumer Information Interface
CIP	Consumer Information Push
class_id	Interface class identification code
COSEM	Companion Specification for Energy Metering
COSEM object	An instance of a COSEM interface class
DC	Data Concentrator used for PLC communication
DLMS	Device Language Message Specification
ERP	Enterprise Resource Planning
FC	Frame Counter
G3	G3 PLC supporting IPv6
GCM	Galois/Counter Mode, an algorithm for authenticated encryption with associated data
UTC	Coordinated Universal Time
CSD	Circuit Switched Data
HDLC	High-level Data Link Control
HES	Head End System similar to MDC
HLS	COSEM High Level Security
IC	COSEM Interface Class
IEC	International Electrotechnical Commission
LLC	Logical Link Control (Sublayer)
LLS	COSEM Low Level Security
LN	COSEM Logical Name
MDC	Meter Data Collect similar to HES
MDM	Meter Data Management
OBIS	Object Identification System
PDU	Protocol Data Unit
PUSH	the data is pushed by the meter to the HES using the Data Notification service
SAP	Service Access Point

SMS	Short Message Service
L_SAP	Link layer Service Access Point

Table 1: List of used abbreviations

1.5. Revisions History

Version	Revisions	Date	Author
0.0	Initial Draft Version	16.11.2017	R. Thor
0.1	1st Draft Release	20.12.2017	R. Thor
0.2	Update according: KSMW PA2502 Companion Standard Review List Rev 0.2.xlsx	29.01.2018	R. Thor
0.3	Update according: KSMW PA2502 Companion Standard Review List Rev 0.3.xlsx	22.02.2018	R. Thor
0.4	Update according: KSMW PA2502 Companion Standard Review List Rev 0.4.xlsx	06.03.2018	R. Thor
0.5	Update according: KSMW PA2502 Companion Standard Review List Rev 0.5.xlsx	19.03.2018	R. Thor
0.6	Update according: KSMW PA2502 Companion Standard Review List Rev 0.6.xlsx	06.04.2018	R. Thor
1.0	Update according: KSMW PA2502 Companion Standard Review List Rev 1.0.xlsx	03.07.2018	R. Thor
1.1	Update according: KSMW PA2502 Companion Standard Review List Rev 1.1.xlsx	18.09.2018	R. Thor

Table 2: Revisions History

2. System Architecture

The entire smart metering program of KSMW is following the overall architecture as shown in figure 1. This companion standard specifies mainly functionalities inside the electricity meter (E-Meter) and the communication between the electricity meter and the Head End System (HES)

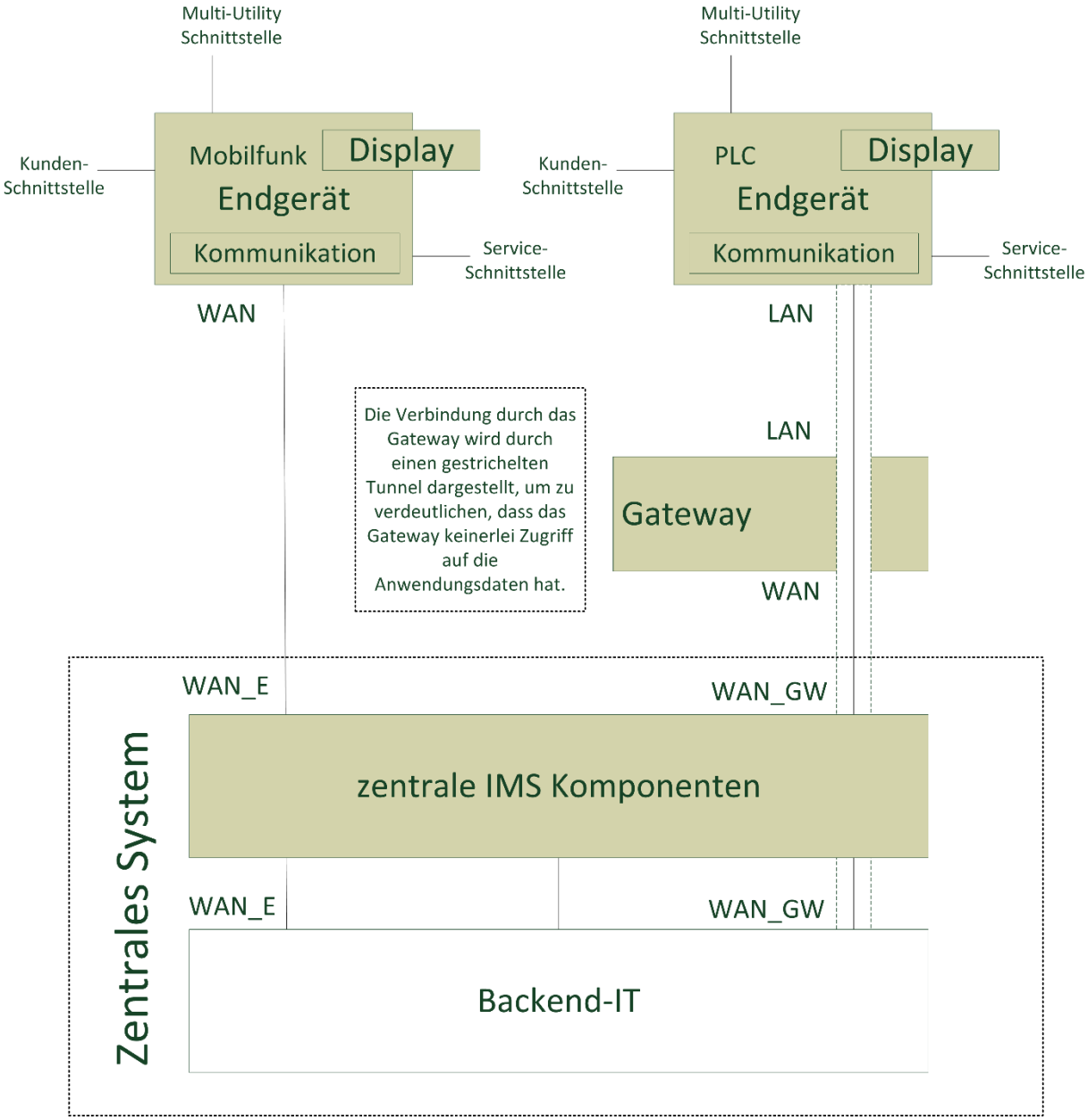


Figure 1: Architecture Overview

An overview of the used interfaces for the electricity meter are shown in Table 1.

Electricity meter

Interface	Description	Technology
WZ - Service Schnittstelle	Service Interface	Infrared optical interface
H1 - Kunden Schnittstelle	Consumer Interface	Wired M-Bus
M2 - Multi-Utility Schnittstelle	Multi-Utility interface	Wired / Wireless M-Bus
LAN	Local Area Network interface	G3-PLC
WAN	Wide Area Network interface	Cellular

Table 3: Used Interfaces

2.1. WZ - Service Interface

The WZ interface is specified as an optical infrared interface conform to IEC 62056-21.
The baud rate must be at least 9600 baud. The main intention of the WZ is to act as service interface.
It needs to be mentioned that a meter read conform to IEC 62056-21 is not allowed, all read-out procedures need to be compliant to DLMS / COSEM.

2.2. H1 - Consumer Interface

The H1 interface is specified as a wired M-Bus interface conform to EN 13757-2 with a fixed baud rate is at 2400 baud.

The physical interface is defined as RJ12 Modular Jack 6P6C connector with the following pinout!!

1 - NC
2 - NC
3 - MBUS1 (+)
4 - MBUS2 (-)
5 - NC
6 - NC

Figure 2: RJ12 connector (Tab Down) front view

The H1 interface is defined as a wired M-Bus master and must support 4 Mbus loads as a minimum (=> total of 6mA on 32V)

This interface allows only one-way communication by pushing data to an attached device. It is not allowed to receiving data via the H1 interface.

Further, realizing the H1 via the optical port is not an allowed option.

2.3. M2 - Multi Utility Interface

The Multi Utility interface uses the M-Bus technology to connect additional submeters like Gas, Water or Heat-meters to the E-meter.

The E-meter supports either wired or wireless M-Bus, or wired and wireless M-Bus.

A detailed description of the M-Bus interface is available in [5].

2.4. LAN/WAN – HES Interface

There are 2 possible interfaces specified for the communication between the electricity meter and the Head End System (HES)

LAN interface:

The LAN interface is specified as Powerline Communication interface by using OFDM method G3 for PLC transmission based on the ITU-T G.9901 (2017) and ITU-T G.9903 (2017) in the CENELEC A and FCC band.

A detailed description of the G3-PLC interface is available in [2].

WAN interface:

A detailed description of the P2P WAN interface is available in [3].

2.5. User Interface (Display)

The meter provides a display and push buttons as the local user interface.
Using the push button, the user can step through a menu structure on the display to check the consumption data and further information.

The meter support 1 or 2 push buttons

- [A]–Button (support mandatory):
  Used for stepping through the menu structure and display items
- [R]–Button (support optional):
  Used for triggering a manual billing profile capture (Demand Reset)

A detailed description of the display handling is available in [4].

3. DLMS/COSEM Application Layer

3.1. DLMS Security Suites

This specification mandates the use of security suite 1 initially (which adds support for digital signatures key agreement, and an authentication mechanism based on ECDSA using ECC P-256 asymmetric cryptography, all the while also supporting the suite 0 mechanisms for authenticated encryption and key wrapping)

However, it must be possible to support the Security Suite 2 (which is similar to suite 1 with respect to supported services, but mandates the use of stronger keys) in the future and this includes updating of all installed devices by configuration change or remote FW upgrade. It's the manufacturers' responsibility to ensure enough available resources on the delivered products to comply with this requirement.

3.2. Logical Devices

In DLMS/COSEM, metering equipment is modelled in physical and logical devices.
- The actual device is the physical device.
- The physical device can contain multiple logical devices.

For this companion standard it is decided that there will be only 1 logical device (the management logical device).

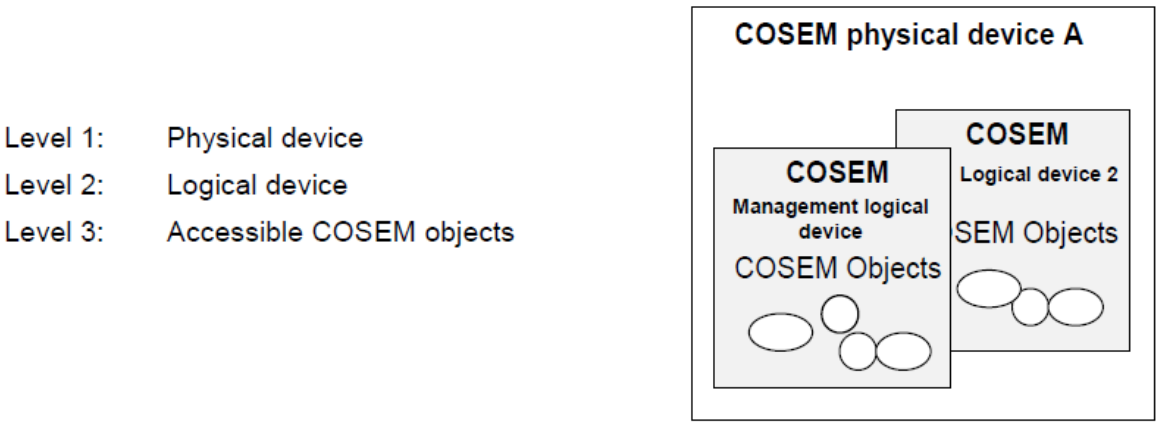


Figure 3: COSEM Device Management

The following object provides the necessary information about the available logical device:

Object / Attribute Name	Class	Ver.	OBIS code
SAP Assignment	17	0	0-0:41.0.0.255

Table 4: Logical Device Objects

3.3. Associations and Services

3.3.1. Supported Services

Maximum PDU size
The following settings for the maximum PDU size for transmit and receive must be respected in the server for all interfaces:
- Max Receive PDU Size = 1224 bytes
- Max Transmit PDU Size = 1224 bytes

Conformance Block
The Conformance Block defines the minimum set of supported application layer services:
- General-protection (1)
- General-block-transfer (2)
- Attribute0-supported-with-get (10)
- Block-transfer-with-get (11)
- Block-transfer-with-set (12)
- Multiple-references (14)
- Data-Notification (16)
- Access (17)
- Get (19)
- Set (20)
- Selective-access (21)
- Action (23)

⇨ For multiple references services in GET request service and ACCESS request service, a minimum of 16 references must be supported.
⇨ For multiple references services in the Set and Action service, the minimum is limited to one.

Regardless of the limitations above, the GET or the ACCESS Request apdu must not be larger than the max apdu size.

If the data-notification service needs to be protected and/or needs block-transfer, then the general-glo-ciphering service and/or the general-block-transfer service are used for this purpose.

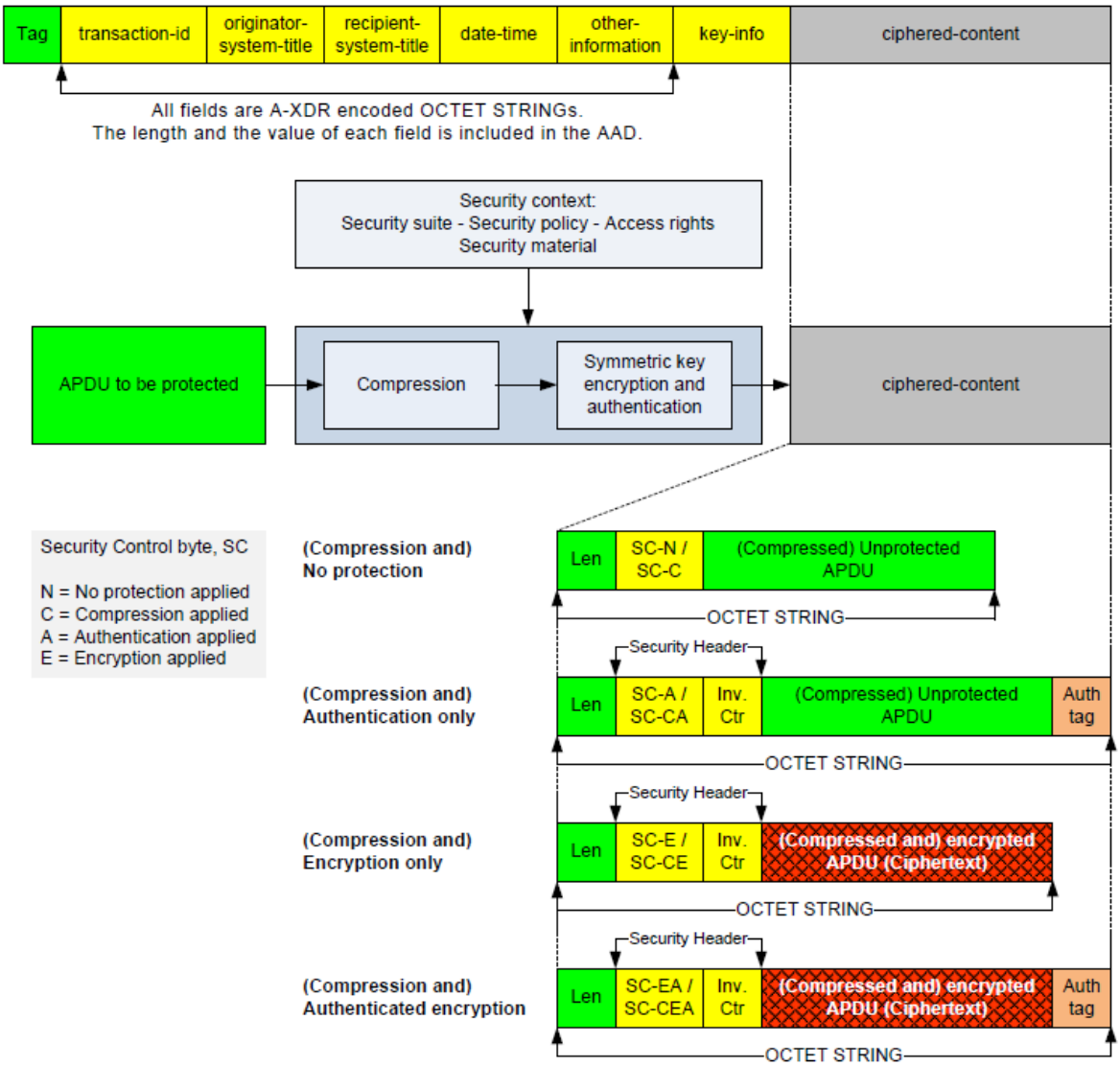
For the service specific GET, SET, ACTION and ACCESS services the meter must support the global as well as the dedicated protection services (glo-ciphering-Get/Set/Action and ded-ciphering-Get/Set/Action).

For the general variants, general-glo and general-ded ciphering must be supported, as well as general-ciphering using identified and wrapped keys (general-ciphering with agreed keys are not in scope for this specification as their use is cumbersome since every request-response exchange requires a new agreement process).
Additionally, the general-signing service must also be supported.

If both ciphering and signing is required, then the digital signature is applied first.

For the GET and SET services the meter must support the service specific block-transfer mode.
Additionally, the general-block-transfer service must be supported.
The combination of several block transfer mechanisms on the same apdu is not supported.
It must be possible use the GET and SET services for the largest object of the Data Model without multiple GET and SET request/response operations. In case the data becomes larger than the PDU size, the general-block-transfer shall be used.

3.3.1.1. General-Ciphering APDU structure



Definition of the ‘additional fields’

⇨ Transaction-id: => used

Identifies the transaction between two parties; it is generated by the client and included in the request APDU. The server shall use the same transaction-id in the response APDU.

⇨ Originator-system-title: => used
Unique identifier for identifying the party that applied the protection.

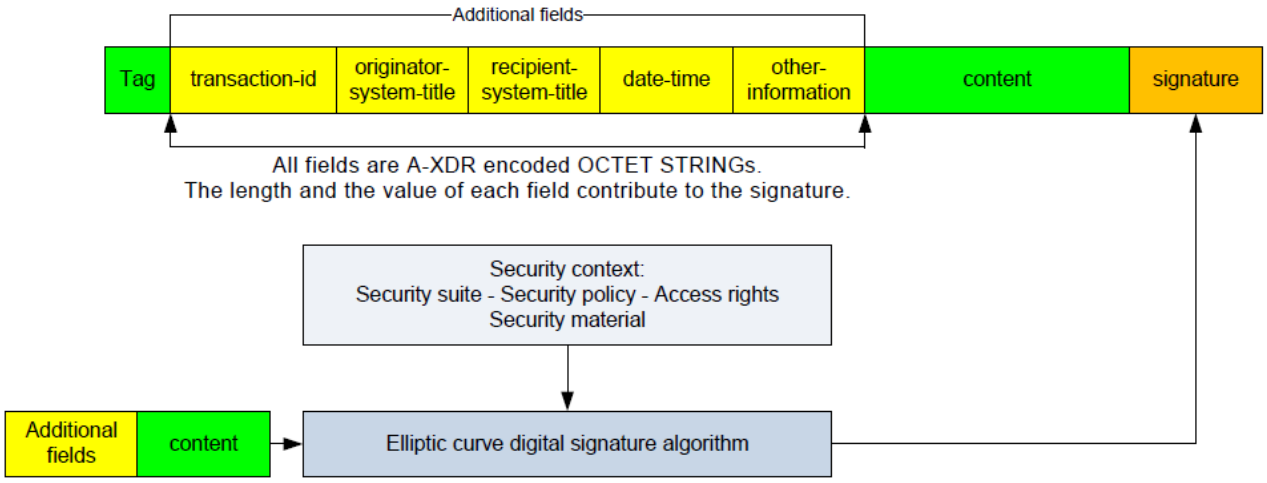
⇨ Recipient-system-title: => used
Unique identifier for identifying the party that shall verify the protection.

⇨ Date-time: => optional
Date and Time of the invocation of the .request / .response service primitive; If the client includes a date-time in the request APDU, the server shall include a date-time in the response APDU.

⇨ Other-information: => not used
Holds additional information concerning the protection.

⇨ Key-info: => used
The Key_Info parameter carries information on the symmetric key that has been used by the originator / is to be used by the recipient.
The possible options are
- Identified_Key: supported
- Wrapped_Key: supported
- Agreed_Key: not supported

3.3.1.2. General-Signing APDU structure



Definition of the ‘additional fields’

⇨ Transaction-id: => used
Identifies the transaction between two parties; it is generated by the client and included in the request APDU. The server shall use the same transaction-id in the response APDU.

⇨ Originator-system-title: => used

Unique identifier for identifying the party that applied the protection.

⇨ Recipient-system-title: => used
Unique identifier for identifying the party that shall verify the protection.

⇨ Date-time: => optional
Date and Time of the invocation of the .request / .response service primitive. It the client includes a date-time in the request APDU, the server will also include a date-time in the response APDU.

⇨ Other-information: => not used
Holds additional information concerning the protection.

3.3.2. Invoke-Id-and-Priority

The GET, SET and ACTION services are using the Invoke-Id-And-Priority byte (type Unsigned8).

Invoke-Id-And-Priority:

Bit 0-3 (invoke-id-zero …)	unsigned 4 bit (LSB bit 3) number incremented with each invocation of the Data-Notification service
Bit 4-5 (reserved)	is set to 0,
Bit 6 (service_class)	must be set by the client in order to get an answer from the meter. The meter only answers if this Bit is set in the request.
Bit 7 (priority)	must be set by the client in case a higher prioritised request response is required.

