Release Notes DSMR V4.0.4

Dutch Smart Meter Requirements

By order of:	**Netbeheer Nederland**
Date:	**March 22th, 2012**
Version:	**4.0.4**
Status:	**Final**

Change summary

Version	Change
4.0	Initial version of the release notes for the DSMR 4.0
4.0.1	First update of the release notes for the DSMR 4.0
4.0.2	Second update of the release notes for the DMSR 4.0
4.0.3	Third update of the release notes for the DMSR 4.0
4.0.4	Forth update of the release notes for the DMSR 4.0

CONTENTS

1 Introduction 4
1.1 Normative references 4

2 DSMR v4.0.4 Main Changes 5

3 DSMR v4.0.4 P1 Changes 8

4 DSMR v4.0.4 P2 Changes 12

5 DSMR v4.0.4 P3 Changes 13

6. In DSMR v4.0.4 GPRS Changes 44

1 INTRODUCTION

This document describes the changes incorporated in the Dutch Smart Meter Requirements version v4.0.4 compared to the previous version of the DSMR v4.0. The intention of this document is to make the changes in the various DSMR documents more transparent for the readers.

1.1 Normative references

The following standards are referred to in document. For undated references the latest edition applies.

Ref No	Document	Description
1.	NTA 8130 NL:2007	Netherlands Technical Agreement -"Minimum set of functions for metering of electricity, gas and thermal energy for domestic customers"
2.	Dutch Smart Meter Requirements v4.0 final Main	The main document of the Dutch Smart Meter Requirements, containing all definitions and most of the use cases and requirements
3.	Dutch Smart Meter Requirements v4.0 final P1	Companion standard P1
4.	Dutch Smart Meter Requirements v4.0.2 final P2	Companion standard P2
5.	Dutch Smart Meter Requirements v4.0 final P3	Companion standard P3
6.	Dutch Smart Meter Requirements v4.0 final GPRS	Additional document describing the requirements for the GPRS infrastructure as part of the Dutch Smart Meter Specification.
7.	Dutch Smart Meter Requirements v4.0 Release notes 4.0.1	
8.	Dutch Smart Meter Requirements v4.0 Release notes 4.0.2	
9.	Dutch Smart Meter Requirements v4.0 Release notes 4.0.3	

2 DSMR V4.0.4 MAIN CHANGES

This section lists all the changes incorporated in the Dutch Smart Meter Requirements v4.0.4 Final Main document.

1. Changed requirement DSMR-M 4.3.55 to:

Description	As required by MID the software version identification of Legally Relevant software shall be easily provided by the measuring instrument.						
Rationale	The version identification of Legally Relevant software shall easily be shown on the display.						
Fit criterion	The version identification of Legally Relevant software must be shown on the display in the test mode of the G-meter.						
History	Mar. 2011	**Origin**	TST	**Port**	n.a.	**Applicable**	G meter

2. Changed requirement DSMR-M 4.3.56 to:

Description	It must be possible to activate a test mode in the G meter.						
Rationale	Testing of a meter must be done in a reasonable time. This is not possible if the standard resolution is not precise enough. In that case it must be possible to activate a test mode in the G meter during which the registers have a 0,1 litre resolution for G4 meters and a 1 litre resolution for meters >= G6. In test mode the Legally Relevant Software is shown in the display						
Fit criterion	It must be possible to activate a test mode in the G meter during which the registers have a 0,1 litre resolution for G4 meters and a 1 litre resolution for meters >= G6. In this test mode also the Legally Relevant Software is shown in the display. In case of a display with sleeping mode functionality: ▪ After activating the display by pushing the button, test mode is activated by a manufacturer specific action.. The code for the LR software is shown in test mode in the next sequence: Display test → Index value → LR → Display test →…… ▪ Return to sleeping mode after a manufacturer specific timeout (and optional by an action) In the case of a display without sleeping mode functionality activating of the test mode is done: ▪ by a manufacturer specific action. The code for the LR software is shown in test mode in the next sequence: Display test → Index value → LR → Display test → ….. ▪ Return to normal mode after a manufacturer specific timeout (and optional by an action). ▪ Testing at Qmin may not take more than 30 minutes. ▪ Test results shall be reproducible and repeatable (as described in MID).						
History	Nov. 2010	**Origin**	TST	**Port**	n.a.	**Applicable**	G meter

3. In section 4.2 replaced requirement 4.4.8 with:
DSMR-M 4.4.8a

Description	The equipment shall support functionality to configure the supported authentication mechanism on P0 and P3 port.						
Rationale	This functionality give the opportunity to the Central System to select an other authentication mechanism when one authentication mechanism is not safe anymore.						
Fit criterion	It shall be possible to configure for HLS mechanism 3,4 and 5 or any combination for both P0 and P3 whether the meter accepts the authentication request or reject the authentication request.						
History	Jan. 2011	**Origin**		**Port**	P0, P3	**Applicable**	E Meter