The access and the data-notification services are using the Long-Invoke-Id-And-Priority (type Unsigned32)

Long-Invoke-Id-And-Priority:

Bit 0-23 (invoke-id-zero …)	unsigned 24 bit (LSB bit 23) number incremented with each invocation of the Data-Notification service
Bit 28 (self-descriptive)	is set to 0,
Bit 29 (processing-option)	is set to 0,
Bit 30 (service_class)	is set to 0,
Bit 31 (priority)	is set to 0,

The meter must return the Invoke-Id_And-Priority or Long-Invoke-Id-And-Priority as received from the client.

3.3.3. AARQ and RLRQ pdus

InitiateRequest field
An AARQ carrying non-ciphered context information (context_id different from Logical_Name_Referencing_With_Ciphering) must be rejected by the server with an "error action" in case security policy >0

Context name Security policy	Logical_Name_Referencing_No_Ciphering =0	Logical_Name_Referencing_With_Ciphering =0	Logical_Name_Referencing_No_Ciphering >0	Logical_Name_Referencing_With_Ciphering >0
RLRQ	No InitiateRequest Unciphered InitiateRequest	Ciphered InitiateRequest	Not possible	Ciphered InitiateRequest
AARQ	Unciphered InitiateRequest	Ciphered InitiateRequest	Not possible	Ciphered InitiateRequest

Table 5: InitiateRequest Field

Calling_AP_title
The AARQ request used when opening the association shall carry the client system title (SysT-C) in the calling-ap-title field.
The AARE response from the meter shall correspondingly carry the server system title (SysT-S) in the responding-ap-title field.
⇨ This is also required for the Public Client in case of reading the frame counter value objects (refer to 11 Appendix 1: Frame Counter Readout for more information)

Calling_AE_qualifier
The AARQ request used when opening the association may carry the clients' public key certificate for the clients' digital signature key calling-ae-qualifier field in case the certificate has not been previously imported.
It is used in combination with HLS authentication mechanism 7 (ECDSA). Due to the fact that using this feature would complicate the setup of the PKI structure (it would require that every defined client (installation, management, …) corresponds to its own sub-CA), this field will be ignored by the server. This means that the server will not use the public key carried in this field to verify the f(StoC), instead requiring that the public key of the calling party be known and trusted by the security setup corresponding to the client being associated with.

However, if the client includes its signing certificate in the AARQ (calling-AE-qualifier), the server shall include its own certificate in the called-AE-qualifier of the AARE.

User Information field

If in the “Association Release Request “service (sent by the client) the optional parameter “user information” is present, then server must answer with the “Association Release Response” service with the parameter “user information” also present.

If in the RLRQ the parameter “user information” is not present, then it must also be not present in the RLRE.

3.3.4. Association Behavior

Lost Associations
If the server responds to any Get or Set or Action or Access request from the client with an ”ExceptionResponse” due to a lost association then the client has to send an AARQ again (has to establish the association again)

Associations on different communication ports
The following rules apply:
- On WZ, only one association can be opened at a time.
- On LAN, several associations may be opened at the same time.
- At different communication ports, several associations (with the same client or with different clients) may be opened at the same time.
- If a client wants to use several communication ports at the same time it must open an association at each communication port separately.
- Synchronization of Internal memory access must be handled by the manufacturer.

3.4. Application Layer Error Handling

The device follows the definitions of the IDIS package 2 specification in relation to the DLMS/COSEM application layer error handling.
Please refer to the following chapters in the IDIS package 2 specification [D]:
- ⇨ 8.2.3 Error handling in the application layer

3.5. Application Layer Security

The server may support several security contexts. Each available client requires a dedicated security context, which is configured by its security setup object.

Since there is only one logical device the meter’s server system title is always the same for all associations.

The Management and the Maintenance clients are the highest authority within the meter and are responsible for the security setting of the other clients.
All settings to the security context of the existing clients go through the ‘Security Setup objects that are assigned to these clients.

3.5.1. Security Policy

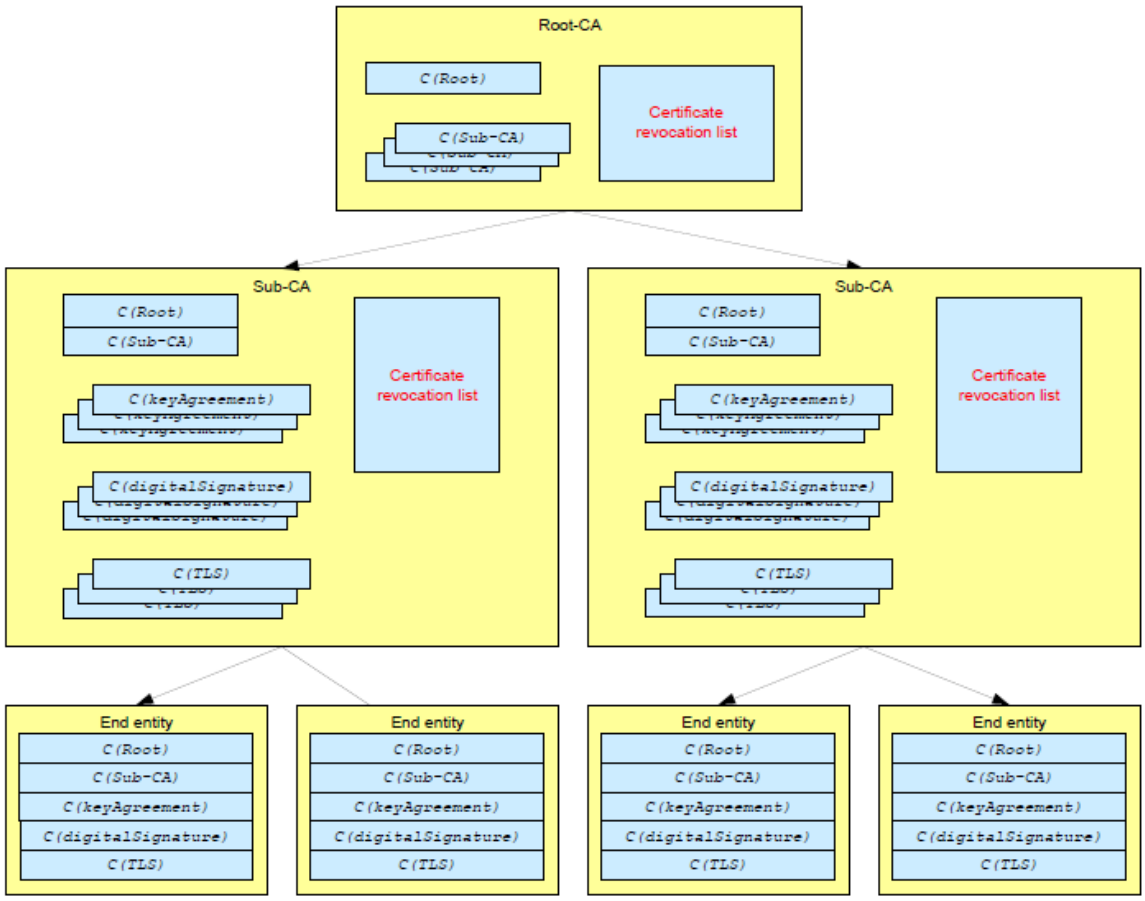
DLMS/COSEM allows different security policy levels including authentication and ciphering. The security policy can be either unused (means that no cryptographic protection is required) or any combination of the following options:

- unused
- authenticated request,
- encrypted request,
- digitally signed request (only applicable for security suite 1and 2 support),
- authenticated response,
- encrypted response
- digitally signed response (only applicable for security suite 1 and 2 support)

3.5.2. Certificate handling

Using the features of the DLMS security suites 1 and 2 like the generation of digital signatures and key agreement requires the support of Public Key certificates.

In general, the following PKI infrastructure is considered in order to create and manage public key certificates for facilitating the use of public key cryptography.



This model proposes different types of certificates:

Root Certification Authority

The Root-CA provides the trust anchor of the PKI. It issues certificates for Sub-CAs.

- ⇨ C(Root) – The Certificate of the Root-CA is self-signed with the Root-CA private key

SubordinateCertification Authority Certificate

A Sub-CA is an organization that issues certificates for end entities.

- ⇨ C(Sub-CA) - The Certificate of the Sub-CA is signed with the Root-CA private key

End Entity

In this context, each End entity can be seen as DLMS/COSEM clients, DLMS/COSEM servers and third parties.

DLMS defines the following certificates that are signed by the Sub-CA private key

- ⇨ C(keyAgreement) - Static Key Agreement Certificate
- ⇨ C(digitalSignature) - Digital Signature Certificate
- ⇨ C(TLS) - TLS-Certificate

Remark:

- ⇨ Static Key Agreement Certificate is not required for this implementation as it's only used for the Static Unified Model C(0e, 2s, ECC CDH) ECDH key agreement algorithm. The Static Unified Model algorithm is not used in this implementation.
- ⇨ TLS-Certificate is not required for this implementation as TLS is not supported

The following key pairs need to be present on the meter (we only consider suite 1, so P256 ECC key pairs):

- ⇨ Digital signature key pair (ECC P-256, private key and associated certificate)

The digital signature key pair must be generated by the meter itself (the private key must never leave the device).
Following the generation of the key pair, the meter should generate a CSR, which can then be extracted and signed, resulting in the corresponding certificate that can then be imported.

The following trusted certificates need supporting by the meter at a minimum:

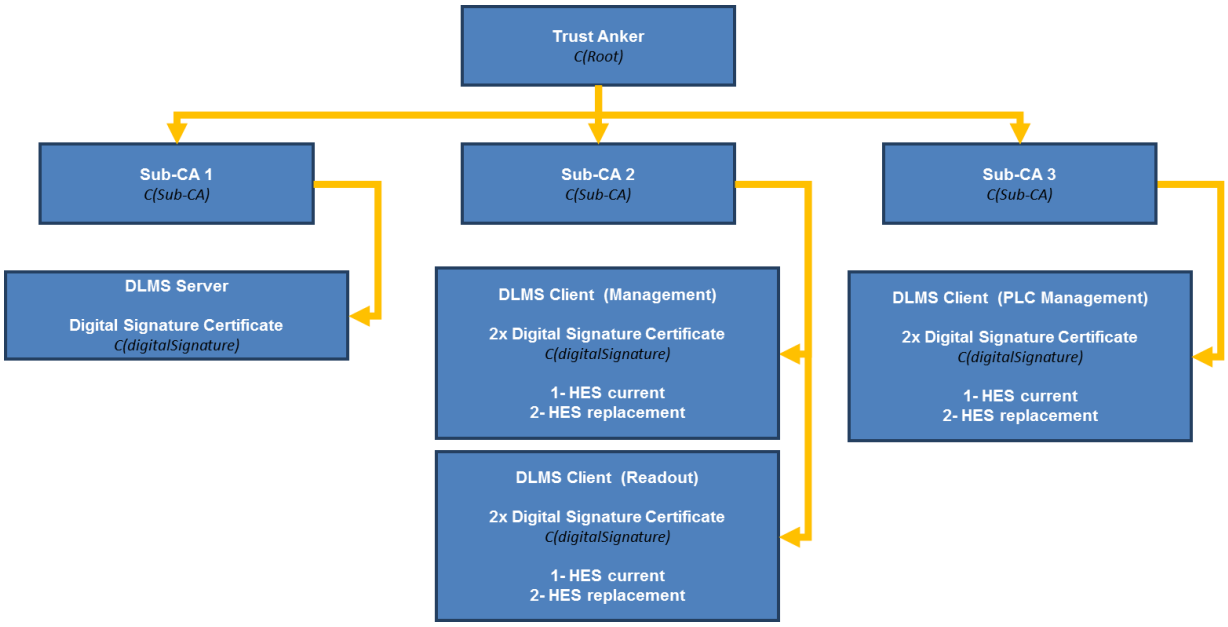
- ⇨ Trust Anchor:
  - Root CA Certificate
- ⇨ For the Sub-CA:
  - Sub CA Certificate 1 (for DLMS Server certificates)
  - Sub CA Certificate 2 (for DLMS Client certificates – Management and Readout clients)
  - Sub CA Certificate 3 (for DLMS Client certificates – PLC Management client)
- ⇨ For the DLMS server:
  - Meter certificate (for meter digital signature key pair)
- ⇨ For each DLMS client that uses Public key based features of Security Suite 1 and 2:
  - Up to 2 HES certificates (Management Client security setup)
  - Up to 2 HES certificates (Readout Client security setup)
  - Up to 2 HES certificates (PLC Management Client security setup)

All certificates shall have the structure specified for X.509 version 3 certificates as defined for the usage within DLMS/COSEM (please refer to chapter 9.2.6.4 Certificate and certificate extension profile in the Green Book [C])..
Only the mandatory fields of the certificate and certificate extensions will be used.
Certificates using optional fields may be rejected.
Please find some example certificates in Appendix 2: Certificate Examples

A chain of trust is established by validating each certificate from the end entity up to the root certificate.

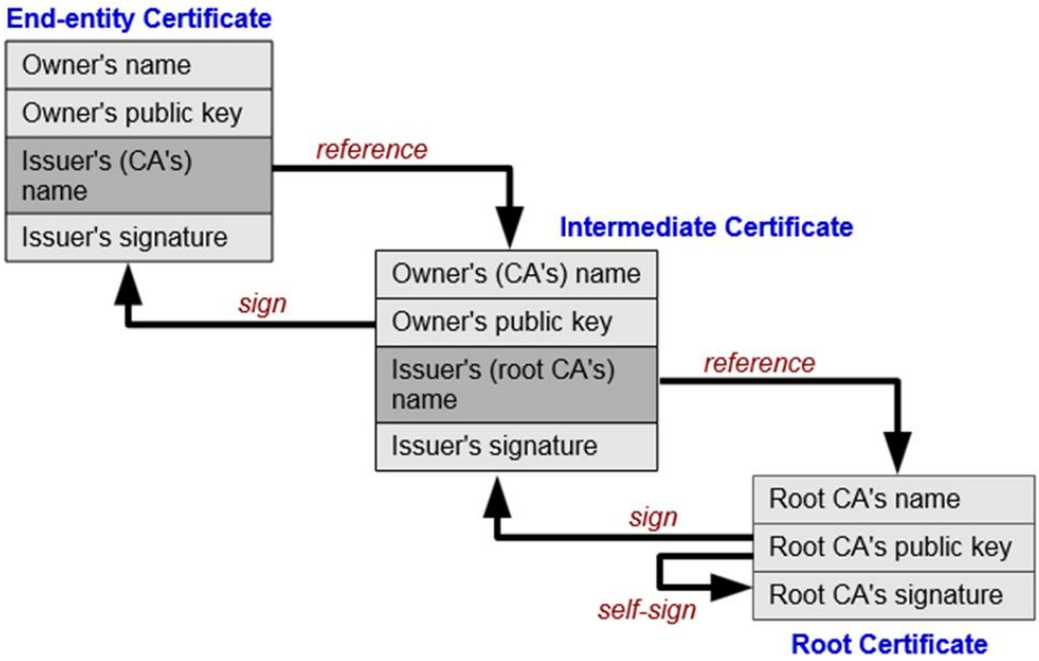
In order to use the clients all required certificates have to be in place. This means, initial provisioning of the certificates must happen during manufacturing.

The initial provisioning of the root-certificate is out of scope of the DLMS standards and in the responsibility of the meter manufacturer.



3.5.2.1. Updating Certificates

When updating any certificates within this structure, the chain of trust must be respected and validated.
To limit the impact on the embedded devices, the validation of the trust chain is only required at the time of importing a new certificate.
Further to this, the validation of the chain is always running from newly imported certificate down to the root certificate.



- Importing a new Root certificate
    - Root certificate is validated by its own Root Public Key
- Importing a new Sub-CA certificate
    - New Sub-CA certificate is validated by the Root Public Key
    - Root certificate is validated by its own Root Public Key
- Importing a new DigitalSignature certificate
    - New DigitalSignature certificate is validated by the Sub-CA Public Key
    - Sub-CA certificate is validated by the Root Public Key
    - Root certificate is validated by its own Root Public Key

The DigitalSignature certificates are required for the establishment of a communication to the device and mustn't be automatically invalidated when changing anything in the trust chain.

Following this principle allows renewing the complete chain of trust beginning with the Root certificates, over the Sub-CA certificates up to the DigitalSignature certificates without losing access to the device.

- ⇨ Any issues during the exchange of the Root certificate will not end up in losing access to the device
- ⇨ Any issues during the exchange of the Sub-CA certificate will not end up in losing access to the device

The import of a Root certificate after production is expected to follow the secure FW-update process as described in 0

Remote Firmware Upgrade. Updating the Root certificate is a replacement of the old certificate with the new one.

The import of a Sub-CA uses the 'import_certificate' method of the associated security_setup object. In case the number of supported Sub-CA certificates is at its maximum, the import is rejected. The method 'remove_certificate' allows the removal of a certificate before a new import might be accepted.

The exchange of the Client DigitalSignature certificates needs special consideration as these are mandatory for establishing any communication. Removing or invalidating this certificate will end up in losing any way to access the meter via the impacted client again.
Due to the fact that removing an old certificate before importing a new one creates a potential risk in the exchange procedure, the meter will actually support at least 2 possible DigitalSignature certificates per client.
This way, a new certificate can be imported before removing the one currently in use.

The Server DigitalSignature certificate cannot be removed from the server. When an update of this certificate is needed, a new key pair is generated, a new certificate request is generated and a new certificate is imported. The current key pair remains valid and active until the new certificate is imported.
The time between the generation of a new key pair and the import of the new certificate is limited to 24h. If no new certificate is imported within this timeframe, the new key pair must be dismissed.

3.5.3. Key handling

Depending on the security policy set and the individual access right definition of the attributes and methods, the following keys will be used according to their security context:

- ⇨ Global unicast encryption key
- ⇨ Global authentication key
- ⇨ Dedicated unicast encryption key

The following rules concerning the keys apply:

- At a given point of time there exists one specific set of keys (dedicated, global) per security context.
- There exists always one unique master key per device.
- The support of Dedicated Keys is mandatory. The meters must accept RLRQ and AARQ with or without a Dedicated Key.
- The lifetime of the Global Keys of each security context is limited by the range of the associated Frame Counters. The global key must be changed explicitly by the client.
- Dedicated keys are valid during the lifetime of an association; i.e. the dedicated key is generated and taken in use with the opening of the association. The key is destroyed automatically by the server upon closing of the association.
- If a dedicated service is requested by the client but the dedicated key is not known by the meter, then the meter returns an error: exception response(service-not-allowed, operation-not-possible)

There are 2 way of changing the keys for a client. Which way to use, depends on the usage and the capabilities of the individual client.
- Using the key_transfer method of the client security setup object
- Using the key_agreement method of the client security setup object

The master key can be changed using the object "Current Security Setup".
- Maintenance Client allows changing the master key using the key_transfer method
- Management Client allows changing the master key using the key_agreement method

Possible responses from the meter when changing a key using either the "key_transfer" or "key_agreement" method:
- If the "new" key is accepted, then the meter sends Action Response (same invoke_id and priority as the request):
  SUCCESS ciphered with "currently used" key. From this point on, meter uses the "new" key (replacing the "currently used" key with "new" key) and resets FC.
- If the type of the data in the Action Request is not correct then the meter answers with Action Response (same invoke_id and priority as the request):
  Data_Access_Error=type-unmatched.
- If the content of the data in the Action Request is not correct (e.q. new key wrapped with a wrong master key) then the meter answers with Action Response (same invoke_id and priority as the request):
  Data_Access_Error=other-reason.
- If the meter cannot decrypt the APDU (request encrypted with invalid key)
  Response(state-error=service-not-allowed, service-error=operation-not-possible).

3.5.3.1. Key exchange via key transfer

The AES key wrap algorithm is used to exchange keys in the meter via the 'key_transfer" method. This algorithm is using the master key for the wrapping algorithm.
There is only one master key in the meter which is used to exchange the keys.

The Security Setup class allows changing of the following keys using the key transfer method:
- ⇨ Global unicast encryption key
- ⇨ Global authentication key
- ⇨ Master key

3.5.3.2. Key exchange via key agreement

The ECDH key agreement algorithm with the Ephemeral Unified Model C(2e, 0s, ECC CDH) is used to exchange keys in the meter via the 'key_agreement" method. This algorithm is using ephemeral key pairs which are signed using the digital signature keys of both parties (meter and HES most commonly).

The Security Setup class allows changing of the following keys using the key agreement method:

⇨ Global unicast encryption key
⇨ Global authentication key
⇨ Master key

3.5.4. Frame Counter Handling

Depending on the security policy set and the individual access right definition of the attributes and methods, the following keys will be used according to their security context:

⇨ Global unicast encryption key
⇨ Global authentication key
⇨ Dedicated unicast encryption key

Each meter must store the following frame counters per security context:

Key	Frame counter Tx	Frame counter Rx	Storage
Global unicast encryption key	FCTxu	FCRxu	Non volatile Valid until key change
Global authentication key	na	na	na
Dedicated unicast encryption key	FCTxu-d	FCRxu-d	Volatile Valid for current Association only

Table 6: Frame counter per security index

The following rules concerning the frame counters apply:

- The transmit frame counter is incremented for every message sent.
- The server shall process the frame counter in the received message and validate it according to the following rule
    - A message is rejected if the frame counter in the received message is smaller or equal to the frame counter in the previously received message.

- FCs used with global keys are reset (to 0) when a new global key is established
- When operating with global keys then the client re-synchronizes its FCs by either reading the FCs from the meters (via public client, only available for the WZ clients) or exchanging the global key (generating a request with an estimated higher FC, close to the max FC).