DSMR-M 4.4.8b

Description	The equipment shall support functionality to configure different HLS mechanisms for P0 and P3 port. .						
Rationale	Some Grid Operators use a PDA connected to the P0 port for commissioning the E-Meter using HLS mechanism 4 with a secret that is shared with a group of meters. Access to the meter via the P3 port using such shared secret shall be prevented.						
Fit criterion	The HLS mechanism on P0 and P3 port can be configured independently from each other.						
History	Jan. 2011	**Origin**		**Port**	P0, P3	**Applicable**	E Meter

4. In section 5.2.1 changed requirement 4.5.13:

Description	The E meter shall provide functionality to retrieve actual meter reads.						
Rationale	Under some circumstances an actual meter read is needed (for example, consider a call-centre agent handling a customer complaint). This is required in NTA 8130 (see § 5.2.4).						
Fit criterion	The information provided as actual meter readings shall at least contain the following information: ▪ Actual meter reading E using kWh as the unit of measurement; ▪ Most recent meter reading G available in the E meter ~~(if not older than 24 hours)~~ using m^3 as the unit of measurement;						
History	Nov. 2007	**Origin**	NTA 8130 ((§5.2.4)	**Port**	P3	**Applicable**	E meter

5. In section 6.1.2.1 removed requirement 4.6.5

6. In section 6.1.2.1 removed requirement 4.6.6

7. In section 6.1.2.2 changed requirement 4.6.8:

Description	The equipment shall log the event of successful verification of a new version of the firmware.
Rationale	For maintenance reasons it is important to verify if new firmware was received by the equipment and at what time and date it was verified.
Fit criterion	The log information for the event shall at least contain the following information: • Time stamp at which the new version of the firmware was verified

	~~• Version number of the verified firmware.~~						
History	Nov. 2007	**Origin**	I&M	**Port**	n.a.	**Applicable**	E meter

8. In section 6.1.2.3 changed requirement 4.6.9:

Description	The metering equipment shall deploy the new version ~~either~~ immediately. ~~or time based (at a designated date and time).~~						
Rationale	The metering equipment shall deploy the new version ~~either~~ immediately. ~~or at a designated date and time.~~						
Fit criterion	The new version of the firmware is the operational version of the firmware in the equipment. If the deployment date coincides with a power outage, the upgrade shall be deployed after power on. In this case no error shall be raised.						
History	Nov. 2007	**Origin**	I&M	**Port**	n.a.	**Applicable**	E meter

9. Changed title for section 6.1.6.2 in 'Removed'

10. In section 6.1.6.2 removed requirement 4.6.49

3 DSMR V4.0.4 P1 CHANGES

This section lists all the changes incorporated in the Dutch Smart Meter Requirements V4.0.4 Final P1 document.

1. Added number of rows in table in section 5.12

Value	OBIS reference	Attribute	Class ID	Value Format	Value Unit
Text message codes: numeric 8 digits	0-0:96.13.1.255	2 Value	1 Data	Sn (n=0..8),, tag 9	
Text message max 1024 characters.	0-0:96.13.0.255	2 Value	1 Data	Sn (n=0..2048), tag 9	
Instantaneous current L1 in A resolution.	**1-0:31.7.0.255**	**2 Value**	**3 Register**	**F3(0,0), tag 18**	**A**
Instantaneous current L2 in A resolution.	**1-0:51.7.0.255**	**2 Value**	**3 Register**	**F3(0,0), tag 18**	**A**
Instantaneous current L3 in A resolution.	**1-0:71.7.0.255**	**2 Value**	**3 Register**	**F3(0,0), tag 18**	**A**
Instantaneous active power L1 (+P) in W resolution	**1-0:21.7.0.255**	**2 Value**	**3 Register**	**F5(3,3), tag 18**	**kW**
Instantaneous active power L2 (+P) in W resolution	**1-0:41.7.0.255**	**2 Value**	**3 Register**	**F5(3,3), tag 18**	**kW**
Instantaneous ac-	**1-0:61.7.0.255**	**2**	**3**	**F5(3,3), tag 18**	**kW**

Value	OBIS reference	Attribute	Class ID	Value Format	Value Unit
tive power L3 (+P) in W resolution		**Value**	**Register**		
Instantaneous active power L1 (-P) in W resolution	**1-0:22.7.0.255**	**2 Value**	**3 Register**	**F5(3,3), tag 18**	**kW**
Instantaneous active power L2 (-P) in W resolution	**1-0:42.7.0.255**	**2 Value**	**3 Register**	**F5(3,3), tag 18**	**kW**
Instantaneous active power L3 (-P) in W resolution	**1-0:62.7.0.255**	**2 Value**	**3 Register**	**F5(3,3), tag 18**	**kW**
Device-Type	0-n:24.1.0.255	9 Device type	72 M-Bus client	F3(0,0), tag 17	
Equipment identifier (Gas)	0-n:96.1.0.255	2 Value	1 Data	Sn (n=0..96), tag 9	

2. Added a sentence in section 5.12:

The following table holds data objects represented with P1 Interface together with OBIS reference including object Attribute and Value Format for Reduced ID codes.