- FCs used with dedicated keys are reset (to 0) when a new association is established (new ded key generated by client – transmitted with InitiateRequest, encrypted with global key)
- Frame Counters used with dedicated keys are independent of the FCs used with global keys.
- Frame Counters used with dedicated keys are handled internally in the meter (no access via COSEM object provided)
- When operating with dedicated keys the client re-synchronizes its FCs by first closing the current association (using global unicast keys) and after re-opening the association (using global unicast keys) by changing the dedicated keys (the FCs are automatically reset).

- When the maximum value of the FC has been reached, any following invocation of the corresponding encryption function shall return an error and the FC shall not be

incremented. It the responsibility of the HES to exchange the keys before the FC reaches its maximum.

The following rules concerning the frame counters for key changes apply:

- If only the Glo-authentication key is changed then none of the FCs is reset.
- If the Glo-unicast key is changed then the FC of the Glo-unicast key is reset automatically to 0.
- If the master key is changed then none of the FCs is reset.

The following rules concerning the frame counters for the HLS5 association opening apply:

⇨ pass 1 and 2: secure initiate request (++**global FC**),
⇨ pass 3 and 4: wrap in **global-action-request** (++**global FC** (challenge); ++**global FC** (glo-action-request security))

⇨ All further GET, SET and ACTION requests depend on the negotiated keys
  - Global keys (++**global FC**)
  - Dedicated keys (++**ded FC**)

3.6. Clients

The logical device can have several associations. The following chapters define in detail the usage and capabilities of these clients. There is a one to one assignment between the clients and the physical interface. The only exception here is the Public Client, which is accessible on the local WZ interface as well as the LAN/WAN interface.

The access rights of these clients are indicated in [1].

There will be no direct access possible between the interfaces, for example from the WZ interface to the LAN/WAN interface or vice versa.
Access to the H1 interface is provided through the objects offered by the logical device.

Below type of clients need to be supported

- Public client (WZ and LAN / WAN)
- Management client (LAN / WAN)
- Data Readout client (LAN / WAN)
- FW Update client (LAN only)
- PLC Management client (LAN only)
- Installation client (WZ)
- Maintenance client (WZ)

- Certification client (WZ)
- CIP - customer information client (H1)

Please find here an overview of the relation between clients and interfaces:

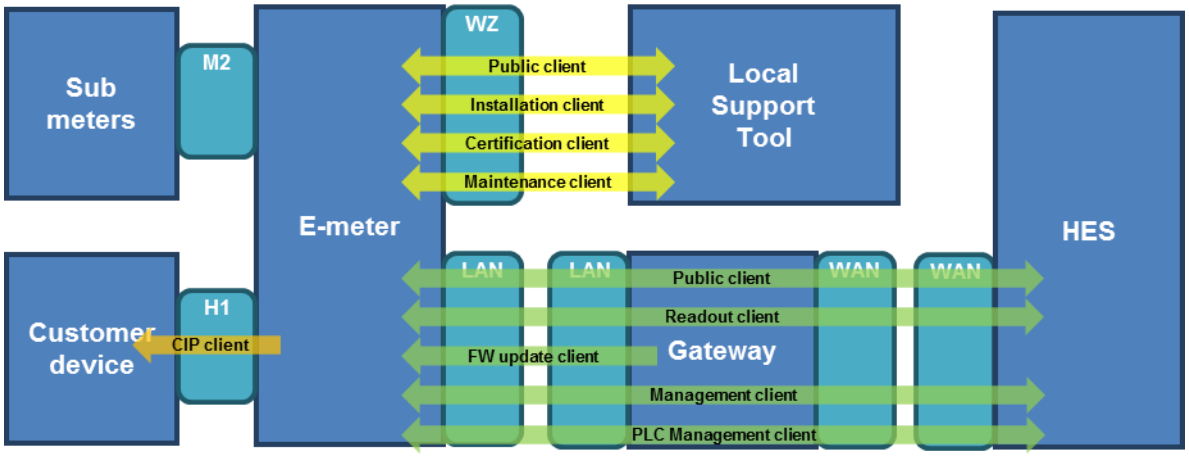


Figure 4: Overview of relation between clients and interfaces for G3-PLC meter

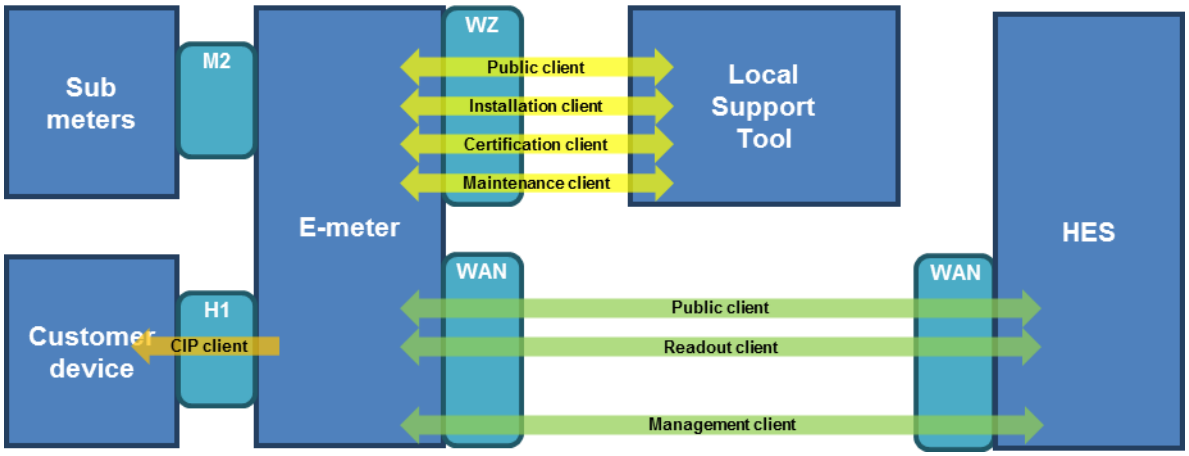


Figure 5: Overview of relation between clients and interfaces for Cellular meter

For managing the association and security context of the clients, the following objects are required

Object / Attribute Name	Class	Ver.	OBIS code
Current Association	15	3	0-0:40.0.0.255
Current Security Setup	64	1	0-0:43.0.0.255
Security setup - Consumer Information	64	1	0-0:43.0.103.255
Association LN - Management Client	15	3	0-0:40.0.1.255
Security setup - Management Client	64	1	0-0:43.0.1.255
Association LN - Data Readout Client	15	3	0-0:40.0.2.255
Security setup - Data Readout Client	64	1	0-0:43.0.2.255
Association LN - PLC Management Client	15	3	0-0:40.0.4.255
Security setup - PLC Management Client	64	1	0-0:43.0.4.255

Association LN - Installation Client	15	3	0-0:40.0.5.255
Security setup - Installation Client	64	1	0-0:43.0.5.255
Association LN - Maintenance Client	15	3	0-0:40.0.6.255
Security setup - Maintenance Client	64	1	0-0:43.0.6.255
Association LN - Certification Client	15	3	0-0:40.0.7.255
Security setup - Certification Client	64	1	0-0:43.0.7.255

Table 7: Security context clients

Current association
There is one association LN object defined, which supports all possible clients. This is the 'Current Association' object.
This object represents the information for the currently open association.

Association LN
The Association LN objects support the association management of the individual client.
It provides the method for changing the HLS_secret which is required for the Installation, Maintenance and Certification client. => supporting HLS mechanism 6 (SHA-256).
The new HLS secret is keywrapped. The key wrapping algorithm is as specified by the security suite. The KEK is the master key.
For HLS mode 6 (SHA-256) the new HLS secret length must be between 128bit (16byte) and 256bit (32byte)

Security Setup
The Management and the Maintenance clients are the highest authorities within the DLMS server. All DLMS server related settings regarding security go through the 'Current Security Setup' object.
For the client security context, individual security setup objects exist that allow the changing the settings and keys for these clients.

For managing the corresponding frame counters, the following objects are required:

Object / Attribute Name	Class	Ver.	OBIS code
Rx frame counter - unicast key - Management Client	12544	0	0-0:43.1.1.255
Rx frame counter - unicast key - Data Readout Client	12544	0	0-0:43.1.2.255
Rx frame counter - unicast key – PLC Management Client	12544	0	0-0:43.1.4.255
Rx frame counter - unicast key - Installation Client	12544	0	0-0:43.1.5.255
Rx frame counter - unicast key - Maintenance Client	12544	0	0-0:43.1.6.255
Rx frame counter - unicast key - Certification Client	12544	0	0-0:43.1.7.255

Table 8: Frame Counter objects

The Frame Counter Readout objects are using a manufacturer-defined class (refer to 11 Appendix 1: Frame Counter Readout for more information).

Rx frame counter - unicast key
Detailed description:

Method description

Method 1: get_frame_counter (data)

The challenge is a 64-byte random (octet-string).
data ::= challenge

challenge
Data type: octet-string
Authorized value: Decoded as Hexa (Size = 64)

Upon invocation of this method, the meter will generate a response to the challenge by performing an HMAC-SHA256(K, m) where K = the AK of the corresponding client and m is the concatenation of Server System Title, Client System Title, received challenge and the frame counter (SysT-S || SysT-C || Challenge || FC) to be returned.
data ::= structure
{
challenge_response:
frame_counter:
}

challenge_response
The challenge response value
Data type: octet-string
Authorized value: Decoded as Hexa (Size = 32)

frame_counter:
The actual frame counter value
Data type: unsigned32
Authorised values: 0x00000000 … 0xFFFFFFFF

3.6.1. Public Client

This client can be used for any communication between **the HES or local support tools and the E-meter**.

Reading basic device configuration information (e.g. SAP, COSEM logical device name, association, serial nrs, ...)

Client L_SAP: 016

⇨ Access:
Accessible on WZ and LAN/WAN interface

⇨ Mandatory Services supported by a Server:
- Block-transfer-with-get
- Get

- General-block-transfer
- Action
- Access

⇨ Establishment:
- AARQ service using LOWEST SECURITY

⇨ Release:
- RLRQ service
- Closing or losing WZ transport layer connection (HDLC connection)
- Closing or losing LAN transport layer connection (G3-PLC)
- A power-down will automatically close the association

⇨ Security settings:
No security; i.e. the COSEM client may access the meter with:
LOWEST SECURITY (Logical_Name_Referencing_NoCiphering, Security policy 0, COSEM_lowest_level_security_mechanism_name(0))

3.6.2. Management Client

This client is used for communication between **the HES and the E-meter**.

Required for the management of the device, setting configuration parameters, retrieving data and execute authorized actions in the meter.
In combination with the Gateway, it allows end-to-end security principle for critical commands like disconnection, load limitation or activation of FW images.

Client L_SAP: 001
⇨ Access:
Accessible on LAN/WAN interface only

⇨ Mandatory Services supported by a Server:
- Block-transfer-with-get
- Block-transfer-with-set
- Get
- Glo-get
- Set
- Glo-set
- Multiple-references
- Selective Access
- Action
- Glo-action
- General-block-transfer
- General-protection
- Access

⇨ Establishment:

- AARQ service using HLS mode **7**

⇨ Release:
- RLRQ service
- Closing or losing LAN transport layer connection (G3-PLC)
- A power-down will automatically close the association

⇨ Security settings:
- using **'Current Security Setup'**
  - Security suite 1
  - Security policy = all messages are authenticated and encrypted
  - Applicable keys:
    - Global Unicast Encryption key
    - Global Authentication key
  - Applicable certificates:
    - Sub CA Certificate 2
    - HES certificate 1
    - HES certificate 2

  ⇨ The client_system_title is transmitted as part of the AARQ and copied into the COSEM object security setup, attribute: client_system_title
  ⇨ From this time instance on the meter uses this client_system_title to decipher the APDUs sent by the corresponding Client.

3.6.3. Data Readout Client

This client is used for communication between **the HES and the E-meter**.

Required for regular data readout of Energy Registers, Load Profile….
The Data Readout client is the recipient of the alarm push messages.

Client L_SAP: 002

⇨ Access:
Accessible on LAN/WAN interface only

⇨ Mandatory Services supported by a Server:
- Block-transfer-with-get
- Block-transfer-with-set
- Get
- Glo-get
- Set
- Glo-set
- Multiple-references
- Selective Access

- Data-Notification
- Action
- Glo-action
- General-block-transfer
- General-protection
- Access

⇨ Establishment:
- AARQ service using HLS mode 7

⇨ Release:
- RLRQ service
- Closing or losing LAN transport layer connection (G3-PLC)
- A power-down will automatically close the association

⇨ Security settings:
- using **'Security Setup - Data Readout client'**
  - Security suite 1
  - Security policy = all messages are authenticated and encrypted
  - Applicable keys:
    - Global Unicast Encryption key
    - Global Authentication key
  - Applicable certificates:
    - Sub CA Certificate 2
    - HES certificate 1
    - HES certificate 2

  ⇨ The client_system_title is transmitted as part of the AARQ and copied into the COSEM object security setup, attribute: client_system_title
  ⇨ From this time instance on the meter uses this client_system_title to decipher the APDUs sent by the corresponding Client.

3.6.4. FW Update Client

This client is used for communication between **the Gateway and the E-meter**.

Only used for sending image blocks for the image transfer object.
This client is suited to support broadcast **without** encryption and authentication

Client L_SAP: 003

⇨ Access:
Accessible on LAN interface only

⇨ Mandatory Services supported by a Server:
- Action

⇨ Establishment:
- Always established (the context is automatically re-established upon power up)

⇨ Release:
- Never released

⇨ Security settings:
No security; i.e. the COSEM client may access the meter with:
LOWEST SECURITY (Logical_Name_Referencing_NoCiphering, Security policy 0, COSEM_lowest_level_security_mechanism_name(0))

3.6.5. PLC Management Client

This client is used for communication between **the PLC Network Management System and the E-meter**.

Required for the management of the PLC network specific configuration parameters and data and the execution of authorized actions in the meter.
In combination with the Gateway, it allows end-to-end security principle for critical PLC network management commands.

Client L_SAP: 004
⇨ Access:
Accessible on LAN interface only

⇨ Mandatory Services supported by a Server:
- Block-transfer-with-get
- Block-transfer-with-set
- Get
- Glo-get
- Set
- Glo-set
- Multiple-references
- Selective Access
- Action
- Glo-action
- General-block-transfer
- General-protection
- Access

⇨ Establishment:
- AARQ service using HLS mode **7**

⇨ Release:
- RLRQ service
- Closing or losing LAN transport layer connection (G3-PLC)
- A power-down will automatically close the association

⇨ Security settings:
- using **'Current Security Setup'**
  - Security suite 1
  - Security policy = all messages are authenticated and encrypted
  - Applicable keys:
    - Global Unicast Encryption key
    - Global Authentication key
  - Applicable certificates:
    - Sub CA Certificate 3
    - HES certificate 1
    - HES certificate 2

  ⇨ The client_system_title is transmitted as part of the AARQ and copied into the COSEM object security setup, attribute: client_system_title
  ⇨ From this time instance on the meter uses this client_system_title to decipher the APDUs sent by the corresponding Client.

3.6.6. Installation Client

This client is used for communication between **a local Installation Tool and the E-meter**. Required for the initial setup and configuration at the customer premises during the installation of the meter.

Client L_SAP: 005

⇨ Access:
Accessible on WZ interface only

⇨ Mandatory Services supported by a Server:
- Block-transfer-with-get
- Block-transfer-with-set
- Get
- Glo-get
- Set
- Glo-set
- Multiple-references
- Selective Access
- Action
- Glo-action
- General-block-transfer
- General-protection
- Access

⇨ Establishment:
- AARQ service using HLS mode 6

⇨ Release:
- RLRQ service
- Closing or losing transport layer connection (HDLC connection)
- A power-down will automatically close the association

⇨ Security settings:
- using **'Association LN - Installation Client'**
  - Authentication method HLS mode 6
  - HLS secret:
    - Key for SHA-256 authentication
- using **'Security Setup – Installation client'**
  - Security suite 1
  - Security policy = all messages are authenticated
  - Applicable keys:
    - Global Unicast Encryption key
    - Global Authentication key

⇨ The client_system_title is transmitted as part of the AARQ and copied into the COSEM object security setup, attribute: client_system_title
⇨ From this time instance on the meter uses this client_system_title to decipher the APDUs sent by the corresponding Client.

3.6.7. Maintenance Client

This client is used for communication between **a local Maintenance Tool and the E-meter**. Required for the initial setup and configuration at the customer premises during the installation of the meter.

Client L_SAP: 006

⇨ Access:
Accessible on WZ interface only

⇨ Mandatory Services supported by a Server:
- Block-transfer-with-get
- Block-transfer-with-set
- Get
- Glo-get
- Set
- Glo-set
- Multiple-references
- Selective Access
- Action
- Glo-action
- General-block-transfer
- General-protection

- Access

⇨ Establishment:
- AARQ service using HLS mode 6

⇨ Release:
- RLRQ service
- Closing or losing transport layer connection (HDLC connection)
- A power-down will automatically close the association

⇨ Security settings:
- using **'Association LN - Maintenance Client'**
  - Authentication method HLS mode 6
  - HLS secret:
    - Key for SHA-256 authentication
- using **'Security Setup – Maintenance client'**
  - Security suite 1
  - Security policy = all messages are authenticated and encrypted
  - Applicable keys:
    - Global Unicast Encryption key
    - Global Authentication key

⇨ The client_system_title is transmitted as part of the AARQ and copied into the COSEM object security setup, attribute: client_system_title
⇨ From this time instance on the meter uses this client_system_title to decipher the APDUs sent by the corresponding Client.

3.6.8. Certification Client

This client is used for communication between **a local support tool and the E-meter** required for the certification process.

Required for the initial setup and configuration at the customer premises during the installation of the meter.

Client L_SAP: 007

⇨ Access:
Accessible on WZ interface only

⇨ Mandatory Services supported by a Server:
- Block-transfer-with-get
- Block-transfer-with-set
- Get
- Glo-get
- Set
- Glo-set

- Multiple-references
- Selective Access
- Action
- Glo-action
- General-block-transfer
- General-protection
- Access

⇨ Establishment:
- AARQ service using HLS mode 6

⇨ Release:
- RLRQ service
- Closing or losing transport layer connection (HDLC connection)
- A power-down will automatically close the association

⇨ Security settings:
- using **'Association LN - Certification Client'**
  - Authentication method HLS mode 6
  - HLS secret:
    - Key for SHA-256 authentication
- using **'Security Setup – Certification client'**
  - Security suite 1
  - Security policy = all messages are authenticated and encrypted
  - Applicable keys:
    - Global Unicast Encryption key
    - Global Authentication key

  ⇨ The client_system_title is transmitted as part of the AARQ and copied into the COSEM object security setup, attribute: client_system_title
  ⇨ From this time instance on the meter uses this client_system_title to decipher the APDUs sent by the corresponding Client.

3.6.9. CIP (Consumer information push) Client

This client is used for communication from **the E-meter to a suitable device connected to the H1 interface.**

Required for periodic transmission of a predefined set of attributes via the customer interface. Foreseen communication is one way only i.e. Push from Server to Client.

Client L_SAP: 103

⇨ Access:
Accessible on H1 interface only

⇨ Mandatory Services supported by a Server:

- Data-Notification
- General-block-transfer
- General-protection
- Attribute0-supported-with-get

⇨ Establishment:
- Always established (the context is automatically re-established upon power up)

⇨ Release:
- Never released

⇨ Security settings:
- using **'Security Setup – Consumer Information'**
  - Security suite 1
  - Security policy = all messages are encrypted
  - Applicable keys:
    - Global Unicast Encryption key

    ⇨ The client_system_title is not required in this setup as this client supports transmit only.

4. Communication profiles and services

4.1. WZ – Service Interface

The DLMS/COSEM communication via the WZ interface is based on HDLC stack using an optical port.

Support of IEC 62056-21 (former. IEC 1107) is not allowed.

4.1.1. HDLC Profile

The HDLC channel is configured and managed via the following COSEM object:

Object / Attribute Name	Class	Ver.	OBIS code
IEC HDLC setup - HDLC Optical port	23	1	0-0:22.0.0.255

Table 9: HDLC objects

4.1.2. Service Interface Deactivation

The meter supports 2 features that deactivate the access via the Service Interface for a period of time

- Remote deactivation by setting a timer value
- Automatic lockout after failed association attempts via the service interface

For managing the configurable parameters, the following object is required:

Object / Attribute Name	Class	Ver.	OBIS code
Optical port temp disable	1	0	0-0:94.43.140.255
Optical port lockout	1	0	0-0:94.43.141.255

Table 10: Deactivation objects

Optical port temp disable
The object optical port temp disable allows the deactivation of the service interface.
The value of attribute 2 works as a count down on the basis of seconds. During this period the service interface is disabled and can't be accessed in any way.
The attribute shows the current remaining time in seconds until the access is enabled again.

Disabling of access:

- By setting the count down value in attribute 2 to a value > 0

Enabling of access:

- By setting the count down value in attribute 2 to the value = 0
- The count down value in attribute 2 reaches the value = 0

Setting this value to its maximum (0xFFFF) prevents the countdown of the register. The access to the service interface is kept disabled until its enabled again by setting the value back to 0 by remote communication.

Optical port lockout
In the event of multiple consecutive faulty association attempts via the service interface, the meter must automatically deactivate the interface for a period of time. After expiry of the period, the service interface is reactivated again.
This object allows the configuration of the lockout parameters which are the number of failed association attempts and the lockout period time.

value ::= structure
{
failed_association_attempts:
lockout_period:
}

failed_association_attempts:
Defines the number of consecutive failed association attempts before the activation of the lockout period.
A value of 0 disables the feature

Data type: unsigned
Authorised values: 0-255

lockout_period:
Defines the lockout period length in number of seconds.

Data type: long_unsigned
Authorised values: 0-65535
The automatic lockout uses the ‘optical port temp disable’ object for the actual deactivation of the service interface. For this, the value defined in the lockout_period is copied to the value attribute of the ‘optical port temp disable’ object and which causes the deactivation.

4.1.3. Operation Mode

Communication between the client and the Meter is supported in the following operation modes:
PULL for 1-way or 2-way communications initiated by the client

Operation mode / usage	DLMS service for HDLC communication
PULL	GET, SET, ACTION, ACCESS

Table 11: Operation Modes

4.2. H1- Consumer Interface

The DLMS/COSEM communication via the H1 interface is based on the wired M-Bus data link layer stack in combination with a wired-Mbus port.

4.2.1. M-Bus Profile

In order to support the DLMS data transfer on the wired M-Bus transport layer, please refer to chapter 10.5 in the Green Book [C].

The foreseen communication is one way only i.e. Push from Server to Client.
In this case, the data is sent using the broadcast functionality of the M-Bus.
The details are available in the following sections of the Green Book [C].

- 10.5.3.4.2 MBUS-DATA service primitives
  Chapter 10.5.3.4.2.1 MBUS-DATA.request and 10.5.3.4.2.3 MBUS-DATA.confirm are applicable as only broadcast needs be supported.
- 10.5.3.4.3 MBUS-DATA protocol specification
  Chapter 10.5.3.4.3.1 Sending COSEM APDUs is applicable as only broadcast needs be supported.
- 10.5.4 Identification and addressing scheme
  - 10.5.4.4 Link Layer Address for M-Bus broadcast
    The Link Layer Address of LLA = 0xFF is reserved for broadcast.
  - 10.5.4.5 Transport layer address
    The Transport layer addressing is using a CI field in the range of 0x00-0x1F without M-Bus data header. In this case, the transport layer can provide segmentation and reassembly.
  - 10.5.4.6 Application addressing extension – M-Bus wrapper
    The DLMS/COSEM AL needs to identify the partners involved in the AA: each AA is bound to a pair of client and server SAPs.
    In this case, the serverSAP = 0x01 (Management Logical Device) and the client SAP = 0x67 (Client L_SAP: 103, CIP Client)

The following object supports setting up the wired M-Bus master interface for data transmission:

Object / Attribute Name	Class	Ver.	OBIS code
M-Bus master port setup - Consumer Information Interface	74	0	0-2:24.6.0.255

Table 12: M-Bus objects CIP

4.2.2. Operation Mode

Communication between the client and the Meter is supported in the following operation mode:
PUSH for 1-way communication initiated by the Meter

Operation mode / usage	DLMS service for wired M-Bus communication
PUSH	DATA-NOTIFICATION (unconfirmed)

Table 13: Operation Modes

4.3. M2 - Multi Utility Interface

The Multi Utility interface is either using the wired or wireless M-Bus protocol

A detailed description of the M-Bus interface is available in [5].

The M-bus data is then mapped to the corresponding COSEM objects in the E-meter. A direct access through the E-meter to the submeters is not supported.

The implementation of this interface follows the IDIS package 2 specification [D]:
Please refer to the following chapters:
- 5.3.1 Wired M-Bus

The following object supports setting up the Multi Utility interface for data communication to and from the M-Bus submeters:

Object / Attribute Name	Class	Ver.	OBIS code
M-Bus master port setup 1 – wired M-Bus	74	0	0-0:24.6.0.255
M-Bus master port setup 2 – wireless M-Bus	74	0	0-0:24.6.0.255
M-Bus client channel x	72	1	0-x:24.1.0.255

Table 14: M-Bus objects

M-Bus client channel
This object hold all necessary configuration and information for setting up and maintaining the connection to a submeter via wired- or wireless M-Bus
Before a submeter can be used, the attributes of the M-Bus client channel object require setting up correctly via the HES:
- Mbus_port_reference:
  This attribute identifies the physical interface the submeter is using
  - For wired M-Bus devices, ‘M-Bus master port setup 1 – wired M-Bus’
  - For wireless M-Bus devices, ‘M-Bus master port setup 2 – wireless M-Bus’

- Capture_definition:
  The M-Bus client allows a flexible adaptation to the submeter data elements via the capture_definition attribute.
  A minimum of up to 4 capture elements must be supported.
- The sequence of the capture configuration is reflected in the instance count of the M-Bus Value objectsCapture_period:
  Specifies in seconds the readout interval of the connected meters
  - For wired M-Bus devices, the value depends on the usage of the retrieved data
  - For wireless M-Bus devices, this value should be set to 0 (externally triggered)
- Primary Address:
  This attribute is only used in combination with the wired M-Bus interface. The Primary address is automatically assigned during the installation process if it was initially set to 0. Setting this attribute to a specific value by the HES is possible if no installation process is required to integrate a M-Bus device.
- Identification_number; manufacturer_id; version and device_type
  These attributes form the M-Bus address. An installation and communication to an M-Bus device is only possible, if these attributes are correctly configured.
- Access_number
  This attribute contains the value of the access number, delivered by the M-Bus header
- Status
  This attribute contains the value of the status byte, delivered by the M-Bus header
- Alarm
  Not supported
- Configuration
  This attribute contains the value of the configuration field, delivered by the M-Bus header.
- Encryption_key_status
  Provides information on the status of the encryption key exchange.

The methods of this object support the installation and maintenance of the connected M-Bus device.

- Slave_install
  - For wired M-Bus devices, this method starts the binding process
  - For wireless M-Bus devices, this method has no effect and returns ‘other reason’
- Slave_deinstall
  This method de-installs the M-Bus slave device and prepares the M-Bus client for the installation of the new M-Bus slave device.
- Capture
  - For wired M-Bus devices, this method triggers an ad-hoc data reading
  - For wireless M-Bus devices, this method has no effect and returns ‘other reason’
- Reset_alarm
  Not supported
- Synchronize_clock
  This method synchronises the M-Bus device clock with the clock of the M-Bus client
- Data_send
  This method sends data to the M-Bus slave device.
- Set_encryption_key
  This method sets the encryption key in the M-Bus client and enables encrypted communication with the M-Bus slave device.
- Transfer_key

This method transfers an encryption key to the M-Bus device. Upon the response of the M-Bus device the attribute 14 (encryption_key_status) is updated accordingly.

Most methods have no effect if the M-Bus device is exclusively operated in an unidirectional mode (with the exception of the set_encryption_key method)

4.4. LAN/WAN- HES Interface

The DLMS/COSEM communication via the LAN and WAN interfaces depends on the used technology

4.4.1. G3-PLC Profile

The DLMS/COSEM communication via the LAN interface is based on UDP over the IPv6 stack in combination with the G3-PLC transport layer.
A detailed description of the DLMS/COSEM related communication profiles and services is available in the G3-PLC Implementation Guide [2].

4.4.2. GPRS Profile

The DLMS/COSEM communication via the WAN interface is based on TCP over the IPv4 or IPv6 stack in combination with the GPRS cellular radio transport layer.
A detailed description of the DLMS/COSEM related communication profiles and services is available in the P2P WAN Implementation Guide [3].

4.4.3. Operation Mode

Communication between the HES and the Meter is supported in the following operation modes:
PULL for 1-way or 2-way communications initiated by the client
PUSH for 1-way communication initiated by the Meter

Operation mode / usage	DLMS service for IP communication
PULL	GET, SET, ACTION, ACCESS
PUSH	DATA-NOTIFICATION (unconfirmed)

Table 15: Operation Modes

5. Meter Functionality

5.1. Identification Numbers

The E-meter requires a number of identification items.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
COSEM logical device name	1	0	0-0:42.0.0.255
Device ID 1, E-meter manufacturing number	1	0	0-0:96.1.0.255
Device ID 2, Comms Module manufacturing number	1	0	0-0:96.1.1.255
Parameter Record Number	1	0	1-0:0.2.1.255

Table 16: Identification Objects

COSEM logical device name (and System title)
This object represents the COSEM logical device name.
The value shares the same requirement as the system title of being unique worldwide. Both values shall originate from the common base of the manufacturer identifier and manufacturer serial number.

COSEM LDN:

- 3 bytes manufacture identifier + 3 bytes meter types + 10 bytes meter serial No. (the last 10 digits in meter serial No.):

MC	MC	MC	MT	MT	MT	SN	SN	SN	SN	SN	SN	SN	SN	SN	SN
K	F	M	1	0	0	0	1	0	0	0	0	0	0	0	1
0x4B	0x46	0x4D	0x31	0x30	0x30	0x30	0x31	0x30	0x30	0x30	0x30	0x30	0x30	0x30	0x31

MC: Manufacturer Code according FLAG coded as ASCII (byte 1,2,3)
MT: Meter Type (byte 4,5,6)
100 single phase meter
200 poly phase DC connected meter
300 poly phase CT connected meter
SN: manufacturer specific serial number ASCII encoded (byte 7,8,10,11,12,13,14,15,16)

System Title:

- 3 bytes manufacture identifier + 5 bytes meter serial No. (the last 10 digits in meter serial No are converted to 10 digits HEX code)

MC	MC	MC	SNb	SNb	SNb	SNb	SNb
K	F	M					
0x4B	0x46	0x4D	0x00	0x05	0xF5	0xE1	0x01

MC: Manufacturer Code according FLAG coded as ASCII (byte 1,2,3)
SNb: manufacturer specific serial number coded as hexadecimal (byte 4,5,6,7,8)

Example meter serial No. 1KFM0100000001
The last 10 digits in the serial No. is (Decimal)0100000001. The HEX code is 0x0005F5E101
Cosem logical name: KFM1000100000001
System title: 4B464D0005F5E101

Device ID 1
This object represents the E-meter identification number according DIN 43863-5 (14-digit alphanumeric number sequence).

Device ID 2
This object represents the Communication Module identification number according DIN 43863-5 (14-digit alphanumeric number sequence).

Parameter Record Number
This number is available for the identification of the applied configuration set.

5.2. Energy Registration

The resolution of the energy registers must be available in Wh and varh.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Active energy import (+A)	3	0	1-0:1.8.0.255
Active energy export (−A)	3	0	1-0:2.8.0.255
Reactive energy import (+R)	3	0	1-0:3.8.0.255
Reactive energy export (−R)	3	0	1-0:4.8.0.255

Table 17: Energy Registration Objects

Active/Reactive energy register
The cumulative energy registers for active and reactive energy
⇨ Register resolution is in line with the resolution on the Display for consistent reading
See chapter 2.3 Display resolution and units in the Display Implementation Guide [4]
- Direct connected meter – wrap around at 999 999 999Wh/varh
- CT connected meter – wrap around at 99 999 999Wh/varh

5.3. Demand Registration

The resolution of the energy registers must be available in W and var.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Demand Register – Active Energy Import (+A)	5	0	1-0:1.4.0.255
Demand Register – Active Energy Export (-A)	5	0	1-0:2.4.0.255
Demand Register – Reactive Energy Import (+R)	5	0	1-0:3.4.0.255
Demand Register – Reactive Energy Export (-R)	5	0	1-0:4.4.0.255
Maximum Demand Register - Active Energy Import (+A)	4	0	1-0:1.6.0.255

Maximum Demand Register - Active Energy Export (-A)	4	0	1-0:2.6.0.255
Maximum Demand Register - Reactive Energy Import (+R)	4	0	1-0:3.6.0.255
Maximum Demand Register - Reactive Energy Export (-R)	4	0	1-0:4.6.0.255
Cumulative Maximum Demand Register - Active Energy Import (+A)	3	0	1-0:1.2.0.255
Cumulative Maximum Demand Register - Active Energy Export (-A)	3	0	1-0:2.2.0.255
Cumulative Maximum Demand Register - Reactive Energy Import (+R)	3	0	1-0:3.2.0.255
Cumulative Maximum Demand Register - Reactive Energy Export (-R)	3	0	1-0:4.2.0.255

Table 18: Demand Objects

Demand Register

The Demand Measurement Period (attribute 8 – period) is defined with 15min (900s). Only one period (attribute 9 – number of periods) is supported.
The attribute status is not used and should remain set to 0
The handling of the demand measurement period in special cases matches the behavior of the Load Profile interval.
Please refer to the following chapter of the IDIS package2 specifications [D]:

- 7.5.8 Events

⇨ Register resolution is in line with the resolution on the Display for consistent reading
See chapter 2.3 Display resolution and units in the Display Implementation Guide [4]
- Direct connected meter – wrap around at 99 999W/var
- CT connected meter – wrap around at 9 999W/var

Maximum Demand Register

The maximum demand register stores the highest measured average demand since the start of current billing period.
At the end of a demand measurement period the last_averge_value is compared with the value in the maximum demand register. If the new value is greater than the value of the maximum demand register the maximum demand register is updated (value and capture time)
The attribute status is not used and should remain set to 0
By invoking the reset method of the maximum demand object, the attribute value is set to 0 and the attribute capture_time is set to the time of the reset execution.

⇨ Register resolution is in line with the resolution on the Display for consistent reading
See chapter 2.3 Display resolution and units in the Display Implementation Guide [4]
- Direct connected meter – wrap around at 99 999W/var
- CT connected meter – wrap around at 9 999W/var

Cumulative Maximum Demand Register

The cumulative maximum demand register contains the sum of all maximum demand register values for the past billing periods.

⇨ Register resolution is in line with the resolution on the Display for consistent reading
See chapter 2.3 Display resolution and units in the Display Implementation Guide [4]
- Direct connected meter – wrap around at 9 999 999W/var
- CT connected meter – wrap around at 999 999W/var

5.4. Date and Time Handling

The meter clock synchronisation follows the same rules as defined in the IDIS package 2 specification [D]:
Please refer to the following chapters:
- 6.6 Meter Clock Synchronization

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Clock	8	0	0-0:1.0.0.255
Clock Time Shift Limit	3	0	1-0:0.9.11.255
Local Time	1	0	1-0:0.9.1.255
Local Date	1	0	1-0:0.9.2.255

Table 19: Date and Time Objects

Time and Date synchronisation via NTP is not in scope for the E-meter

For reading date and time using the attribute 2 - ‘time’, the following conditions apply:
- Current date and time as local time (hundredths of seconds set to 0x00 if not supported)
- Day of Week must be handled correctly
- Deviation must be handled correctly
- Status must be handled correctly.

For setting date and time using the attribute 2 - ‘time’, the following conditions apply:
- Current date and time as local time (hundredths of seconds will be ignored)
- Day of Week will be ignored
- Deviation will be ignored
- Status will be ignored

The following rules concerning the time difference must be considered:
Difference between new time and old time < 2 seconds
⇨ Clock is not adjusted
Difference between new time and old time >= 2 seconds and < ClockTime Shift Limit
⇨ Clock is adjusted without any further actions
Difference between new time and old time >= ClockTime Shift Limit
⇨ Clock is adjusted and “Clock Adjusted” events are triggered..

The clock shift limit is fixed to 9 seconds.

For setting time and date between 02:00 and 03:00 o'clock on the day of the Daylight Saving Change with undefined deviation (0x8000) and undefined status (0xFF), the following conditions apply:

- Normal time to Summer time (setting forward from 02:00 to 03:00 => hour between 02:00 and 03:00 does not exist)
  - ⇨ New date and time is refused with an error response (other-reason)
- Summer time to Normal time (setting backward from 03:00 to 02:00 => hour between 02:00 and 03:00 does exist twice)
  - ⇨ New date and time is accepted and reassumes the currently active season

The objects Local time and Local Date are for display and readout purposes only.
They should not be used for remote communication.

The real time clock keeps running during a power down period of the device.
In case a long power down leads to a discharged power reserve, the real time clock might stop running. In this case, the date and time information is considered as invalid.
As the real date and time information is in fact lost, the RTC must be initialised based on the last known timestamp at the power down.

- ⇨ Power up time = Power down time + 1 second.

5.4.1. Scheduler behaviour on date and time change

The schedulers only execute an action when crossing a scheduled execution time instance.

The application is confronted with a time and date change due to 2 events

1. Changing date and time remotely by setting the clock
2. Resuming date and time after a power fail

The following behaviour is expected for objects using the scheduler class ID 22:

- Power down/up:
  If there is at least one execution time instance scheduled during power down/up period:
  - ⇨ action is executed once after power up.
- Date and Time shift forward:
  If there is at least one execution time instance scheduled during the shift forward period:
  - ⇨ action is executed once after date and time shift.
- Date and Time shift backward:
  - ⇨ NO action

Configuring the scheduler execution time to a time instance in the past does not lead to an execution of an action.

5.5. Calendar and Tariff Handling

The meter calendar and tariff handling follows the same rules as defined in the IDIS package 2 specification [D]:
Please refer to the following chapters:

- 6.2 Remote Tariff Programming

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Tariffication Activity calendar	20	0	0-0:13.0.0.255
Tariffication Special days table	11	0	0-0:11.0.0.255
Tariffication script table	9	0	0-0:10.0.100.255
Register activation – Energy	6	0	0-0:14.0.1.255
Register activation – Maximum Demand	6	0	0-0:14.0.2.255
Currently active energy tariff	1	0	0-0:96.14.0.255
Default tariffication script	1	0	0-0:96.14.15.255
Tariff Activation Event Log	7	1	0-0:99.98.11.255

Table 20: Tariffication Objects

Tariffication Activity calendar

The activity calendar must support at least the following:

- season_profile => at least 4 seasons
- week_profile_table => at least 4 entries, exactly one per season
- day_profile_table => at least 4 entries
- day_profile => at least 5 switching times per day (15min boundaries)

Tariffication Special days table

Special days table must support a minimum of 200 entries in order to cover all fixed and flexible Austrian holidays (Easter Monday, Corpus Christi, Ascension, Whit Monday, …) until 2050.

Tariffication script table

The tariffication script table is limited to only 2 tariff switching scripts.

Script identifier	Action
1	Registers and actions corresponding to tariff 1 are activated
2	Registers and actions corresponding to tariff 2 are activated

Table 21: Tariff Scripts

Register activation

The following tariff rate registers for energy are supported:

Tariff Rate	Energy type	OBIS code	Activation
T1	+A	1-0:1.8.1.255	Daily from 22.00 till 6.00
	-A	1-0:2.8.1.255	Daily from 22.00 till 6.00
	+R	1-0:3.8.1.255	Daily from 22.00 till 6.00
	-R	1-0:4.8.1.255	Daily from 22.00 till 6.00
T2	+A	1-0:1.8.2.255	Daily from 6.00 till 22.00
	-A	1-0:2.8.2.255	Daily from 6.00 till 22.00
	+R	1-0:3.8.2.255	Daily from 6.00 till 22.00
	-R	1-0:4.8.2.255	Daily from 6.00 till 22.00

Table 22: Tariff Objects

Default tariffication script
Holds the script selector number as defined in the Tariffication Script Table which must be activated in the case of invalid tariff information or invalid clock.

Currently active energy tariff
Holds the name of the currently active mask as defined in the Register activation – Energy object

Tariff Activation Event Log
Records every tariff change rate

min capacity:	minimum of 30 entries
structure:	clock.time, value
capture_period:	0 (externally triggered)
captured objects:	clock.time; currently active tariff
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)

Example of T1 and T2 tariff setup:
- 2 tariff registers each for
  - active import energy
  - active export energy
  - reactive import energy
  - reactive export energy
- 2 possible tariff rates (T1 and T2)

```
RegisterActivation-Energy
        Logical_name ::= 0-0:14.0.1.255
        Register_assignment ::=      {
                                            { class_id ::= 3, logical_name ::= 1-0:1.8.1.255},
                                            { class_id ::= 3, logical_name ::= 1-0:1.8.2.255},
                                            { class_id ::= 3, logical_name ::= 1-0:2.8.1.255},
                                            { class_id ::= 3, logical_name ::= 1-0:2.8.2.255},
                                            { class_id ::= 3, logical_name ::= 1-0:3.8.1.255},
                                            { class_id ::= 3, logical_name ::= 1-0:3.8.2.255},
                                            { class_id ::= 3, logical_name ::= 1-0:4.8.1.255},
                                            { class_id ::= 3, logical_name ::= 1-0:4.8.2.255},
                                     }
        Mask_list ::=   {
                                { mask_name ::= “T1”, index_list ::= { 1, 3, 5, 7 } },
                                { mask_name ::= “T2”, index_list ::= { 2, 4, 6, 8 } }
                        }

        Active_mask ::= “T2”
```

Currently Active Tariff
Logical_name ::= 0-0:96.14.0.255
Value ::= “T2”

5.6. Billing Profile

There is one Billing Profile for electricity metering defined.

Data of billing period 1

min capacity:	15 months with monthly billing period, 26 captured objects
structure:	clock.time, values
capture_period:	0 (externally triggered via “End of billing period 1 scheduler”, “Ad-Hoc End of billing period 1” or manually via push button)
captured objects:	clock.time; period_counter; A+ total; A+ rate1; A+ rate2; A- total; A- rate1; A- rate2; R+ total; R+ rate1; R+ rate2; R- total; R- rate1; R- rate2; P+ max value; P+ max timestamp; P- max value; P- max timestamp; Q+ max value; Q+ max timestamp; Q- max value; Q- max timestamp P+ cum max value, P- cum max value, Q+ cum max value, Q- cum max value,
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)

This Profile uses the asynchronous type of capturing which is triggered on a regular basis by a scheduler (synchronously) and/or asynchronously by events.

The synchronous capturing is using a single action scheduler with the date and time trigger configuration set to 00:00:00 o’clock on every 1st of the month.
It is possible to disable the monthly capturing by setting the capture date and time to undefined or an empty array. The status if active or not must be visible on the meter display.