Every line (except the last one) is ended with a CR/LF (Carriage Return / Line Feed).

3. In section 5.13 corrected the checksum in the example P1 telegram and added new lines

- **Instantaneous current per phase**
- **Instantaneous active power (+P) per phase**
- **Instantaneous active power (-P) per phase**

```
/ISk5\2MT382-1000

1-3:0.2.8(50)
0-0:1.0.0(101209113020W)
0-0:96.1.1(4B384547303034303436333935353037)
1-0:1.8.1(123456.789*kWh)
1-0:1.8.2(123456.789*kWh)
1-0:2.8.1(123456.789*kWh)
```

```
1-0:2.8.2(123456.789*kWh)
0-0:96.14.0(0002)
1-0:1.7.0(01.193*kW)
1-0:2.7.0(00.000*kW)
0-0:17.0.0(016.1*kW)
0-0:96.3.10(1)
0-0:96.7.21(00004)
0-0:96.7.9(00002)
1-0:99.97.0(2)(1-0:96.7.19)(101208152415W)(0000000240*s)(101208151004W)(0000000301*s)
1-0:32.32.0(00002)
1-0:52.32.0(00001)
1-0:72:32.0(00000)
1-0:32.36.0(00000)
1-0:52.36.0(00003)
1-0:72.36.0(00000)
0-0:96.13.1(3031203631203831)
0-
0:96.13.0(303132333435363738393A3B3C3D3E3F303132333435363738393A3B3C3D3E3F303132333435363738393A3
B3C3D3E3F303132333435363738393A3B3C3D3E3F303132333435363738393A3B3C3D3E3F)
1-0:31.7.0.255(001*A)
1-0:51.7.0.255(002*A)
1-0:71.7.0.255(003*A)
1-0:21.7.0.255(01.111*kW)
1-0:41.7.0.255(02.222*kW)
1-0:61.7.0.255(03.333*kW)
1-0:22.7.0.255(04.444*kW)
1-0:42.7.0.255(05.555*kW)
1-0:62.7.0.255(06.666*kW)
0-1:24.1.0(003)
0-1:96.1.0(32323232414243443132333435363738 39)
0-1:24.2.1(101209110000W)(12785.123*m3)
0-1:24.4.0(1)
!141B
```

4. Added number of rows at the end of the table in section 6.1 of P1 companion standard.

Value	OBIS reference	NTA Use Case reference
Instantaneous current L1	**1-0:31.7.0.255**	**Use case 3: Provide actual meter reads through P1**
Instantaneous current L2	**1-0:51.7.0.255**	**Use case 3: Provide actual meter reads through P1**
Instantaneous current L3	**1-0:71.7.0.255**	**Use case 3: Provide actual meter reads through P1**
Instantaneous active power L1 (+P)	**1-0:21.7.0.255**	**Use case 3: Provide actual meter reads through P1**

Instantaneous active power L2 (+P)	**1-0:41.7.0.255**	**Use case 3: Provide actual meter reads through P1**
Instantaneous active power L3 (+P)	**1-0:61.7.0.255**	**Use case 3: Provide actual meter reads through P1**
Instantaneous active power L1 (-P)	**1-0:22.7.0.255**	**Use case 3: Provide actual meter reads through P1**
Instantaneous active power L2 (-P)	**1-0:42.7.0.255**	**Use case 3: Provide actual meter reads through P1**
Instantaneous active power L3 (-P)	**1-0:62.7.0.255**	**Use case 3: Provide actual meter reads through P1**

4 DSMR V4.0.4 P2 CHANGES

All changes made to the P2 Companion Standard are covered by the revised issue of the P2 Companion Standard. These changes are not listed in this document.

5 DSMR V4.0.4 P3 CHANGES

This section lists all the changes incorporated in the Dutch Smart Meter Requirements V4.0.4 Final P3 document.

1. In section 3.2 replaced the text from the start of this section until the paragraph called “Transfer Key handling” with the following:

3.2.1 General

After commissioning, the *security policy* attribute in the Security Setup object (OBIS 0-0:43.0.0.255) shall have the value 3 (Message Encryption and Message Authentication). If this value is not the factory setting, the value can only be changed via the method 'security activate'. Once this attribute has the value 3, it is not possible to lower the value.