Event that can trigger the asynchronous capturing are
- button press (demand reset button)
- remote command
- additional single action scheduler for triggering at a specific date and time

Each asynchronous capturing activates a reset lock of 15min that prevents any further asynchronous billing captures (for example by another button press).

- Triggering the demand reset by a remote command during the 15min reset lock ignores the request and generates an error response (other-reason)
- Triggering the demand reset by button press or additional single action scheduler during the 15min reset lock ignores the request without further indication.

The reset lock does not affect the synchronous trigger by the scheduler.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Predefined Scripts - MDI reset / end of billing period	9	0	0-0:10.0.1.255
End of billing period 1 scheduler	22	0	0-0:15.0.0.255
Ad-Hoc End of billing period 1 scheduler	22	0	0-0:15.1.0.255
Data of billing period 1	7	1	0-0:98.1.0.255
Billing period counter	1	0	0-0:0.1.0.255

Table 23: Billing Objects

Predefined Scripts - MDI reset / end of billing period
The activation of this script executes the billing period closure process.
The following steps are executed

1. increment billing period counter
2. add maximum demand register values to cumulative maximum demand register values
3. execute capture method of the billing profile
4. execute reset methods of the maximum demand register values

Billing period counter
The numbering starts with 00 and increments with every historical reset. (First historical value identified with 01). The valid range is from 00 to 99, rolling over when the maximum is reached.

5.6.1. Billing Profile Handling

The Billing profile handling follows the definitions of the IDIS package2 specifications [D]:
Please refer to the following chapters:
- 7.6 Billing profile for general metering.

Retrieving the data of the Billing Profile is possible by reading the entire attribute ‘buffer’, or using the selective access by range (access selector 1) or by entry (access selector 2). The support of access by range (including support for selected_ values) and access by entry is mandatory.

For further clarification to the selective access on the Billing Profile, please refer to the following chapter of the IDIS package2 specifications [D]:
- 7.7 Reading profiles with parameterized access “from”-“to”.

5.7. Load Profile

There are 2 Load Profiles for electricity metering defined.

Load profile with period 1 (15min)

capacity:	60 days with 15 min (**exactly** 5760 entries), 6 captured objects
structure:	clock.time, profile_status, values
capture_period:	15 minutes (900 seconds)
captured objects:	clock.time, profile_status, A+, A-, R+, R-
buffer encoding:	normal: clock with every entry

selective access: by range and by entry
sorted method: unsorted (FIFO)
profile_status: please see 5.7.3 **Profile Status**

Load profile with period 2 (24h)
capacity: 60 days with daily entries (**exactly** 60 entries), 6 captured objects
structure: clock.time, profile_status, values
capture_period: daily (86400 seconds) capturing at midnight (local time)
captured objects: clock.time, profile_status, A+, A-, R+, R-
buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)
profile_status: please see 5.7.3 **Profile Status**

Both Profiles are synchronous profiles, which are triggered only on a regular basis at the end of the capture period (Load profile 1, Load profile 2).
Special events (e.g. power outages) do not affect the capturing directly but may lead to special entries in the profile status.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Profile status - Load profile with period 1	1	0	0-0:96.10.1.255
Load profile with period 1 (15min)	7	1	1-0:99.1.0.255
Profile status - Load profile with period 2	1	0	0-0:96.10.2.255
Load profile with period 2 (24h)	7	1	1-0:99.2.0.255

Table 24: Load Profile Objects

5.7.1. Load Profile Handling

The load profile handling follows the definitions of the IDIS package2 specifications [D]:
Please refer to the following chapters:
- 7.5 Synchronous Load Profiles.

IDIS [D] defines here the general usage of the Load Profile as a synchronous load profile.
In this Companion Standard only the mandatory features for buffer encoding and sort method are considered.

buffer encoding: normal: clock with every entry
Please refer to the following chapters 7.5.6 Access to the stored values and 7.5.6.1 Normal Read
sorted method: unsorted (FIFO)
Please refer to the following chapters 7.5.2 Sort Order and 7.5.2.2 Unsorted

Retrieving the data of the Load Profiles is possible by reading the entire attribute 'buffer', or using the selective access by range (access selector 1) or by entry (access selector 2). The support of access by range (including support for selected_ values) and access by entry is mandatory.

For further clarification to the selective access on the Load Profile, please refer to the following chapter of the IDIS package2 specifications [D]:
- 7.7 Reading profiles with parameterized access "from"-"to".

5.7.2. OptIN/Opt OUT on Consumption Profile Registration

The consumer can opt-in or opt-out (scheduled or on demand) on the profile registration of his consumption values.
This may apply to load profile 1 (1-0:99.1.0.255) and/or load profile 2 (1-0:99.2.0.255)

Due to legal restrictions, it is not allowed to modify the capture_period of the affected profiles to enable/disable the capturing of data.

The capturing of data in the load profiles is enabled/disabled by executing the corresponding script in the Loadprofile control script table.

To support scheduled operation, a single action scheduler object (class_id 22) and a script table object (class_id 9) offering the necessary scripts to opt-in and to opt-out is used.

The enabling and disabling is recorded in the corresponding event logs (standard event log) according to chapter 5.10.

The load profile applies the new opt in/out setting immediately for the next scheduled interval capturing.

Executing the enable/disable script does not interrupt the currently running interval.

The information of the current Opt IN/OUT status of the profiles is available remotely by reading the Load profile control status object.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Load profile control status	1	0	0-0:96.5.3.255
Load profile control schedule	22	0	0-0:15.0.5.255
Load profile control script table	9	0	0-0:10.0.109.255

Table 25: Load Profile Handling Objects

Script identifier	Action
1	Activate the capturing in load profile 1 and 2

2	Deactivate the capturing in load profile 1 and 2
3	Activate the capturing in load profile 1
4	Deactivate the capturing in load profile 1
5	Activate the capturing in load profile 2
6	Deactivate the capturing in load profile 2

Table 26: Load Profile Scripts

5.7.3. Profile Status

In all load profiles a simplified status code is used for every entry.
The Profile status code has a size of 1 byte and it is shown in hexadecimal form.

The following table describes the state and the function of all bits:

Flag	Description
Bit 7 PDN	**Power down:** This bit is set to indicate that a total power outage has been detected during the affected capture period.
Bit 6	**Reserved:** The reserved bit is always set to 0.
Bit 5 CAD	**Clock adjusted**: The bit is set when the clock has been adjusted by more than the synchronization limit.
Bit 4 CDI	**Capturing disabled**: Indicates the status of the load profile opt in/out setting. The bit is set if the data capturing is disabled (**opt out**) and cleared during normal operation (**opt in**)
Bit 3 DST	**Daylight saving**: Indicates whether or not the daylight saving time is currently active. The bit is set if the daylight saving time is active (summer) and cleared during normal time (winter).
Bit 2 DNV	**Data not valid**: Indicates that the current entry may not be used for billing purposes without further validation because a special event has occurred.
Bit 1 CIV	**Clock invalid:** The power reserve of the calendar clock has been exhausted. The time is declared as invalid. At the same time the DNV bit is set.
Bit 0 ERR	**Critical error**: A serious error such as a hardware failure or a checksum error has occurred. If the ERR bit is set then also the DNV bit is set.

Table 27: Profil Status Flags

The usage of the bits in this status code follows the definition in the IDIS package 2 [D] with the exception of the status BIT 4- CDI which is marked as ‘reserved’.

5.7.4. Load Profile Event Handling

IDIS describes the behaviour of the profile and the setting of the status bits considering different events.
Please refer to the following chapter of the IDIS package2 specifications [D]:
- 7.5.8 Events

5.7.4.1. Applying Opt IN / Opt OUT

Changing the Opt IN/OUT status of either Load Profile 1 and/or Load Profile 2 will trigger a change in the CDI flag for the next scheduled interval entry in the corresponding profile following the status change.

The figure and table below show an OptOUT/Opt IN change event (from 01:15 to 04:52) affecting all capture periods between 01:00 and 05:00. For the capture periods which completely fall into the disabled period (03:00, 04:00), no entry is registered in the load profile buffer.

Table 28: OptIN / OptOUT change example

Date / Time	Status bits							Register_1	Register_2
	PDN	CDI	CAD	DST	DNV	CIV	ERR		
2016-02-15 / 00:00:00	0	0	0	0	0	0	0	2180	110
2016-02-15 / 01:00:00	0	0	0	0	0	0	0	2201	118
2016-02-15 / 02:00:00	0	1	0	0	0	0	0	2212	129
2016-02-15 / 05:00:00	0	1	0	0	0	0	0	2421	133
2016-02-15 / 06:00:00	0	0	0	0	0	0	0	2467	134
2016-02-15 / 07:00:00	0	0	0	0	0	0	0	2548	162

Table 29: Example for load profile 1 (1h)

Here the same example for 15min integration period.

Date / Time	Status bits							Register_1	Register_2
	PDN	CDI	CAD	DST	DNV	CIV	ERR		
2016-02-15 / 01:00:00	0	0	0	0	0	0	0	2201	118
2016-02-15 / 01:15:00	0	0	0	0	0	0	0	2212	129
2016-02-15 / 01:30:00	0	1	0	0	0	0	0	2213	132
2016-02-15 / 05:00:00	0	1	0	0	0	0	0	2421	133
2016-02-15 / 05:15:00	0	0	0	0	0	0	0	2467	134
2016-02-15 / 05:30:00	0	0	0	0	0	0	0	2548	162

Table 30: Example for load profile 1 (15min)

In the special case that the OptOUT/Opt IN change event should happen during a power failure of the device, the marking in the profile occurs on the next scheduled entry after the power up.

5.7.4.1. Crossing midnight boundary

The first entry of a new day is always at 00:00:00.

Here the example for 15min integration period.

Date / Time	Status bits							Register_1	Register_2
	PDN	CDI	CAD	DST	DNV	CIV	ERR		
2016-02-15 / 23:15:00	0	0	0	0	0	0	0	2201	118
2016-02-15 / 23:30:00	0	0	0	0	0	0	0	2212	129
2016-02-15 / 23:45:00	0	0	0	0	0	0	0	2213	132
2016-02-16 / 00:00:00	0	0	0	0	0	0	0	2421	133
2016-02-16 / 00:15:00	0	0	0	0	0	0	0	2467	134
2016-02-16 / 00:30:00	0	0	0	0	0	0	0	2548	162

Table 31: Example for load profile 1 (15min)

Here the example for 24h integration period.

Date / Time	Status bits							Register_1	Register_2
	PDN	CDI	CAD	DST	DNV	CIV	ERR		
2016-02-15 / 00:00:00	0	0	0	0	0	0	0	2201	118
2016-02-16 / 00:00:00	0	0	0	0	0	0	0	2212	129
2016-02-17 / 00:00:00	0	0	0	0	0	0	0	2213	132
2016-02-18 / 00:00:00	0	0	0	0	0	0	0	2421	133

Table 32: Example for load profile 2 (24h)

5.7.4.1. Season Change

The season change (DST change) follows the IDIS definition:

The activation or deactivation of the daylight saving time does not create any additional entries in the buffer. The timestamp together with the DST bit contains enough information to identify clearly, when the season change occurred and if the buffer data was captured when daylight saving time was active or not. The time stamp shows the time before the change.

Here the example for 15min integration period for the change from **normal** to **summer** time:

Date / Time	Status bits							Register_1	Register_2
	PDN	CDI	CAD	DST	DNV	CIV	ERR		
2016-02-27 / 01:15:00	0	0	0	0	0	0	0	2201	118
2016-03-27 / 01:30:00	0	0	0	0	0	0	0	2212	129
2016-02-27 / 01:45:00	0	0	0	0	0	0	0	2213	132
2016-03-27 / 03:00:00	0	0	0	1	0	0	0	2421	133
2016-02-27 / 03:15:00	0	0	0	1	0	0	0	2467	134
2016-03-27 / 03:30:00	0	0	0	1	0	0	0	2548	162
2016-02-27 / 03:45:00	0	0	0	1	0	0	0	2596	187

Table 33: Example for load profile 1 (15min)

Here the example for 15min integration period for the change from **summer** to **normal** time:

Date / Time	Status bits							Register_1	Register_2
	PDN	CDI	CAD	DST	DNV	CIV	ERR		
2016-10-30 / 01:15:00	0	0	0	1	0	0	0	2201	118
2016-10-30 / 01:30:00	0	0	0	1	0	0	0	2212	129

2016-10-30 / 01:45:00	0	0	0	1	0	0	0	2213	132
2016-10-30 / 02:00:00	0	0	0	1	0	0	0	2221	133
2016-10-30 / 02:15:00	0	0	0	1	0	0	0	2421	134
2016-10-30 / 02:30:00	0	0	0	1	0	0	0	2467	162
2016-10-30 / 02:45:00	0	0	0	1	0	0	0	2548	187
2016-10-30 / 02:00:00	0	0	0	0	0	0	0	2596	198
2016-10-30 / 02:15:00	0	0	0	0	0	0	0	2634	235
2016-10-30 / 02:30:00	0	0	0	0	0	0	0	2654	254
2016-10-30 / 02:45:00	0	0	0	0	0	0	0	2692	267
2016-10-30 / 03:00:00	0	0	0	0	0	0	0	2786	291
2016-10-30 / 03:15:00	0	0	0	0	0	0	0	2933	311

Table 34: Example for load profile 1 (15min)

5.8. Disconnector and Limiter

Disconnection and reconnection of the electricity supply is supported by the following objects:

Object / Attribute Name	Class	Ver.	OBIS code
Disconnect control activity calendar	20	0	0-0:13.0.1.255
Disconnect control special days table	11	0	0-0:11.01.255
Disconnect control scheduler	22	0	0-0:15.0.1.255
Disconnect control script table	9	0	0-0:10.0.106.255
Disconnect control	70	0	0-0:96.3.10.255
Limiter Import	71	0	0-0:17.0.0.255
Limiter Export	71	0	0-0:17.0.1.255
Event Object - Disconnect Control Log	1	0	0-0:96.11.2.255
Disconnect Control Log	7	1	0-0:99.98.2.255

Table 35: Disconnect Objects

The state diagram and the possible state transitions are shown in the figure below:

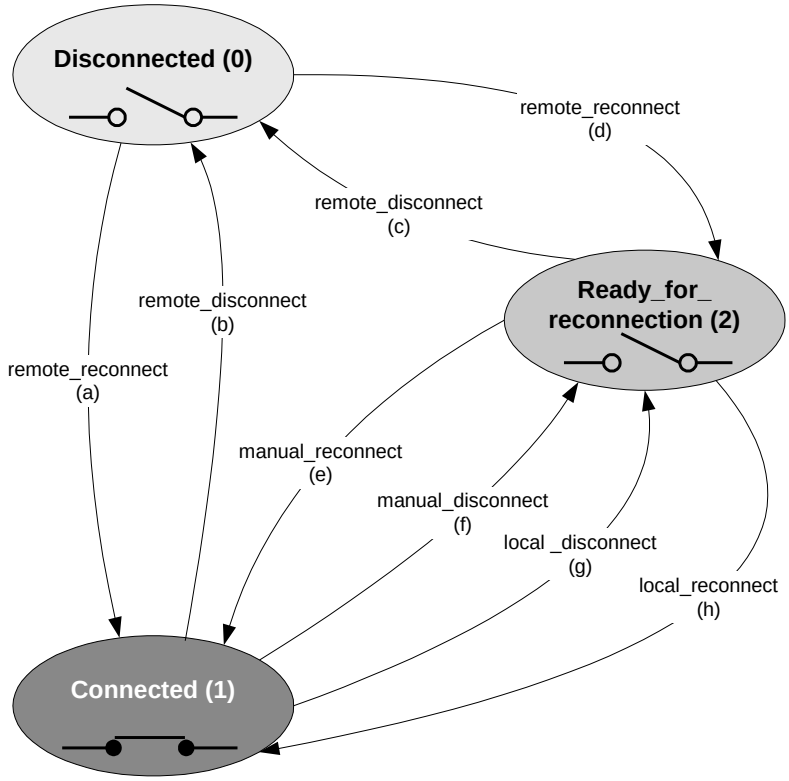


Figure 6: Disconnector state transitions

Disconnect script table

The disconnect script table contains the scripts which act on the Disconnect Control object as follows:

Script identifier	Action
1	SET control_state to "Ready_for_reconnection (2)" Performs a local disconnection according to transition "**local_disconnect (g)**".
2	SET control_state to "Connected (1)" Performs a local reconnection according to transition "**local_reconnect (h)**"
3	execute method "remote_disconnect(0)" Performs a remote disconnection according to transition "**remote_disconnect (b)**" or "**remote_disconnect (c)**", depending on the control mode setting.
4	execute method "remote_reconnect(0)"_ Performs a remote reconnection according to transition "**remote_reconnect (a)**" or "**remote_reconnect (d)**", depending on the control mode setting.

Table 36: Disconnect Scripts

If the state transition is not allowed by the control mode, then the action is ignored.

The action service to method 1 "execute(data)" of the Disconnect script table object is not allowed for any remote client

Disconnect control activity calendar
Using a dedicated activity calendar for the disconnector, the meter can serve as a possible replacement for a ripple control unit or a timer switch (an interruptable load is connected directly to the breaker of the meter, which means that a tipple control unit or a timer switch in combination with a switching contactor is not required anymore).
In this case, the disconnector of the meter follows a switching program, which is preset by the central system and stored in the meter. However, it must be possible to overwrite the switching program for the disconnector by a remote command from the central system (eg disconnector "OFF" or "ON"). The next opposite command (either from the internal switching table or remotely) changes the state of the breaker.

The activity calendar must support at least the following:
- season_profile => at least 4 seasons
- week_profile_table => at least 4 entries, exactly one per season
- day_profile_table => at least 4 entries
- day_profile => at least 5 switching times per day

Disconnect control special days table
Allows the definition of special days for the disonnector control activity calendar.

Special days table must support a minimum of 200 entries in order to cover all fixed and flexible Austrian holidays (Easter Monday, Corpus Christi, Ascension, Whit Monday, …) until 2050.

Disconnect control scheduler
With the help of the single action scheduler the remote operation of the disconnector can be executed at a specific, delayed time instance. In this case the actual dis/re-connection (triggered by the single action schedule via script 3 or 4) is still interpreted as a remote operation.
- Only the access to script 3 and 4 is allowed for attribute 2.

Disconnect control
The behaviour of the disconnector on any remote, local and manual disconnection or reconnection commands is dependent by the control_mode setting of the object.

control _mode	Disconnection				Reconnection			
	Remote		Manual	Local	Remote		Manual	Local
enum:	(b)	(c)	(f)	(g)	(a)	(d)	(e)	(h)
(0)	–	–	–	–	–	–	–	–
(1)	x	x	x	x	–	x	x	–
(2)	x	x	x	x	x	–	x	–
(3)	x	x	–	x	–	x	x	–
(4)	x	x	–	x	x	–	x	–
(5)	x	x	x	x	–	x	x	x
(6)	x	x	–	x	–	x	x	X
NOTE 3 In Mode (0) the disconnect control object is always in 'connected' state.								
NOTE 4 Local disconnection is always possible unless the corresponding trigger is inhibited.								

Figure 7: Disconnection commands

The following behaviour of the disconnector is specified for **remote** management, using direct commands or with the help of the scheduler.

- Disconnection (by remote disconnection, either directly or scheduled)
  => remote_disconnect (b)
- Ready for reconnection (by remote disconnection, either directly or scheduled)
  => remote_reconnect (d)
- Reconnection (by manual button press on the E-meter)
  => manual reconnect (e)

The following behaviour of the disconnector is specified for local management (load limitation using the limiter object).

- Disconnection / Ready for reconnection (triggered by limiter object)
  => local_disconnect (g)
- Reconnection (by manual button press on the E-meter)
  => manual reconnect (e)

⇨ **Default control mode = 3**

However, disabling and direct reconnection of the disconnector must be possible as well.

⇨ Mode 0 – for disabling

⇨ Mode 4 – for supporting of direct reconnection in case of remote reconnection (remote_reconnect (a))

To avoid any inconsistencies between control_mode and control_state, any changes to the control mode are only allowed when the current control_state is supported by the new control_mode.

Limiter import and export

With the help of the limiter, the local operation of the disconnector can be executed depending on configurable consumption of the consumed Power. In this case the actual dis/re-connection triggers the script 1 for the local disconnection.

Limiter import: The monitored value is the 'Instantaneous active import power (+P)' with the obis code 1-0:1.7.0.255.
Limiter export: The monitored value is the 'Instantaneous active export power (-P)' with the obis code 1-0:2.7.0.255.

Setting the threshold to 0 disables the functionality of the limiter

All event concerning the disconnector functionality will be recoded into the Disconnect Control log:

Disconnect Control log

min capacity:	minimum of 10 entries
structure:	clock.time, value
capture_period:	0 (externally triggered)
captured objects:	clock.time; disconnect event
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)

5.9. Power Quality

5.9.1. Instantaneous Power values

For the monitoring of the instantaneous power values, the following functionality must be available:

- Instantaneous Voltage per phase
- Instantaneous Current total and per phase
- Instantaneous Frequency
- Instantaneous power +P/-P/+Q/-Q/+S/-S total and per phase
- Instantaneous power factor total and per phase
- Instantaneous phase angle (U-I) L1/L2/L3

The Instantaneous Readout Profile allows capturing a snapshot of the instantaneous values for a consistent readout. The capturing is triggered by executing the corresponding method of the profile generic object.