3.2.2 Access Security

The following table defines the requirements for Data Access Security for the Public and Management clients via the P0 and P3:

Data security / Acces security	no security	authenticated	encrypted	authenticated and encrypted	Remark about Data security (in DLMS "security policy): See Blue Book, page 71: NOTE The security policy can only be strengthened. Strengthening is setting the policy to a higher numeric value.
No security (Lowest Level Security)	allowed for Public Client only	not allowed	not allowed	not allowed	
LLS (Low Level Security)	not allowed	not allowed	not allowed	not allowed	
HLS2	not allowed	not allowed	not allowed	not allowed	
HLS3	allowed	allowed	allowed	allowed	See DSMR M 4.4.8a and 4.4.8b
HLS4	allowed	allowed	allowed	allowed	See DSMR M 4.4.8a and 4.4.8b
HLS5	allowed	allowed	allowed	allowed	See DSMR M 4.4.8a and 4.4.8b

HLS 3 and 4 do use the HLS secret
HLS 5 does use Authentication Key and Encryption Key
The same encryption methods (data security) must be used for HLS 5, 4 en 3

Access via P0 is allowed for commissioning purposes and maintenance purposes. The access can be toggled via the *P0_enable* bit.

3.2.3 Transport Security

3.2.3.1 Data exchange

For data transport security for data exchange, the following applies:
If the Public Client has established an Association with the Management Logical Device, data exchange may be done in an unciphered application context. Neither message encryption nor message authentication is required.
If the Management Client has established an Association with the Management Logical Device, the following requirements apply:

1. All data exchange shall take place in a ciphered application context, indicated by *context_id* 3 (Logical Name Services and Ciphering) for the field *application_context_name* in the AARQ.

2. Only 'Message Encryption & Message Authentication' with the mechanisms provided by DLMS/COSEM Security Suite 0 shall be used. This security suite contains the following Authentication and Encryption Algorithms:

Security Suite ID	Authentication Algorithm	Encryption Algorithm
0	AES-GCM-128	AES-GCM-128 and AES-128 for key wrapping

   Using both Message Authentication and Message Encryptions means that the *security_policy* attribute in the Security Setup object (Logical Name 0-0:43.0.0.255) shall have the value 3.

3. Resulting from 2, the *InitiateRequest* and *InitiateResponse* xDLMS APDU's in the AARQ and RLRQ respectively shall be encrypted and authenticated.

4. All other fields in the AARQ and RLRQ shall be transmitted in clear text, according to the examples in the Green Book.

5. Only Global Encryption Keys shall be used for message encryption. The use of dedicated keys that are transported via the AARQ is NOT allowed.

If the Management Client has established an Association with the Management Logical Device, data exchange in an unciphered application context (with *security policy* set to 0), is intended to be used for the purpose of commissioning the meter via P0.

4 In section 4.2.3 added a sentence:
Clearing is possible by the Management Client (object 0-0:97.97.0.255) for errors and object 0-0:97.98.0.255 for alarms)

5 In section 4.2.3 added an error for Power Up:

Group	Byte	Bit	Meaning	Events
Other Errors	1 (LSB)	0	Clock invalid	6
		1	Replace battery	7
		2	Power Up	2
		3	not used	
		4	not used	
		5	not used	
		6	not used	
		7	not used	

6 In section 4.2.3 replaced the text below the table with normal Errors with:
For a detailed description see the corresponding event. Critical errors, New M-Bus device discovered channel x, Power Up and the replacement of the battery must be cleared via the management client, all others clear themselves if the corresponding error condition has disappeared.

7 In section 4.2.4 added an alarm for Power Up

Group	Byte	Bit	Meaning	Events
Other Alarms	1 (LSB)	0	Clock invalid	6
		1	Replace battery	7
		2	Power Up	2
		3	not used	
		4	not used	
		5	not used	
		6	not used	
		7	not used	

8 In section 5.5, Object Data of billing period 1 (Class ID: 7) removed the X for the capture method via the Management Client

Data of billing period 1 (Class ID: 7) Monthly billing values				P	M	Pr
1	Logical name	Octet-string	**0-0:98.1.0.255**		R	
2	buffer	array			R	
3	capture_objects[1]	array	{8,0-0:1.0.0.255,2,0}; {3,1-0:1.8.1.255,2,0}; {3,1-0:1.8.2.255,2,0}; {3,1-0:2.8.1.255,2,0}; {3,1-0:2.8.2.255,2,0}; {4,0-1:24.2.1.255,2,0}; {4,0-1:24.2.1.255,5,0}; {4,0-2:24.2.1.255,2,0}; {4,0-2:24.2.1.255,5,0}; {4,0-3:24.2.1.255,2,0}; {4,0-3:24.2.1.255,5,0}; {4,0-4:24.2.1.255,2,0}; {4,0-4:24.2.1.255,5,0}; (= clock;+A rate 1;+A rate 2;−A rate 1;−A rate 2; 4 M-Bus register values & capture times of the M-Bus registers by the M-Bus devices) Can be extended with additional tariff registers		R	
4	capture_period	double-long-unsigned	0, triggered from single action scheduler with billing period 1		R	
5	sort_method	enum	1, unsorted (FIFO)		R	
6	sort_object	object definition	None, unsorted		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	13 months		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

[1] The value of a captured object may be replaced by “null-data” if it can be unambiguously recovered from the previous value (for example for time: if it can be calculated from the previous value and capture_period; or for a value: if it is equal to the previous value

9 In section 5.7 Object Event Log (Class ID: 7) corrected an erroneous Obis-code

10 In section 5.7 Object Event Log (Class ID: 7) removed the X for the capture method via the Management Client

Event Log (Class ID: 7) **Standard event log containing errors and alarms**				**P**	**M**	**Pr**
1	Logical name	Octet-string	**0-0:99.98.0.255**		R	
2	buffer	Array			R	
3	capture_objects	Array	{8,0-0:1.0.0.255,2,0}; {1,0-0:96.11.0.255,2,0} (= clock;event code) (See definition of event codes in paragraph 4.2.1)		R	
4	capture_period	double-long-unsigned	0, asynchronously		R	
5	sort_method	Enum	1, unsorted (FIFO)		R	
6	sort_object	object definition	None, unsorted		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	100		R	
	Specific methods	***m/o***				
1	reset ()	M			X	
2	capture ()	M			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

11 In section 5.8 Fraud Detection Log (Class ID: 7) removed the X for the capture method via the Management Client

Fraud Detection Log (Class ID: 7) **Event log containing all fraud detection events**			**P**	**M**	**Pr**
Logical name	Octet-string	**0-0:99.98.1.255**		R	
buffer	array			R	
capture_objects	Array	{8,0-0:1.0.0.255,2,0}; {1,0-0:96.11.1.255,2,0} (= clock;tamper event code) (See definition of event codes in paragraph 4.2.1)		R	

capture_period	double-long-unsigned	0, asynchronously		R	
sort_method	enum	1, unsorted (FIFO)		R	
sort_object	object definition	None, unsorted		R	
entries_in_use	double-long-unsigned			R	
profile_entries	double-long-unsigned	30		R	
Specific methods	***m/o***				
reset ()	m			X	
capture ()	m			~~X~~	
Reserved from previous versions					
Reserved from previous versions					

12 In section 5.9 Power Failure Event Log (Class ID: 7) removed the X for the capture method via the Management Client

Power Failure Event Log (Class ID: 7)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:99.97.0.255**		R	
2	buffer	array			R	
3	capture_objects	Array [2]	{8,0-0:1.0.0.255,2,0}; {3,0-0:96.7.19.255,2,0} (= timestamp; duration of long power failures in any phase) Timestamp = end of power failure		R	
4	capture_period	double-long-unsigned	0, asynchronously		R	
5	sort_method	enum	1, unsorted (FIFO)		R	
6	sort_object	object definition	None, unsorted		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	10		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

13 In section 5.10 Control Log (Class ID: 7) removed the X for the capture method via the Management Client

Control log (Class ID:7) **Changes of the states related to the disconnect control are recorded (changing threshold, connect, disconnect)**				P	M	Pr
1	logical_name	octet-string	**0-0:99.98.2.255**		R	
2	buffer	array			R	
3	capture_objects	array	{8, 0-0:1.0.0.255,2,0}, clock; {1, 0-0:96.11.2.255,2,0}, control event code {71, 1-0:17.0.0.255,3,0}, limiter threshold Event codes must be defined in chapter 4.2.1		R	
4	capture_period	double-long-unsigned	Value = 0, asynchronously		R	
5	sort_method	enum	Value = 1, unsorted (FIFO)		R	
6	sort_object	object definition	None, unsorted		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	10		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

14 In section 5.16 General display readout (Class ID: 7) removed the X for the capture method via the Management Client

General display readout (Class ID: 7) **Auto scroll readout list**				**P**	**M**	**Pr**
1	Logical name	Octet-string	**0-0:21.0.1.255**		R	
2	buffer	array	last readout		R	
3	capture_objects	Array [64]	readout objects, a maximum of 64 entries. See Annex B for the items in the list		RW	
4	capture_period	double-long-unsigned	10, update period [s]		R	
5	sort_method	enum	1, unsorted (FIFO)		R	

6	sort_object	object definition	None, unsorted		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	1		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

15 In section 5.16 Alternate display readout (Class ID: 7) removed the X for the capture method via the Management Client

Alternate display readout (Class ID: 7) Manual scroll readout list			**P**	**M**	**Pr**
Logical name	Octet-string	**0-0:21.0.2.255**		R	
buffer	array	last readout		R	
capture_objects	Array [64]	readout objects, a maximum of 64 entries. See Annex B for the items in the list		RW	
capture_period	double-long-unsigned	10, update period [s]		R	
sort_method	enum	1, unsorted (FIFO)		R	
sort_object	object definition	None, unsorted		R	
entries_in_use	double-long-unsigned			R	
profile_entries	double-long-unsigned	1		R	
Specific methods	***m/o***				
reset ()	m			X	
capture ()	m			~~X~~	
Reserved from previous versions					
Reserved from previous versions					