Instantaneous measurements are supported by the following objects:

Object / Attribute Name	Class	Ver.	OBIS code
Instantaneous voltage L1,L2*,L3*	3	0	1-0:x.7.0.255 x=32,52,72
Instantaneous current L1,L2*,L3*	3	0	1-0:x.7.0.255

			x=31,51,71
Instantaneous current (sum over all phases)	3	0	1-0:90.7.0.255
Instantaneous net frequency; any phase	3	0	1-0:14.7.0.255
Instantaneous power +P;-P;+Q;-Q;+S;-S total,L1,L2*,L3*	3	0	1-0:x.7.0.255 x=1,2,3,4,9,10 21,22,23,24,29,30 31,32,33,34,39,40 41,42,43,44,49,50
Instantaneous power (|+P|+|-P|)	3	0	1-0:15.7.0.255
Instantaneous power factor import (+P/+S) total,L1,L2*,L3*	3	0	1-0:x.7.0.255 x=13,33,53,73
Instantaneous power factor export (|-P/-S|) total,L1,L2*,L3*	3	0	1-0:x.7.0.255 x=84,85,86,87
Instantaneous phase angle (U-I) L1,L2*,L3*	3	0	1-0:81.7.x.255 x=40,51,62
Instantaneous Readout Profile	7	0	0-0:21.0.5.255

(*) Only required for Poly Phase meters (PP)

Instantaneous Readout Profile

capacity: 1 entry, 45 capture objects
structure: clock.time, values
capture_period: on external capture only
captured objects: clock.time, all listed instantaneous values (depending on SP or PP),
buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)

5.9.2. Voltage Cut, Sag and Swell detection

For the monitoring of voltage cut, sags and swells, the following functionality must be available:

- Number of voltage cuts, sags and swells L1/L2/L3
- Duration of voltage cuts, sags and swells L1/L2/L3
- Magnitude of last voltage sags and swells L1/L2/L3

Further, the following configuration items must be supported:

- Configuration of cut, sag and swell thresholds
- Configuration of cut, sag and swell time thresholds

The events will be recoded in a specific Power Quality event log

Voltage sag and swell detection is supported by the following objects:

Object / Attribute Name	Class	Ver.	OBIS code
Threshold for voltage sag	1	0	1-0:12.31.0.255
Time threshold for voltage sag	1	0	1-0:12.43.0.255
Number of Voltage Sags L1/L2*/L3*	1	0	1-0:x.32.0.255 x=32,52,72
Duration of last Voltage Sags L1/L2*/L3*	3	0	1-0:x.33.0.255

			x=32,52,72
Magnitude of last Voltage Sags L1/L2*/L3*	3	0	1-0:x.34.0.255 x=32,52,72
Threshold for voltage swell	1	0	1-0:12.35.0.255
Time threshold for voltage swell	1	0	1-0:12.44.0.255
Number of Voltage Swells L1/L2*/L3*	1	0	1-0:x.36.0.255 x=32,52,72
Duration of last Voltage Swells L1/L2*/L3*	3	0	1-0:x.37.0.255 x=32,52,72
Magnitude of last Voltage Swells L1/L2*/L3*	3	0	1-0:x.38.0.255 x=32,52,72
Threshold for missing voltage (voltage cut)	3	0	1-0:12.39.0.255
Time threshold for voltage cut	3	0	1-0:12.45.0.255
Number of Voltage Cuts L1/L2*/L3*	1	0	1-0:x.40.0.255 x=32,52,72
Duration of last Voltage Cuts L1/L2*/L3*	3	0	1-0:x.41.0.255 x=32,52,72
Event Object - Power Quality Log	1	0	0-0:96.11.4.255
Power Quality Log	7	1	0-0:99.98.4.255

(*) Only required for Poly Phase meters (PP)

The detection of a Voltage Cut event prevails the functionality of the Voltage Sag detection.

- In the case that a Voltage Cut event is detected first, the Voltage Sag event entry is not recorded
- In the case that a Voltage Cut and a Voltage Sag event is detected at the same time, the Voltage Sag event entry is not recorded
- In the case that a Voltage Sag event is detected first, the Voltage Cut event entry is also recorded

On re-establishment of the Voltage to its normal condition (no active Under-, Overvoltage or Voltage Cut), the Voltage Normal event is recorded.
A hysteresis of 2% to the threshold values and a stabilisation period of 5s is applied to declare the re-establishment of the voltage after a Under-, Overvoltage or Voltage Cut condition.

In case of a complete device power down, it's assumed that for a Single Phase meter the voltage in L1 and for a Poly Phase meter the voltages in L1, L2 and L3 fell below the Voltage Cut threshold. Generating Voltage Cut entries in the Power Quality event log depends on the duration of the power down time being longer than the Voltage Cut time threshold or not.

- if not, no entries in the Power Quality log are required
- if yes, Voltage Cut/Normal event entries in the Power Quality log are required.

Voltage Cut and Voltage Normal events are logged with the power down and power up timestamps as part of the power up procedure.

The event time stamps in the power quality event log represent the start time of the Sag-, Swell-, Cut- or Normal condition. It is not the timestamp at logging the event into the power quality log.

Power Quality log
min capacity: minimum of 100 entries

structure: clock.time, value
capture_period: 0 (externally triggered)
captured objects: clock.time; power quality event
buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)

5.9.3. Power fail detection

For the monitoring of power fails, the following functionality must be available:

- Number of power fails
- Number of long power fails

Further, the following configuration items must be supported:

- Configuration of time thresholds for long power fails

The events will be recoded in a specific Power Failure event log

Power fail detection is supported by the following objects:

Object / Attribute Name	Class	Ver.	OBIS code
Time threshold for long power failure	3	0	0-0:96.7.20.255
Number of power failures	1	0	0-0:96.7.21.255
Number of long power failures	1	0	0-0:96.7.9.255
Duration of last long power failure	1	0	0-0:96.7.19.255
Power Failure Event Log	7	1	1-0:99.97.0.255

Table 37: Power Failure objects

Power Failure log
min capacity: minimum of 10 entries
structure: clock.time, value
capture_period: 0 (externally triggered)
captured objects: clock.time; long power failure duration
buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)

5.9.4. Power Quality profile

The Power Quality Profile periodically records objects based on the options below:

- the average Voltage per phase
- the average Current per phase
- the min Voltage per phase
- the max Voltage per phase

The period length for the averaging of voltage and current as well as the monitoring interval for the min / max detection is based on the configuration of the object “Measurement Period 3 for Instantaneous values”

Power Quality

capacity:	8 days with 10 min (1152 entries), based on 12 captured objects
structure:	clock.time, profile_status, values
capture_period:	1,5,10,15,30,60 min, the value is equal to the measurement period of all the captured objects. Further, the measurement period of the captured objects and the capture_period must be synchronised. Setting this value to 0 disables the profile recording
captured objects:	clock.time, profile_status, Average V per phase, Min V per phase, Max V per phase
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)
profile_status:	please see 5.7.3 **Profile Status**

The handling of the profile capture period in special cases follows the definition of the load profile event handling in chapter 5.7.4 **Load Profile Event Handling**.

Power quality profiling is supported by the following objects:

Object / Attribute Name	Class	Ver.	OBIS code
Profile status – Power Quality profile	1	0	0-0:96.10.3.255
Power Quality profile	7	1	1-0:99.14.0.255
Measurement Period 3 for Instantaneous values	1	0	1-0:0.8.2.255
Average voltage L1/L2*/L3*	3	0	1-0:x.24.0.255 x=32,52,72
Average current L1/L2*/L3*	3	0	1-0:x.24.0.255 x=31,51,71
Min voltage L1/L2*/L3*	3	0	1-0:x.23.0.255 x=32,52,72
Max voltage L1/L2*/L3*	3	0	1-0:x.26.0.255 x=32,52,72

(*) Only required for Poly Phase meters (PP)

Measurement Period 3 for Instantaneous values

The interval period for the average Voltage/Current calculation and the min/max Voltage detection is defined by the object “Measurement Period 3 for Instantaneous values. The handling of the averaging period in special cases follows the definition of the load profile event handling in chapter 5.7.4 **Load Profile Event Handling**.

Average voltage and current

The algorithm of the averaging calculation is based on the number of captured instantaneous voltage and current samples within the interval period. This allows to generate meaningful data in the event of a disrupted interval due to power fail or a time change event.
For example:

⇨ Voltage of 220V and Current of 10A with an averaging period of 10min
When there is power off for 5min within this period, the results for the average voltage and current would be still 220V and 10A based on the collected samples while the device was powered.

Min and Max Voltage

The algorithm of the voltage min/max calculation is based on comparing the captured instantaneous voltage samples to the last recorded values:

- For the Voltage minimum: update if the voltage sample value is smaller
- For the Voltage maximum: update if the voltage sample value is higher

At the start of a new integration period, the min/max registers are initialized with the first captured instantaneous voltage sample.

5.10. Standard Event Log

The event codes used for the Standard Event Log can be found in [1].

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Event Object - Standard Event Log	1	0	0-0:96.11.0.255
Standard Event Log	7	1	0-0:99.98.0.255

Standard Event Log

min capacity:	minimum of 100 entries
structure:	clock.time, value
capture_period:	0 (externally triggered)
captured objects:	clock.time; standard event
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)

5.11. Fraud Detection Event Log

The event codes used for the Fraud Detection Log can be found in [1].

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code

Event Object - Fraud Detection Log	1	0	0-0:96.11.1.255
Fraud Detection Log	7	1	0-0:99.98.1.255

Fraud detection Event Log

min capacity: minimum of 30 entries
structure: clock.time, value
capture_period: 0 (externally triggered)
captured objects: clock.time; fraud detection event
buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)

5.12. Specific Security Event Log and Event Counter

This event log follows the specific security event logging functionality.

It's required to have all specified security relevant events registered to this security event log, even if they are actually already available in other dedicated logs.

The security event log organises the individual events into event groups. Each event group has an associated event counter (Group Event Counter – G_EC) that will be incremented if one of the associated events is captured to this log.
The management client may reset the counters for monitoring the number of events for a specific period of time.

The following events must be registered in the Security Event Log:

Event Code	Name	Description
Registration of successful or failed authentication for a specific client (G_EC_01)		
26	Communication started on remote interface LAN/WAN	Indicates that the communication was started on the remote interface LAN/WAN
27	Communication ended on remote interface LAN/WAN	Indicates that the communication has ended on the remote interface LAN/WAN
28	Communication started on local interface WZ	Indicates that the communication was started on the locale interface WZ
29	Communication ended on local interface WZ	Indicates that the communication has ended on the local interface WZ
46	Association authentication failure (n time failed authentication)	Indicates that a user tried to gain access with wrong credentials (intrusion detection) or HLS access challenge processing failed
FW Upgrade (G_EC_02)		

17	Firmware ready for activation	Indicates that the new firmware has been successfully downloaded and verified, i.e. it is ready for activation
18	Firmware activated	Indicates that a new firmware has been activated
51	FW verification failed	Indicates the transferred firmware verification failed i.e. cannot be activated.
Manual change of date and time (G_EC_03)		
4	Clock adjusted (old date/time)	Indicates that the clock has been adjusted. The date/time that is stored in the event log is the old date/time before adjusting the clock.
5	Clock adjusted (new date/time)	Indicates that the clock has been adjusted. The date/time that is stored in the event log is the new date/time after adjusting the clock.
Fraud attempts (G_EC_04)		
40	Terminal cover removed	Indicates that the terminal cover has been removed.
41	Terminal cover closed	Indicates that the terminal cover has been closed.
42	Strong DC field detected	Indicates that a strong magnetic DC field has been detected.
43	No strong DC field anymore	Indicates that the strong magnetic DC field has disappeared.
44	Meter cover removed	Indicates that the meter cover has been removed.
45	Meter cover closed	Indicates that the meter cover has been closed.
Start-up, Reset or Reboot (G_EC_05)		
1	Power Down	Indicates a complete power down of the device. Please note that this is related to the device and not necessarily to the network.
2	Power Up	Indicates that the device is powered again after a complete power down.
15	Watchdog error	Indicates a watch dog reset or a hardware reset of the microcontroller.
Error and alarm register reset (G_EC_06)		
10	Error register cleared	Indicates that the error register was cleared.
11	Alarm register cleared	Indicates that the alarm register was cleared.
Device specific failures (G_EC_07)		
12	Program memory error	Indicates a physical or a logical error in the program memory.
13	RAM error	Indicates a physical or a logical error in the RAM.
14	NV memory error	Indicates a physical or a logical error in the non volatile memory
16	Measurement system error	Indicates a logical or physical error in the measurement system
49	Decryption or authentication failure (n time failure)	Decryption with currently valid key (global or dedicated) failed to generate a valid APDU or authentication tag
50	Replay attack	Receive frame counter value less or equal to the last successfully received frame counter in the received APDU Event signalizes as well the situation when the DC has lost the frame counter synchronization.
Reconfiguration of cryptographically relevant parameters (G_EC_08)		
48	Global key(s) changed	One or more global keys changed
Disconnector specific events (G_EC_09)		
59	Disconnector ready for manual reconnection	Indicates that the disconnector has been set into the Ready_for_reconnection state and can be manually reconnected
60	Manual disconnection	Indicates that the disconnector has been manually disconnected.
61	Manual connection	Indicates that the disconnector has been manually connected.
62	Remote disconnection	Indicates that the disconnector has been remotely disconnected.
63	Remote connection	Indicates that the disconnector has been remotely connected.
64	Local disconnection	Indicates that the disconnector has been locally disconnected (i.e. via the limiter or current supervision monitors).

68	Disconnect/Reconnect failure	Indicates that a failure of disconnection or reconnection has happened (control state does not match output state)
69	Local reconnection	Indicates that the disconnector has been locally re-connected (i.e. via the limiter or current supervision monitors).
Limiter specific events (G_EC_10)		
65	Limiter threshold exceeded	Indicates that the limiter threshold has been exceeded.
66	Limiter threshold ok	Indicates that the monitored value of the limiter dropped below the threshold.
67	Limiter threshold changed	Indicates that the limiter threshold has been changed

Table 38: Security Event Log

The security event log further registers the following additional information with each event entry:

- client SAP/server SAP => out of the Current Association LN
- client system title => out of the Current Security Setup LN

In case the triggering event does not provide the relevant information for client_SAP/server_SAP and/or client_system_title, values must be set to 0xFF.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Event Object - Security Event Log	1	0	0-0:96.11.9.255
Security Event Log	7	1	0-0:99.98.9.255
Security Group Event Counter Object - (G_EC_01)	1	0	0-0:96.15.21.255
Security Group Event Counter Object - (G_EC_02)	1	0	0-0:96.15.22.255
Security Group Event Counter Object - (G_EC_03)	1	0	0-0:96.15.23.255
Security Group Event Counter Object - (G_EC_04)	1	0	0-0:96.15.24.255
Security Group Event Counter Object - (G_EC_05)	1	0	0-0:96.15.25.255
Security Group Event Counter Object - (G_EC_06)	1	0	0-0:96.15.26.255
Security Group Event Counter Object - (G_EC_07)	1	0	0-0:96.15.27.255
Security Group Event Counter Object - (G_EC_08)	1	0	0-0:96.15.28.255
Security Group Event Counter Object - (G_EC_09)	1	0	0-0:96.15.29.255
Security Group Event Counter Object - (G_EC_10)	1	0	0-0:96.15.30.255

Table 39: Event Log Objects

Security Event log

min capacity: minimum of 100 entries
structure: clock.time, value
capture_period: 0 (externally triggered)
captured objects: clock.time; security event, client_SAP/server_SAP, client system title
buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)

5.13. Configuration Event Log

This event log allows keeping track of configuration event and records which attributes of methods have been accessed for re-configuration purposes.

The trigger for an event entry into the configuration event log is a SET or ACTION access to an attribute or method that causes the parameter change event to be set in the standard event log (event code = 47).

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Configuration Event Detail	1	0	0-0:94.43.150.255
Configuration Event Log	7	1	0-0:99.98.10.255

Configuration Event Detail
Identifies the object and attribute/method that was accessed for configuration purposes.
This object records

- the service_id element
  defines which action to be applied to the referenced object:
  (1) write attribute,
  (2) execute specific method
- the class_id element
- the logical_name element
- the index element
  defines (with service_id 1) which attribute of the selected object is affected; or (with service_id 2) which specific method is to be executed. The first attribute (logical_name) has index 1, the first specific method has index 1 as well.

Configuration Event Log

min capacity:	minimum of 30 entries
structure:	clock.time, value
capture_period:	0 (externally triggered)
captured objects:	clock.time; configuration event detail
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)

5.14. Load Management

The basic meter load management handling follows the same rules as defined in the IDIS package 2 specification [D]:
Please refer to the following chapter:

- 6.8 Load Management by Relay

In addition to the already specified functionality, a more flexible approach in the switching times and intervals is required. In order to be independent of the main tariff rate switching and activation, the load management functionality makes use of dedicated activity calendar and special days objects.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Load Management activity calendar	20	0	0-0:13.0.2.255
Load Management special days table	11	0	0-0:11.0.2.255
Load Management script table	9	0	0-0:10.0.103.255
Load Management relay control 1	70	0	0-1:96.3.10.255

Table 40: Calendar Objects

Load Management activity calendar
Using a dedicated activity calendar for the load management allows the configuration of the load switches independent of the tariff rate switching.

The load management activity calendar must support at least the following:
- season_profile => at least 4 seasons
- week_profile_table => at least 4 entries, exactly one per season
- day_profile_table => at least 4 entries
- day_profile => at least 5 switching times per day

Load Management special days table
Allows the definition of special days for the load management activity calendar.

Special days table must support a minimum of 200 entries in order to cover all fixed and flexible Austrian holidays (Easter Monday, Corpus Christi, Ascension, Whit Monday, …) until 2050.

Load Management script table
The disconnect script table contains the scripts which act on the Load Management object as follows:

Script identifier	Action
1	SET control_state to "Ready_for_reconnection (2)" Performs a local disconnection according to transition "**local_disconnect (g)**".
2	SET control_state to "Connected (1)" Performs a local reconnection according to transition "**local_reconnect (h)**"
3	execute method "remote_disconnect(0)" Performs a remote disconnection according to transition "**remote_disconnect (b)**" or "**remote_disconnect (c)**", depending on the control mode setting.
4	execute method "remote_reconnect(0)"_ Performs a remote reconnection according to transition "**remote_reconnect (a)**" or "**remote_reconnect (d)**", depending on the control mode setting.

Table 41: Disconnect Scripts

If the state transition is not allowed by the control mode, then the action is ignored.

Load Management relay control
The behaviour of the relay on any remote, local and manual disconnection or reconnection commands is dependent by the control_mode setting of the object.

control _mode	Disconnection				Reconnection			
	Remote		Manual	Local	Remote		Manual	Local
enum:	(b)	(c)	(f)	(g)	(a)	(d)	(e)	(h)
(0)	–	–	–	–	–	–	–	–
(1)	x	x	x	x	–	x	x	–
(2)	x	x	x	x	x	–	x	–
(3)	x	x	–	x	–	x	x	–
(4)	x	x	–	x	x	–	x	–
(5)	x	x	x	x	–	x	x	x
(6)	x	x	–	x	–	x	x	X
NOTE 3 In Mode (0) the disconnect control object is always in 'connected' state.								
NOTE 4 Local disconnection is always possible unless the corresponding trigger is inhibited.								

Figure 8: Load Management relay commands

As the Load Management relays only support 2 states, ON or OFF, only the 2 defined methods ‘remote_disconnect’ and ‘remote_connect’ should be used for controlling the relay.

The control mode shall be fixed to
⇨ **control mode = 4**

The following behaviour of the Load Management relay is specified, using the remote methods via the activity calendar or direct commands.
- Disconnection (either directly or scheduled)
  => remote_disconnect (b)
- Reconnection (either directly or scheduled)
  => remote_reconnect (a)

5.15. Display Specific Features

5.15.1. Disabling the display of Load Profile 1 and 2

As a standard feature, it is required to show the Load Profile 1 and Load Profile 2 data on the display of the meter.

Due to privacy reasons, it must be possible to deactivate this functionality.

The information of the current Load Profile display status is available remotely by reading the Load profile display control status object.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Load profile display control status	1	0	0-0:96.5.4.255
Load profile display control schedule	22	0	0-0:15.1.5.255
Load profile display control script table	9	0	0-0:10.1.109.255

Table 42: Load Profile Display Objects

Script identifier	Action
1	Activate displaying of load profile 1 and 2 on the LCD
2	Deactivate displaying of load profile 1 and 2 on the LCD
3	Activate displaying of load profile 1 on the LCD
4	Deactivate displaying of load profile 1 on the LCD
5	Activate displaying of load profile 2 on the LCD
6	Deactivate displaying of load profile 2 on the LCD

Table 43: Load Profile Display Scripts

5.15.2. Displaying consumer information data

The E-meter supports displaying of configurable consumer information messages on the E-meter display.

The following object support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Consumer Message Code - Meter Display	1	0	0-0:96.13.1.255

This object supports up to 64 bytes of printable ASCII characters (range 0x20 .. 0x 7E)
Depending on the capabilities of the meter displays, the message must be sent in a format that is supported by the individual meter

As soon as a message is sent, the meter will show the message on the meter display.
The message on the display will be removed be writing an empty array to this object, either by

- Remote configuration
- Acknowledgement by the Consumer via button press

The status remains over a power fail

5.15.3. Displaying Billing data

The meters support 2 different tariffication schemes that can be used for the customer billing.

- ⇨ 'Central' tariffication scheme
- ⇨ 'Local' tariffication scheme

In the Central tariffication scheme the tariffication is done in the central system. The customer billing is based on the total energy register values (1.8.0, 2.8.0, …) and/or the load profile.
In the Local tariffication scheme the tariffication is done locally in the meter. The customer billing is based on the rated energy registers values (1.8.1, 1.8.2, 2.8.1, 2.8.2, ….) and requires the support of the tariffication calendar and register activation configuration.
The information of which tariffication scheme is currently in use is solely based on the visualisation to the customer on the meter display. The display configuration contains only the data elements that are actually used for the customer billing.
Changing between these schemes requires a reconfiguration of the display elements.