16 In section 5.16 Service display readout (Class ID: 7) removed the X for the capture method via the Management Client

Service display readout (Class ID: 7) Testmode readout list				**P**	**M**	**Pr**
1	Logical name	Octet-string	**0-0:21.0.3.255**		R	
2	buffer	array	last readout		R	

3	capture_objects	Array [64]	readout objects, a maximum of 64 entries. See Annex B for the items in the list		RW	
4	capture_period	double-long-unsigned	10, update period [s]		R	
5	sort_method	enum	1, unsorted (FIFO)		R	
6	sort_object	object definition	None, unsorted		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	1		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

17 In section 6.3 corrected some erroneous Obis-codes in attribute 3:

18 In section 5.16 Load profile with period 1 (Class ID: 7) removed the X for the capture method via the Management Client

19 In section 5.16 Load profile with period 2 (Class ID: 7) removed the X for the capture method via the Management Client

Load profile with period 1 (Class ID: 7) E interval readings every 15 minutes				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:99.1.0.255**		R	
2	buffer	array	The buffer must be filled monotonously, i.e. no irregular entries are allowed = exactly one entry per capture period		R	
3	capture_objects	Array	{8,0-0:1.0.0.255,2,0}; {1,0-0:96.10.2.255,2,0}; {3,1-0:1.8.0.255,2,0}; {3,1-0:2.8.0.255,2,0} (= clock; AMR profile status; +A;−A) Profile status ➔ see paragraph 4.2.5		R	
4	capture_period	double-long-unsigned	900 (15 minutes)		R	

5	sort_method	enum	1 or 3 (unsorted (FIFO) or sorted (largest))		R	
6	sort_object	object definition	none or {8,0-0:1.0.0.255,2,0} (unsorted or sorted by clock)		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	(10 days)		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			X	
3	Reserved from previous versions					
4	Reserved from previous versions					

Load profile with period 2 (Class ID: 7) **Daily Combined billing values**				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:99.2.0.255**		R	
2	buffer	array	The buffer must be filled monotonously, i.e. no irregular entries are allowed = exactly one entry per capture period		R	
3	capture_objects	Array	{8,0-0:1.0.0.255,2,0}; clock {1,0-0:96.10.2.255,2,0} AMR profile status {3,1-0:1.8.1.255,2,0}; +A rate1 {3,1-0:1.8.2.255,2,0} +A rate 2 {3,1-0:2.8.1.255,2,0}; -A rate1 {3,1-0:2.8.2.255,2,0} -A rate 2 {4,0-0.1.24.2.1.255,2,0}, M-Bus Master Value 1 Channel 1 {4,0-0.1.24.2.1.255,5,0}, M-Bus Master Value 1 Channel 1 Capture time of the M-Bus registers by the M-Bus devices) {4,0-0.2.24.2.1.255,2,0}, M-Bus Master Value 1 Channel 2 {4,0-0.2.24.2.1.255,5,0}, M-Bus Master Value 1 Channel 2 Capture time of the M-Bus registers by the M-Bus devices) {4,0-0.3.24.2.1.255,2,0}, M-Bus Master Value 1 Channel 3		R	

			{4,0-0.3.24.2.1.255,5,0}, M-Bus Master Value 1 Channel 3 Capture time of the M-Bus registers by the M-Bus devices) {4,0-0.4.24.2.1.255,2,0) M-Bus Master Value 1 Channel 4 {4,0-0.4.24.2.1.255,5,0}, M-Bus Master Value 1 Channel 4 Capture time of the M-Bus registers by the M-Bus devices) AMR Profile status see paragraph 4.2.5[2]			
4	capture_period	double-long-unsigned	86400 (daily)		R	
5	sort_method	enum	1 or 3 (unsorted (FIFO) or sorted (largest))		R	
6	sort_object	object definition	none or {8,0-0:1.0.0.255,2,0} (unsorted or sorted by clock)		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	40 (40 days)		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

20 In section 6.4 changed the following Object:

Average voltage L1 (Class ID: 3)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:32.24.0.255**		R	
2	Value	long-unsigned	10 minutes average voltage. Averaging scheme 3 is used. ~~for instantaneous values~~		R	
3	Scaler_unit	scal_unit_type	Value = {0,35}, scaler=0, unit=V, resolution: 0 V		R	

[2] This represents the combined statuses of all the devices (using bit 2 of the AMR profile status). To identify the faulty device it is necessary to read the interval data of the different meters.

	Specific methods	***m/o***				
	reset (data)	O				

Average active power (+P) L1 (Class ID: 3)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:21.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,27}, scaler=1, unit=W		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average active power (-P) L1 (Class ID: 3)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:22.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,27}, scaler=1, unit=W		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average reactive power (+Q) L1 (Class ID: 3)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:23.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,29}, scaler=1, unit=var		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average reactive power (-Q) L1(Class ID: 3)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:24.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,29}, scaler=1, unit=var		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average current L1(Class ID: 3)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**1-0:31.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average current. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,33}, scaler=0, unit=A		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average voltage L2 (Class ID: 3) (polyphase meters only)				P	M	Pr
1	Logical name	Octet-string	**1-0:52.24.0.255**		R	
2	Value	long-unsigned	10 minutes average voltage. Averaging scheme 3 is used. ~~for instantaneous values~~		R	
3	Scaler_unit	scal_unit_type	Value = {0,35}, scaler=0, unit=V, resolution: 0 V		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average active power (+P) L2 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:41.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,27}, scaler=1, unit=W		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average active power (-P) L2 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:42.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,27}, scaler=1, unit=W		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average reactive power (+Q) L2 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:43.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,29}, scaler=1, unit=var		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average reactive power (-Q) L2 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:44.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,29}, scaler=1, unit=var		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average current L2(Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:51.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average current. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,33}, scaler=0, unit=A		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average voltage L3 (Class ID: 3) (polyphase meters only)				P	M	Pr
1	Logical name	Octet-string	**1-0:72.24.0.255**		R	
2	Value	long-unsigned	10 minutes average voltage. Averaging scheme 3 is used. ~~for instantaneous values~~		R	
3	Scaler_unit	scal_unit_type	Value = {0,35}, scaler=0, unit=V, resolution: 0 V		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average active power (+P) L3 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:61.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,27}, scaler=1, unit=W		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average active power (-P) L3 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:62.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,27}, scaler=1, unit=W		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average reactive power (+Q) L3 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:63.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,29}, scaler=1, unit=var		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average reactive power (-Q) L3 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:64.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average power. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,29}, scaler=1, unit=var		R	
	Specific methods	***m/o***				
	reset (data)	o				

Average current L3 (Class ID: 3)				P	M	Pr
1	Logical name	Octet-string	**1-0:71.24.0.255**		R	
2	Value	long-unsigned	**10 minutes average current. Averaging scheme 3 is used.**		R	
3	Scaler_unit	scal_unit_type	Value = {0,33}, scaler=0, unit=A		R	
	Specific methods	***m/o***				
	reset (data)	o				

21 In section 7.1, Object 0-x:24.1.0.255, attribute 10 changed the acces right for the management client to R.

22 In section 7.1, Object 0-x:24.1.0.255, attribute 14 added the following remark:
"The states are based on the methods 7 and 8 that are part of the object 'M-bus client object' as defined in the bluebook. For example the status 'encryption_key_set and transferred' means that both methods 7 and 8 are invoked."

M-Bus Client Setup (Class ID: 72) **Setup in M-Bus master for every M-Bus client (4 instances, one per channel, see additional info)**				P	M	Pr
1	Logical name	Octet-string	**0-x:24.1.0.255**[3]		R	
2	mbus_port_reference	Octet-string			R	
3	capture_definition	array			RW	
4	capture_period	double-long-unsigned			RW	
5	primary_address	unsigned			RW	
6	identification_number	double-long unsigned interpreted as BCD[4]	(Only last 8 digits of identiification number) Part of short ID		RW	
7	manufacturer_Id	long_unsigned	**Part of short ID**		RW	
8	version	unsigned	**Part of short ID**		RW	
9	device_type	unsigned	**Part of short ID**		RW	
10	access_number	unsigned			R~~W~~	
11	status	unsigned			R	
12	alarm	unsigned			R	

[3] A new DLMS channel will be assigned to each new device, x=channel number 1..4

[4] See TC294_N0280_Updated_draft_prEN_13757-3_for_launch_of_pub

13	Configuration	long-unsigned	Default value is 99 (unknown), 0 or 15		R	
14	encryption_key_status	enum	Carries the status of the encryption key [5] Enum: (0) no encryption_key (1) encryption_key set (2) encryption_key transferred (3) encryption_key set and transferred (4) encryption_key in use		R	
	Specific methods	***m/o***				
1	slave_install	m			X	
2	slave_deinstall	m			X	
3	capture	m			X	
4	reset_alarm	m			X	
5	synchronize_clock	m			X	
6	data_send	m			X	
7	set_encryption_key	m			X	
8	transfer_key	m			X	

23 In section 7.3.2 changed the text for atribute 2 of the M-Bus Device configuration (Class ID: 4)

M-Bus Device configuration (Class ID: 4) Instance specific (4 instances, one per channel)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**0-x:24.2.2.255** (x=channel number (1..4), One channel per M-Bus device		R	
2	Value	octet-string[255]	String with concatenation of 5 (variable length) information fields: [Model/version] [Hardware version number] [Metrology (firmware) version number] [Other software version number] [Meter Configuration] Each fields is to be terminated with CR/LF (ASCII characters <CR><LF>)		R	

[5] **The states are based on the methods 7 and 8 that are part of the object 'M-bus client object' as defined in the bluebook. For example the status 'encryption_key_set and transferred' means that both methods 7 and 8 are invoked.**