- ⇨ Total energy register values (1.8.0, 2.8.0, …) in the display menus in case of Central Tariffication
- ⇨ Rated energy register values (1.8.1, 1.8.2, 2.8.1, 2.8.2, ….) in the display menus in case of Local Tariffication

The tariffication schemes changes don't impact the TOU activity calendar based activation of TOU tariffication scripts. Everything continues working as before.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Tariffication Scheme	1	0	0-0:94.43.142.255
General display readout (Data Scroll mode)	7	1	0-0:21.0.1.255
Alternate display readout (Standard Data mode)	7	1	0-0:21.0.2.255

Tariffication Scheme
The object defined the behaviour of the Energy Tariff Rate Indicator. There are 2 options available:

- 'central' tariffication scheme (NO tariff indicator used)
- 'local' tariffication scheme (T1 and T2 tariff indicators used)

General Display Readout (Data Scroll Mode)
This object allows the configuration of the display elements shown in the Data Scroll Mode.
The attribute 3, capture_objects, is used for the data element configuration

- ⇨ Please refer to the tab 'Display Configuration' in the Data Model [1] for all supported display data elements

The attribute 4, capture_period, defines the scroll interval in seconds (default 5s).

Alternate Display Readout (Standard Data Mode)
This object allows the configuration of the display elements shown in the Standard Data Mode. The Standard Data typically contains the identification and billing relevant data

The attribute 3, capture_objects, is used for the data element configuration
⇨ Please refer to the tab 'Display Configuration' in the Data Model [1] for all supported display data elements

The Standard Data list contains the legally relevant data elements that must be available for verifying the customer's billing data. Setting up this configuration must be in line with the MID and BEV certification requirements.
This implies for example that certain elements may not be removed from list
- LR FW identifier and signature
- F.F error register

Remark:
Displaying the historical billing data on the LCD requires a specific configuration setting. The OBIS codes typically used for the identification of historical billing data change dynamically in the F field corresponding to the billing period number. As the display list configuration requires a static definition of a capture_object, the following convention is applied for historical billing data which is in line with the OBIS specification:
- Using an OBIS code with the F field set to 101 indicates to the 'last' historical value
- Using an OBIS code with the F field set to 102 indicates to the '2nd to last' billing value

For example:

Capture object definition	***Description of the value displayed***	***OBIS Code (Display)***
{3,1-0:1.8.0.**255**,2,0},	Active energy import (+A) - current value	1.8.0
{3,1-0:1.8.0.**101**,2,0},	Active energy import (+A) - last billing value	1.8.0.(VZ)
{3,1-0:1.8.0.**102**,2,0},	Active energy import (+A) - 2nd to last billing value	1.8.0.(VZ-1)

5.15.4. Displaying Instrumentation data

The meter supports a number of instantaneous power values like voltage, current, net frequency, …… to be shown on the display.

The following object support this functionality:

Object / Attribute Name	**Class**	**Ver.**	**OBIS code**
Alternate display readout (Service Data mode)	7	1	0-0:21.0.3.255

Alternate Display Readout (Service Data Mode)
This object allows the configuration of the display elements shown in the Service Data Mode. The Service Data typically contains the instantaneous values.
The attribute 3, capture_objects, is used for the data element configuration
⇨ Please refer to the tab 'Display Configuration' in the Data Model [1] for all supported display data elements

5.16. Certification Support

Status of the Certification mode must be accessible on the meter via the maintenance interface WZ.
Start and end time of the calibration mode must be stored in the logbook of the E-meter.

Activation of the certification mode:
- Setting the attribute 2 value to TRUE

Deactivation of the certification mode:
- Setting the attribute 2 value to FALSE
- Automatically 12h after the activation
- Loosing date and time over power fail

A power failure itself does not deactivate the Certification mode.
The 12h period after activation runs independent of the meter RTC. Clock changes don't impact the activation period.

The current status of the certification mode is reflected in the Attribute 2 value when reading the attribute.

Actions:
- Activating and deactivating the certification mode
- Define the Energy register resolution on the display
- Define the Demand register resolution on the display
- Reconfiguration of the certification LED source (+A, -A, R1, R2, R3, R4, +VA, -VA,..)
- Reconfiguration of the certification LED pulse rate

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Certification mode	1	0	

Table 44: Certification Objects

Certification mode
Detailed description:

Attribute description

Attribute 2: value

value ::= structure
{
certification_status:
certification_energy_res:
certification_demand_res:

certification_LED_source:
certification_LED_rate:
}

certification_status
Activation or deactivation of the certification mode when writing this value.
Reading this data shows the current activation status of the certification mode.
Data type: Boolean
Authorised values:
0: FALSE
1: TRUE
Default value: 0 (FALSE)

certification_energy_res:
This item defines the resolution of the energy registers on the Display
Data type: Enum
Authorised values:
0: no change to the current settings
1: no decimals = x kWh
2: 1 decimal = x.1 kWh
3: 2 decimal = x.12 kWh
4: 3 decimal = x.123 kWh
5: 4 decimal = x.1234 kWh
Default value: 0 (no change to the current settings)

certification_demand_res:
This item defines the resolution of the demand registers on the Display
Data type: Enum
Authorised values:
0: no change to the current resolution
1: no decimals = x kW
2: 1 decimal = x.1 kW
3: 2 decimal = x.12 kW
4: 3 decimal = x.123 kW
5: 4 decimal = x.1234 kW
Default value: 0 (no change to the current settings)

certification_LED_source:
This item allows switching the energy source of the certification led. It's possible to map up different energy types to this LED in order to allow combinations (like +/-A or +R).
The configuration item is defined as a bit field:
Bit0: import active
Bit1: export active
Bit2: reactive Q1
Bit3: reactive Q2
Bit4: reactive Q3
Bit5: reactive Q4
Bit6: apparent import
Bit7: apparent export

Setting this value to 0 leaves the LED source setting untouched
Data type: unsigned8
Authorised values: 0 to 0xFF
Default value: 0 (no change to the current settings)

certification_LED_rate:
This items defines the pulse rate constant of the certification LED expressed in PULSES/kUNITh.
The value of 0 leaves the LED pulse rate setting untouched
Data type: unsigned16
Authorised values:
0, 1000, 2000, 4000, 8000, 16000
Default value: 0 (no change to the current settings)

5.17. Certification Protected Event Log

This event log allows recording of events that impact the functionality which is subject to legal metrological certification.
The event log itself is part of the functionality under legal control.

The implementation is under the responsibility of the manufacturer and must ensure compliance to the appropriate legal requirements.

The recorded information contains the currently active FW version and the client system title for the identification of who initiated the certification relevant change.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Certification Event Id	1	0	0-0:96.11.98.255
Certification Protected Event Log	7	1	0-0:99.98.98.255

Certification Event Id
The following events are considered as legally relevant events.
- FW download status in case of updating the legally relevant (LR) part of the Meter
  - Event code 17 => LR - Firmware ready for activation
  - Event code 18 => LR - Firmware activated
  - Event code 51 => LR - FW verification/activation failed

Certification Protected Event Log

min capacity:	minimum of 100 entries
structure:	clock.time, status, active LR FW version
capture_period:	0 (externally triggered)
captured objects:	clock.time; certification event id; client system title of activation initiator; active firmware identifier

buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)

5.18. Outputs

5.18.1. Control Outputs

Demand Measurement Period Control Output
The demand measurement period control output acts as a normally closed switch.
The switch opens with the start of a new demand measurement period for 9 seconds (1% of the demand measurement period; 15min=900sec => 9sec). The switch remains closed of the remaining time of the demand measurement period.

⇨ Optionally, the optical interface can be used to send this signal instead of an actual control output.
The send-LED of the optical interface switches ON with the start of a new demand measurement period for 9 seconds .
In case a communication session is active on the optical interface, the demand measurement period signal must be suppressed to not interfere with the communication!!

5.18.2. Pulse Outputs

Energy proportional pulse output
The energy proportional pulse output acts as a normal open switch and closes for the output of an energy pulse.
The signal form at this output is a rectangular pulse with a pulse length between 80 – 100ms.

5.19. Communication logs

The meter supports 2 log for recoding communication related events

1. Communication Event log – records communicated communication specific events and errors
2. Communication Session log – tracks the communication session via the local and remote interfaces including the client information

5.19.1. Communication Event log

The event codes used for the Communication Event Log can be found in [1].

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Event Object - Communication Event Log	1	0	0-0:96.11.5.255

Communication Event Log	7	1	0-0:99.98.5.255

Communication Event Log
min capacity: minimum of 100 entries
structure: clock.time, value
capture_period: 0 (externally triggered)
captured objects: clock.time; communication event code
buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)

5.19.2. Communication Session log

The event codes used for the Session Event Log can be found in [1].
This log registers the following additional information with each event entry:

- client SAP/server SAP => out of the Current Association LN
- client system title => out of the Current Security Setup

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Event Object - Communication Session Log	1	0	0-0:96.11.6.255
Communication Session Log	7	1	0-0:99.98.6.255

Communication Session Log
min capacity: minimum of 100 entries
structure: clock.time, value
capture_period: 0 (externally triggered)
captured objects: clock.time; event, client_SAP/server_SAP, client system title
buffer encoding: normal: clock with every entry
selective access: by range and by entry
sorted method: unsorted (FIFO)

6. Submeters

6.1. M-Bus Identification Numbers

The Submeters require a number of identification items.

The implementation of this identification numbers follows the IDIS package 2 specification [D]:
Please refer to the following chapters:
- 5.3.1.1 Uniqueness of M-bus device identification

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
M-Bus Device ID 1 channel x	1	0	0-x:96.1.0.255
M-Bus Device ID 2 channel x	1	0	0-x:96.1.1.255

Table 45: M-Bus objects

M-Bus Device ID 1
This object contains the ASCII encoded M-Bus Fabrication Number of the M-Bus device

M-Bus Device ID 2
This object contains the ASCII encoded Application Layer Address of the M-Bus device

6.2. M-Bus Data

The data elements form the submeters need conversion into the corresponding COSEM objects

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
M-Bus Value channel x, instance 1	4	0	0-x:24.2.1.255
M-Bus Value channel x, instance 2	4	0	0-x:24.2.2.255
M-Bus Value channel x, instance 3	4	0	0-x:24.2.3.255
M-Bus Value channel x, instance 4	4	0	0-x:24.2.4.255

Table 46: M-Bus objects

Each instance of an M-bus value object is associated to the corresponding configuration in the capture_definition of the M-Bus client object

M-Bus Value

The e-meter automatically configures the scaler_unit according to the corresponding information contained in VIF.

6.3. M-Bus Load Profile

The implementation of the M-Bus profile follows the IDIS package 2 specification [D]:
Please refer to the following chapters:
- 6.3.2.1 M-Bus Master Load profile for channel 1..4

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Profile status - M-Bus Master Load profile channel x	1	0	0-x:96.10.3.255
M-Bus Master Load profile channel x	7	1	0-x:24.3.0.255

The following details apply for the profiles:
M-bus Master Load profile

capacity:	10 days with hourly entries, 6 captured objects
structure:	clock.time, profile_status, M-Bus value objects
capture_period:	default 60 minutes (3600 seconds), allowed range 1,5,10,15,60 min or daily
captured objects:	clock.time, profile_status, M-Bus value instance 1-4
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)
profile_status:	please see 5.7.3 **Profile Status**

6.4. M-Bus Disconnection

The implementation of the M-Bus disconnection functionality follows the IDIS package 2 specification [D]:
Please refer to the following chapters:
- 6.5 Meter Disconnection and Reconnection
- 6.5.2 M-Bus Disconnector script table

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
M-Bus Master Disconnect control object channel x	70	0	0-x:24.4.0.255
M-Bus Disconnect control scheduler	22	0	0-1:15.0.1.255
M-Bus Disconnector script table	9	0	0-1:10.0.106.255
Event Objects - M-Bus Master Control logs channel x	1	0	0-x:96.11.4.255
M-Bus Master Control log channel x	7	1	0-x:24.5.0.255

There is a dedicated Control log associated to each Submeter channel for logging the activities on the disconnector functionality.

M-Bus Master Control log

min capacity:	minimum of 10 entries
structure:	clock.time, value
capture_period:	0 (externally triggered)
captured objects:	clock.time; disconnect event
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)

6.5. M-Bus Event Log

The implementation of the M-Bus event logging functionality follows the IDIS package 2 specification [D]:
Please refer to the following chapters:
- 7.2.4 Extension M objects

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Event Object - M-Bus Event Log	1	0	0-0:96.11.3.255
M-Bus Event Log	7	1	0-0:99.98.3.255

M-Bus Event Log

min capacity:	minimum of 100 entries
structure:	clock.time, value
capture_period:	0 (externally triggered)
captured objects:	clock.time; M-Bus event
buffer encoding:	normal: clock with every entry
selective access:	by range and by entry
sorted method:	unsorted (FIFO)

6.6. M-Bus Clock Synchronisation

The M-Bus clock is synchronized by the E-meter in case the E-meter clock is synchronized, on a regular interval of 24h or by remote invocation of the synchronize_clock method in the corresponding M-Bus client channel setup object.

7. Remote Firmware Upgrade

The remote FW update follows the definition of the IDIS package 2 specification [D].
Please refer to the following chapter of the IDIS package2 specifications [D]:
- 6.9 Firmware Update

This Companion Standard supports only a single instance of the Image Transfer.
In case the meter supports the download of multiple parts of the firmware, it is the manufacturer's responsibility to ensure the proper identification and activation of these images.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Image transfer	18	0	0-0:44.0.0.255
Image transfer activation scheduler	22	0	0-0:15.0.2.255
Predefined Scripts - Image activation	9	0	0-0:10.0.107.255
Active firmware identifier	1	0	1-0:0.2.0.255
Active firmware signature	1	0	1-0:0.2.8.255
Active firmware identifier 1	1	0	1-1:0.2.0.255
Active firmware signature 1	1	0	1-1:0.2.8.255
Active firmware identifier 2	1	0	1-2:0.2.0.255
Active firmware signature 2	1	0	1-2:0.2.8.255

Table 47: FW Upgrade Objects

The active FW identifiers and the version signatures of all individual parts of the firmware are available for readout using the corresponding objects.
The B field of the OBIS codes gives a clear identification of the individual firmware parts
- The metrological relevant part of the FW uses B=0.
- The main application part (non-metrological relevant) of the FW uses B=1
- Other parts (e.g. modem firmware) must use a B field value in the range of B=2..9.

Every image for download to the E-meter requires a digital signature.
This Companion Standard specifies the usage of the following algorithm
=> ECDSA P-256.
The activation of a new FW image requires the successful validation of the digital signature.
The activation must be rejected in case the verification fails.

As part of the image verification, the E-meter checks the integrity of the received image and verifies the correctness of the image_identifier and image_size information (manufacturer specific data provided during the image transfer initialisation)

Reading the attribute 7 of object "Image transfer" allows the retrieval of the digital signature after the image verification and before the image activation ("image_to_activate_signature"). After the image activation, the digital signature can be retrieved from the object "Active firmware version signature".

Image Block size

This data contains the size of image block to use for image transfer, expressed in octets.
This data cannot be written if an image transfer has been initiated (in such case, the writing is rejected).
The appropriate value of this attribute is calculated by the system, considering the quality of the communication, and is then written in the meter.
When initiating an image transfer, the client will read the value of image_block_size, and split the firmware in the required size.
- supported block size: 64 to 1024

Image Transfer enabled
This attribute allows the activation of the FW image transfer when set to TRUE.
Setting this value to FALSE, will inhibit all access to all methods and a currently running transfer will be aborted. The image transfer status returns to 'Image transfer not initiated'.

The FW Update client can be used for the image distribution via broadcast.

8. Event Handling

The meter generates a number of Events for additional information concerning the status of the meter or configuration.

Certain conditions can triggered the event and initiate the logging into the event log. The root cause for the individual trigger depends on the nature of the events. As long as the root cause is still active, the event will not be re-triggered.

These logs register any event with the corresponding timestamp.
The event codes used in the meters can be found in [1].

The list is based on the IDIS package 2 [D] definition.

Please refer to the following chapter of the IDIS package2 specifications [D]:
- 10. Appendix: Event Codes

Not all IDIS event codes are applicable for this specification, so the implementation of these events is not considered as mandatory.
In addition, this implementation requires additional events to be registered that are currently not part of the IDIS definitions. This Companion Standard uses some of the reserved codes for the additionally required event codes.

The supported event logs are the following:

⇨ Standard Event Log
  See 5.10 Standard Event Log
⇨ Fraud Detection Log
  See 5.11 Fraud Detection Event Log
⇨ Power Failure Log
  See 5.9.3 **Power fail detection**
⇨ Power Quality Log
  See 5.9.2 **Voltage Cut, Sag and Swell detection**
⇨ Disconnect Control Log
  See 5.8 Disconnector and Limiter
⇨ Specific Security Event Log
  See 5.12 Specific Security Event Log and Event Counter
⇨ Configuration Event Log
  See 5.13Configuration Event Log
⇨ Tariff Activation Event log
  See 5.5 Calendar and Tariff Handling
⇨ M-Bus Event Log
  See 6.5 M-Bus Event Log
⇨ M-Bus Disconnect Control Logs
  See 6.4 M-Bus Disconnection

Retrieving the data of the event logs is possible by reading the entire attribute ‘buffer’, or using the selective access by range (access selector 1) or by entry (access selector 2). The support of access by range (including support for selected_ values) and access by entry is mandatory.

For further clarification to the selective access on the Event Logs, please refer to the following chapter of the IDIS package2 specifications [D]:

- 7.7 Reading profiles with parameterized access “from”-“to”

9. Error and Alarm Handling

Certain events can trigger setting flags in the error or alarm register.
The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Error Register	1	0	0-0:97.97.0.255
Alarm Register 1	1	0	0-0:97.98.0.255
Alarm Filter 1	1	0	0-0:97.98.10.255
Alarm Descriptor 1	1	0	0-0:97.98.20.255
Alarm Monitor 1	1	0	0-0:16.1.0.255
Alarm Register 2	1	0	0-0:97.98.1.255
Alarm Filter 2	1	0	0-0:97.98.11.255
Alarm Descriptor 2	1	0	0-0:97.98.21.255
Alarm Monitor 2	1	0	0-0:16.1.1.255
Alarm Register 3	1	0	0-0:97.98.2.255
Alarm Filter 3	1	0	0-0:97.98.12.255
Alarm Descriptor 3	1	0	0-0:97.98.22.255
Alarm Monitor 3	1	0	0-0:16.1.2.255
Fatal Error Register	1	0	0-0:97.97.128.255

Table 48: Error and Alarm Objects

9.1. Error and Alarm Register

The error and alarm register show the status of selected events.
Setting an error or alarm flag

- Set when an event is triggered

Resetting an error or alarm flag

- Reset if the root cause of the event is not active any more
- Reset by clearing the error or alarm register by external command

Some flags can only be externally reset and remain active until then independently of the root cause:

- bit 13 (Fraud attempt)

The IDIS package2 specifications [D] defines in detail the relation between alarm register and alarm filter:
Please refer to the following chapters:
- 7.3.2 Alarms

Alarms can trigger an automated notification to the client
Depending on the use cases or the capabilities of the client, it might not be wanted to receive all possible alarms.
The Alarm Filter allows masking out alarm flags that should not raise a notification to the client

The handling of the client notification due to alarms is described in the IDIS package2 specifications [D]:
Please refer to the following chapters:
- 7.3.2.1 Alarming Process

As long as the HES has not acknowledged the reception of the alarm message by clearing the Alarm Descriptor, the meter must keep triggering the automated notification in a 10-minute interval.

The error and alarm register flag assignment can be found in [1].
The meaning of the Error Register bits is the same as for the Alarm Register 1.

9.2. Fatal Error Register

Any error in this register is considered as legally relevant and causes the cancellation of its metrological certification.
The Fatal Error Register (0-0:97.97.128.255) is serving the legal certification requirements and its usage is independent of the generic Error Register (0-0:97.97.0.255).

The bit-allocation in this Fatal Error Register is manufacturer specific.

Setting a fatal error flag
- Set when an error in the metrological part is detected

Resetting a fatal error flag
- Reset is not possible without breaking the metrological seal (manufacturer specific)

Any legally relevant error recorded in this Fatal Error Register will be flagged as a ‘Measurement system error’ in the generic Error Register and Alarm Register 1 in order to trigger an alarm message.

10. PUSH operations

The data notification service (PUSH) allows sending data to the HES, initiated by the meter itself. There are several occasions on which data may be 'pushed', i.e. sent to the HES without being explicitly requested, e.g.

- at scheduled times or intervals;
- if an alarm threshold is exceeded;
- triggered by the HES
- triggered by an event like establishing a connectivity to the HES

The meter basically follows the definitions of the IDIS package 2 specification in relation to the Data Push operation.
Please refer to the following chapters in the IDIS package 2 specification [D]:

⇨ 7.8 PUSH operation

The optional IDIS 2 push on power down is not considered for this application.

In addition, the data notification service (PUSH) also allows sending data to the consumer using the H1 – Consumer Interface

The following object supports this functionality for all possible triggers:

Object / Attribute Name	Class	Ver.	OBIS code
Push script table	9	0	0-0:10.0.108.255

The Push script table contains references for all defined push setup objects:

Script_identifier	logical_name	description
1	0-1:25.9.0.255	Push Setup – Interval 1
2	0-2:25.9.0.255	Push Setup – Interval 2
3	0-3:25.9.0.255	Push Setup – Interval 3
4	0-4:25.9.0.255	Push Setup – On Alarm
5	0-0:25.9.0.255	Push Setup – On Connectivity
6	0-7:25.9.0.255	Push Setup – On Installation
7	0-5:25.9.0.255	Push Setup – On Power Down
8	0-6:25.9.0.255	Push Setup – Consumer Push

10.1. Meter Reading

PUSH operation offers the HES the possibility to trigger a Data-Notification service, either periodically or on demand for retrieving metering data

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Push setup – Interval_1	40	0	0-1:25.9.0.255
Push setup – Interval_2	40	0	0-2:25.9.0.255
Push setup – Interval_3	40	0	0-3:25.9.0.255
Push action scheduler – Interval_1	22	0	0-1:15.0.4.255
Push action scheduler – Interval_2	22	0	0-2:15.0.4.255
Push action scheduler – Interval_3	22	0	0-3:15.0.4.255

Push setup – Interval 1,2,3
This object defines the data elements for periodically pushing to the HES. 3 independent objects allow pushing different data elements, depending on individual intervals or based on other triggers.

⇨ The data push is triggered by the associated push action schedulers

A minimum of 20 capture object definitions for the push_object_list must be supported.
The push_object_list may contain:
- total registers
- rated registers
- profiles
- event logs
- instantaneous values
- ….

The push process takes place within the application context of the ‘Data Readout Client’. The object ‘Security setup - Data Readout Client’ determines the security context.

Push action scheduler – Interval 1,2,3
This object allows the configuration of individual intervals or timestamps for the corresponding Push setup objects.

10.2. Meter Alarm

The handling of a data push due to alarms is described in the IDIS package2 specifications [D]:
Please refer to the following chapters:
- 7.3.2.1 Alarming Process

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Push setup - On Alarm, trigger Alarm monitor 1, 2 or 3	40	0	0-4:25.9.0.255

Push setup - On Alarm

This object defines the data elements for pushing to PAN coordinator in case of an alarm detection.

⇨ The data push is triggered by the alarm monitors 1, 2 or 3

A minimum of 20 capture object definitions for the push_object_list must be supported.
The following data must be pushed to the customer by default:
- Clock, attribute 2 - time
- Alarm Descriptor 1
- Alarm Descriptor 2
- Alarm Descriptor 3

The push process takes place within the application context of the ‘Data Readout Client’. The object ‘Security setup - Data Readout Client’ determines the security context.

10.3. Meter Installation

The handling of a data push at meter installation is described in the IDIS package2 specifications [D]:
Please refer to the following chapters:
- 6.1.3 Meter Registration using Data-Notification

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Push setup - On Installation, trigger	40	0	0-7:25.9.0.255

Push setup - On Installation
This object defines the data elements for pushing to the HES during the installation phase. Logical registration at HES level is achieved by the valid serial number of the meter provided by the Data-Notification service.

⇨ The data push is triggered manually during the installation process by invoking the push method of the push setup object (this action causes the ‘commissioning event‘ being set!)

A minimum of 20 capture object definitions for the push_object_list must be supported.
The following data must be pushed to the customer by default:
- Clock, attribute 2 - time
- Device ID 1, manufacturing number - serial number
- IP address (IPv6 for PLC; IPv4 and IPv6 for P2P)

The push process takes place within the application context of the ‘Data Readout Client’. The object ‘Security setup - Data Readout Client’ determines the security context.

10.4. Meter Connectivity

The Push on Connectivity is triggered each time a new network connection is established.

The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Push setup – On Connectivity	40	0	0-0:25.9.0.255

Push setup - On Connectivity
This object defines the data elements for pushing to the HES in case of establishing a network connectivity.

⇨ The data push is triggered every time a connection to the IP network is established.

A minimum of 20 capture object definitions for the push_object_list must be supported.
The following data must be pushed to the customer by default:
- Clock, attribute 2 - time
- Device ID 1, manufacturing number - serial number
- IP address (IPv6 for PLC; IPv4 and IPv6 for P2P)

The push process takes place within the application context of the 'Data Readout Client'. The object 'Security setup - Data Readout Client' determines the security context.

10.5. CIP – Consumer Information Push

The E-meter supports a local interface for consumer information (H1)

The functionality and security definition is following the IDIS specification [D].

Please refer to the following chapter of the IDIS package 2 specifications [D]:
- 6.11.3 Security on the Consumer Information Interface

and
- 6.11.4 CIP System Title and Error Handling

Chapter 2.2 specifies the physical parameters and the data transport layer of the communication protocol.
The following objects support this functionality:

Object / Attribute Name	Class	Ver.	OBIS code
Push action scheduler - Consumer Information	22	0	0-4:15.0.4.255
Push setup - Consumer Information	40	0	0-6:25.9.0.255

Table 49: CIP Objects

Chapter 2.2 specifies the physical parameters and the data transport layer of the communication protocol.

Push action scheduler - Consumer Information

The action scheduler allows the configuration of the interval for the data push to the customer interface. This object allows up to 60 execution times in order to allow for an interval of down to 1 second.
An empty array represents the deactivation of this interface.

Push setup - Consumer Information
This object defines the data elements for pushing to the customer interface. It's specified as a configurable list with up to 20 possible entries.
The following data must be pushed to the customer by default:
- Clock, attribute 1 – OBIS code
- Clock, attribute 2 – time
- Device ID 1 manufacturing number, attribute 0 – OBIS code, serial number
- COSEM logical device name, attribute 0 – OBIS code, logical device number
- Instantaneous voltage L1, attribute 0 – OBIS code, value, scalar and unit
- Instantaneous voltage L2*, attribute 0 – OBIS code, value, scalar and unit
- Instantaneous voltage L3*, attribute 0 – OBIS code, value, scalar and unit
- Instantaneous current L1, attribute 0 – OBIS code, value, scalar and unit
- Instantaneous current L2*, attribute 0 – OBIS code, value, scalar and unit
- Instantaneous current L3*, attribute 0 – OBIS code, value, scalar and unit
- Instantaneous active import power (+P), attribute 0 – OBIS code, value, scalar and unit
- Instantaneous active export power (-P), attribute 0 – OBIS code, value, scalar and unit
- Active energy import (+A), attribute 0 – OBIS code, value, scalar and unit
- Active energy export (-A), attribute 0 – OBIS code, value, scalar and unit
- Reactive energy import (+R), attribute 0 – OBIS code, value, scalar and unit
- Reactive energy export (-R), attribute 0 – OBIS code, value, scalar and unit

(*) Only required for Poly Phase meters (PP)

The setup object allows as well the configuration of the destination and the sending method.
For the transmission via wired M-Bus, the following setting apply:
- transport_service: 201
  This number is manufacturer specific and here used for wired M-Bus transport service
- destination: 0-2:24.6.0.255:FF:00;
  - Contains the logical_name of the M-Bus master port setup - Consumer Information Interface, octet string (6)
  - The Link Layer Address (LLA), unsigned8
  - Transport Layer Address (CI TL), unsigned8

The individual fields are separated by (':'). The separators are ASCII coded
- Message: 0
  A-XDR encoded xDLMS APDU,

10.6. send_destination_and_method configuration

For LAN interface, the following settings are defined:
Send_destination_and_method configuration:
- transport_service: 1
  This number refers to UDP

- destination: [x:x:x:x:x:x:x:x]:y
  Contains the destination IPv6 address and port number for the data delivery
  Please refer to the following chapter in the IDIS package2 specifications [D]:
  - 11.1 Send_destination_and_method (Push Setup Class, IC 40)
- Message: 0
  A-XDR encoded xDLMS APDU,

For WAN interface, the following settings are defined:

Send_destination_and_method configuration:

- transport_service: 0
  This number refers to TCP
- destination: x.x.x.x:y
  Contains the destination IPv4 address and port number for the data delivery
  Please refer to the following chapter in the IDIS package2 specifications [D]:
  - 11.1 Send_destination_and_method (Push Setup Class, IC 40)
- Message: 0
  A-XDR encoded xDLMS APDU,

10.7. number_of_retries configuration

Defines the maximum number of retries in the case of unsuccessful or skipped push attempts. After a successful push operation, no further push attempts are made until the push operation is triggered again.
A push is treated as successful if a lower layer transmission confirmation has been received.

Using TCP:

- push is considered successful in case the ACK on TCP level has been received

Using UDP:

- push is never considered successful as UDP does not provide a feedback. The device keeps retrying the push as configured.

11. Appendix 1: Frame Counter Readout

11.1. Introduction

In a dynamic network such as G3-PLC, where meters can “hop” from one PAN coordinator to another, managing the frame counter is problematic. As PAN coordinators do not (and in most cases, can not) communicate with each other, this would have to be managed by the head end side. This involves a complex synchronization scheme that leaves quite a lot of room for errors (which would then have to be solved manually by an operator), which should be avoided if possible.

Reading out the frame counter using the public client without any means to authenticate the response leaves room for MITM attacks where the DLMS client can be tricked into reusing IVs when using AES/GCM, and this has the risk of leaking key material.

This proposal explores a way to allow readout of the frame counter for particular clients, but includes an authentication scheme so the DLMS client can be sure the response originates from the meter it was requested from, and also guaranteeing that the returned frame counter has also not been tampered with, allowing it to be used safely.

11.2. Principle

When requesting the frame counter for a particular client for a particular meter, the DLMS client will invoke a method on a vendor-defined IC (get_frame_counter), passing in a 64 byte randomly generated challenge, which shall be generated by a FIPS 140-2 or AIS 31 compliant RNG.

At this point, the meter will first generate a response to the challenge by performing a **HMAC-SHA256(m, K)**, where:

- **m** is defined as **SysT-S || SysT-C || Challenge || FC** where:
  - *SysT-S*: The system title of the destination of the request (the recipient, or server) : 8 bytes.
  - *SysT-C*: The system title of the source of the request (the originator, or client) : 8 bytes.
  - *Challenge*: The random challenge received in the request : 64 bytes.
  - *FC*: The frame counter to be returned : 4 bytes.
- **K** is the authentication key associated to the requested client.

It will then return a struct { challenge-response, frame counter }, where:

- challenge_response is the result of the HMAC_SHA256 calculation.
- frame_counter is the requested frame counter.

The DLMS client, upon reception of the response, can then (provided the AK of the requested client is available to it) validate the challenge-response indeed originates from the meter, and additionally that the returned frame counter has not been tampered with (by performing the same calculation and checking whether the result is the same), and if so, does store the frame

counter for use. Note that the DLMS client only performs this once for every meter that joins the network (and not for every session).

11.3. Requirements

- Availability of the HMAC-SHA256 algorithm.
- The AARQ request used when opening the association for the public client shall carry the client system title (SysT-C) in the calling-ap-title field.
- The AARE response from the meter shall correspondingly carry the server system title (SysT-S) in the responding-ap-title field.

It is mandatory that the client and server system titles are exchanged during application association establishment for the public client when retrieving the frame counters. The client system title shall be included in the AARQ by means of the calling-ap-title field, the server system title in the responding-ap-title of the AARE.

11.1. Implementation

The meter will define a vendor-specific IC (class ID 12544 (0x3100)) (one for every client for which the frame counter needs to be exposed) that has the following structure:

Frame Counter Provider IC	**Class ID 12544 (0x3100), version 0**
Attributes	**Data type**
1. logical_name	octet-string
Methods	**m/o**
1. get_frame_counter(challenge)	m

Method description	

get_frame_counter(challenge)	Request the frame counter for the client that this instance is tied to.
	
	The challenge is a 64-byte random (octet-string).
	
	`challenge : octet-string`
	
	Upon invocation of this method, the meter will generate a response to the challenge by performing an HMAC-SHA256(K, m) where K = the AK of the corresponding client and m is the concatenation of Server System Title, Client System Title, received challenge and the frame counter (SysT-S || SysT-C || Challenge || FC) to be returned. It will then generate a response:
	
	`response : struct {`
	`  challenge_response : octet-string,`
	`  frame_counter : unsigned32`
	`}`

12. Appendix 2: Certificate Examples

The following are examples of certificates:

Trust Anchor: – Root Certificate

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1527056696 (0x5b050938)
    Signature Algorithm: ecdsa-with-SHA256
        Issuer:
            commonName                = SM-Test-Root-CA
        Validity
            Not Before: Apr 30 21:00:00 2018 GMT
            Not After : Apr 30 21:00:00 2028 GMT
        Subject:
            commonName                = SM-Test-Root-CA
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:81:57:ac:a4:2b:f6:c3:e0:ba:b0:8d:04:fc:e1:
                    99:d0:6a:c9:3f:be:62:ff:0b:54:32:c4:2e:b7:5d:
                    b6:ef:43:14:95:3f:43:82:b3:30:53:83:dd:d8:65:
                    4e:08:46:c6:54:0a:82:ec:b4:e7:bb:e3:fa:39:d4:
                    18:99:40:5f:78
                ASN1 OID: prime256v1
        X509v3 extensions:
            X509v3 Basic Constraints: critical
                CA:TRUE, pathlen:0
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Subject Key Identifier:
                20:EF:DB:F0:09:F3:5A:B9:DD:9B:8C:EB:15:24:E7:1F:B2:DC:86:1B
    Signature Algorithm: ecdsa-with-SHA256
         30:45:02:21:00:93:47:57:01:aa:72:dc:b8:4b:5e:fe:2b:71:
         6e:09:08:3f:68:b7:5a:49:fc:09:7b:cc:cf:97:c2:ca:cf:be:
         4f:02:20:61:df:97:79:27:65:3a:b8:b9:9e:c8:13:a5:13:b9:
         46:29:3f:51:d3:6b:e6:83:23:bc:ef:a2:0c:5b:be:4f:ac
-----BEGIN CERTIFICATE-----
MIIBazCCARGgAwIBAgIEWwUJODAKBggqhkjOPQQDAjAaMRgwFgYDVQQDDA9TTS1U
ZXN0LVJvb3QtQ0EwHhcNMTgwNDMwMjEwMDAwWhcNMjgwNDMwMjEwMDAwWjAaMRgw
FgYDVQQDDA9TTS1UZXN0LVJvb3QtQ0EwWTATBgcqhkjOPQIBBggqhkjOPQMBBwNC
AASBV6ykK/bD4LqwjQT84ZnQask/vmL/C1QyxC63XbbvQxSVP0OCszBTg93YZU4I
RsZUCoLstOe74/o51BiZQF94o0UwQzASBgNVHRMBAf8ECDAGAQH/AgEAMA4GA1Ud
DwEB/wQEAwIBBjAdBgNVHQ4EFgQUIO/b8AnzWrndm4zrFSTnH7LchhswCgYIKoZI
zj0EAwIDSAAwRQIhAJNHVwGqcty4S17+K3FuCQg/aLdaSfwJe8zPl8LKz75PAiBh
35d5J2U6uLmeyBOlE7lGKT9R02vmgyO876IMW75PrA==
-----END CERTIFICATE-----
```

For Certification Authorities: – Certification Authority Certificate

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1527057608 (0x5b050cc8)
    Signature Algorithm: ecdsa-with-SHA256
        Issuer:
            commonName                = SM-Test-Root-CA
        Validity
            Not Before: Apr 30 21:00:00 2018 GMT
            Not After : Apr 30 21:00:00 2028 GMT
        Subject:
            commonName                = HES-CA
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:81:8d:69:4f:14:4a:fb:d7:e1:aa:2a:9c:a6:e4:
                    61:f3:2f:b6:15:69:55:55:51:fd:c7:7e:5a:f3:af:
                    da:5f:5e:ba:b1:94:be:8c:aa:8c:09:24:08:ee:97:
                    39:c7:26:82:f7:b0:6f:39:e4:6a:4c:1f:cc:2f:05:
                    b3:39:8c:04:e4
                ASN1 OID: prime256v1
        X509v3 extensions:
            X509v3 Basic Constraints: critical
                CA:TRUE, pathlen:0
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Certificate Policies:
                Policy: X509v3 Any Policy
                  CPS: https://confluence.salzburg-ag.at
            X509v3 Authority Key Identifier:
                keyid:20:EF:DB:F0:09:F3:5A:B9:DD:9B:8C:EB:15:24:E7:1F:B2:DC:86:1B
                DirName:/CN=SM-Test-Root-CA
                serial:5B:05:09:38
            X509v3 Subject Key Identifier:
                1F:3F:CB:93:BB:85:14:4A:7C:1F:DD:48:68:0E:F0:A6:1C:F8:1C:57
    Signature Algorithm: ecdsa-with-SHA256
         30:45:02:21:00:84:60:20:98:fb:25:98:38:c8:dd:94:c3:98:
         fa:f7:2e:8c:35:11:82:41:b8:0c:ca:a4:3f:89:d7:a5:cd:1d:
         9e:02:20:31:15:cc:7b:6a:e2:32:aa:19:b2:01:92:83:be:4d:
         b4:22:3d:37:85:2a:53:af:bb:d6:1b:0d:46:c1:1f:e4:48
-----BEGIN CERTIFICATE-----
MIIB7zCCAZWgAwIBAgIEWwUMyDAKBggqhkjOPQQDAjAaMRgwFgYDVQQDDA9TTS1U
ZXN0LVJvb3QtQ0EwHhcNMTgwNDMwMjEwMDAwWhcNMjgwNDMwMjEwMDAwWjARMQ8w
DQYDVQQDDAZIRVMtQ0EwWTATBgcqhkjOPQIBBggqhkjOPQMBBwNCAASBjWlPFEr7
1+GqKpym5GHzL7YVaVVVUf3Hflrzr9pfXrqxlL6MqowJJAjulznHJoL3sG855GpM
H8wvBbM5jATko4HRMIHOMBIGA1UdEwEB/wQIMAYBAf8CAQAwDgYDVR0PAQH/BAQD
AgEGMEIGA1UdIAQ7MDkwNwYEVR0gADAvMC0GCCsGAQUFBwIBFiFodHRwczovL2Nv
bmZsdWVuY2Uuc2FsemJ1cmctYWcuYXQwRQYDVR0jBD4wPIAUIO/b8AnzWrndm4zr
FSTnH7LchhuhHqQcMBoxGDAWBgNVBAMMD1NNLVRlc3QtUm9vdC1DQYIEWwUJODAd
BgNVHQ4EFgQUHz/Lk7uFFEp8H91IaA7wphz4HFcwCgYIKoZIzj0EAwIDSAAwRQIh
AIRgIJj7JZg4yN2Uw5j69y6MNRGCQbgMyqQ/idelzR2eAiAxFcx7auIyqhmyAZKD
vk20Ij03hSpTr7vWGw1GwR/kSA==
-----END CERTIFICATE-----
```

For the client – Digital Signature Certificate

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1527057712 (0x5b050d30)
    Signature Algorithm: ecdsa-with-SHA256
        Issuer:
            commonName               = HES-CA
        Validity
            Not Before: May 19 21:00:00 2018 GMT
            Not After : May 19 21:00:00 2020 GMT
        Subject:
            commonName               = DLMSTester-signature-Management
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:c8:9e:ab:b2:50:51:00:f8:46:06:c2:a7:a3:25:
                    f4:96:8b:65:4c:f1:fc:b2:33:79:4d:15:1b:0f:76:
                    49:51:e7:1c:c8:92:4d:57:0c:85:0d:2f:c9:09:0b:
                    ab:f1:fd:b5:e8:2d:bf:20:cb:55:fd:ac:2a:49:e1:
                    fe:ae:3d:1a:33
                ASN1 OID: prime256v1
        X509v3 extensions:
            X509v3 Key Usage: critical
                Digital Signature
            X509v3 Authority Key Identifier:
                keyid:1F:3F:CB:93:BB:85:14:4A:7C:1F:DD:48:68:0E:F0:A6:1C:F8:1C:57
                DirName:/CN=HES-CA
                serial:5B:05:0C:C8
    Signature Algorithm: ecdsa-with-SHA256
         30:46:02:21:00:80:ab:ba:bc:95:be:6e:71:b3:03:29:11:fc:
         01:e4:6c:8e:44:e1:13:06:29:7a:51:fa:ee:9d:a7:b1:66:d8:
         ba:02:21:00:f9:c4:c4:cd:77:83:21:fa:88:56:ac:03:b3:1f:
         64:07:db:95:17:67:88:43:d3:38:23:75:4e:e3:22:2f:c0:4f
-----BEGIN CERTIFICATE-----
MIIBfjCCASOgAwIBAgIEWwUNMDAKBggqhkjOPQQDAjARMQ8wDQYDVQQDDAZIRVMt
Q0EwHhcNMTgwNTE5MjEwMDAwWhcNMjAwNTE5MjEwMDAwWjAqMSgwJgYDVQQDDB9E
TE1TVGVzdGVyLXNpZ25hdHVyZS1NYW5hZ2VtZW50MFkwEwYHKoZIzj0CAQYIKoZI
zj0DAQcDQgAEyJ6rslBRAPhGBsKnoyX0lotlTPH8sjN5TRUbD3ZJUeccyJJNVwyF
DS/JCQur8f216C2/IMtV/awqSeH+rj0aM6NQME4wDgYDVR0PAQH/BAQDAgeAMDwG
A1UdIwQ1MDOAFB8/y5O7hRRKfB/dSGgO8KYc+BxXoRWkEzARMQ8wDQYDVQQDDAZI
RVMtQ0GCBFsFDMgwCgYIKoZIzj0EAwIDSQAwRgIhAICruryVvm5xswMpEfwB5GyO
ROETBil6UfrunaexZti6AiEA+cTEzXeDIfqIVqwDsx9kB9uVF2eIQ9M4I3VO4yIv
wE8=
-----END CERTIFICATE-----
```