			The first 4 information fields have a maximum length of 61 characters **each** and contain the info as received from the M-Bus. If the information from the M-Bus is too long then the leftmost octets are skipped. The last information field is 1 character (8 bits as received from M-Bus device).			
3	scaler_unit	scal_unit_type	0 (not a numeric value)		R	
4	status	octet-string	status of M-Bus device		R	
5	capture_time	octet-string	Time of last successful readout. The M-Bus device configuration is (at least) read by the E-Meter from the M-Bus device at the end of the M-Bus device installation procedure.		R	
	Specific methods	***m/o***				
1	reset (data)	o				

24 In section 7.4 M-bus Master Load profile with period 1 (Class ID: 7) removed the X for the capture method via the Management Client

M-Bus Master Load profile with period 1 (Class ID: 7) Hourly interval readings of M-Bus devices (4 instances, one per channel)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**0-x:24.3.0.255** (x=channel number (1..4))		R	
2	buffer	array	The buffer must be filled monotonously, i.e. no irregular entries are allowed		R	
3	capture_objects	Array	{8,0-0:1.0.0.255,2,0}; {1,0-x: 96.10.3.255,2,0} {4,0-x:24.2.1.255,2,0} {4,0-x:24.2.1.255,5,0} (x=channel number (1..4)) (=clock; AMR profile status; M-Bus master value object & capture times of the M-Bus registers by the M-Bus devices) AMR profile status see paragraph 4.2.5		R	
4	capture_period	double-long-unsigned	3600, every hour		R	

5	sort_method	enum	1 or 3 (unsorted (FIFO) or sorted (largest))		R	
6	sort_object	object definition	none or {8,0-0:1.0.0.255,2,0}(unsorted or sorted by clock)		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	240 (10 days)		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

25 In section 7.5 M-bus Event Log (Class ID: 7) removed the X for the capture method via the Management Client

M-Bus Event Log (Class ID: 7) **M-Bus event log containing errors and alarms**				**P**	**M**	**Pr**
Logical name	Octet-string	**0-0:99.98.3.255**		R		
buffer	array	The buffer must be filled monotonously, i.e. no irregular entries are allowed = exactly one entry per capture period		R		
capture_objects	Array	{8,0-0:1.0.0.255,2,0}; {1,0-0.96.11.3.255,2,0} (= clock;event code) M-Bus event codes must be defined, see 4.2.1		R		
capture_period	double-long-unsigned	0, asynchronously		R		
sort_method	enum	1, unsorted (FIFO)		R		
sort_object	object definition	None, unsorted		R		
entries_in_use	double-long-unsigned			R		
profile_entries	double-long-unsigned	≥10		R		
Specific methods	***m/o***					
reset ()	m			X		
capture ()	m			~~X~~		
Reserved from previous versions						
Reserved from previous versions						

26 In section 7.5 M-bus Master Control Log (Class ID: 7) removed the X for the capture method via the Management Client

M-Bus Master Control log (Class ID: 7) **Changes of the states related to the disconnect control are recorded (open, close) (4 instances, one per channel)**				**P**	**M**	**Pr**
1	Logical name	Octet-string	**0-x:24.5.0.255** (x=channel number (1..4))		R	
2	buffer	Array			R	
3	capture_objects	Array	{8,0-0:1.0.0.255,2,0}; {1, 0-x: 96.11.4.255, 2, 0} (x=channel number (1..4)) (=clock; control event code) Event codes must be defined (paragraph 4.2.1)		R	
4	capture_period	double-long-unsigned	0, asynchronously		R	
5	sort_method	Enum	1, unsorted (FIFO)		R	
6	sort_object	Object definition	None, unsorted		R	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	10		R	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			~~X~~	
3	Reserved from previous versions					
4	Reserved from previous versions					

27 In section 8.2 changed the text under the object:
The connection watchdog timer object in the electricity meter (obis code 0-1:94.31.2.255) holds an attribute with the value of the watchdog timer in hours. A watchdog timer makes sure that the ~~connection~~ **modem** is reset after a defined period of no contact with the CS

28 In section 8.3 changed the configuration object:

Configuration object (Class ID: 1)				**P**	**M**	**Pr**
1	logical_name	Octet string	0-1:94.31.3.255		R	
2	value	Structure	See below		RW	
	Specific methods	***m/o***				

Attribute description
Value ::= structure
{
~~Message_content array of Content~~

```
    IP_message_target_address     structure
    {
        Domain_name                   octetstring,
        Port_number                   long_unsigned
    }
    GPRS_operation_mode           enum
    PDP_context_reconnect_count   unsigned
    PDP_context_reconnect_interval   unsigned
    PDP_context_reconnect_multiplier   unsigned
    Control_mode_Ebreaker         enum
    Flags                         bitstring (16)
}

Content ::= structure
{
    choice
    {
        octetstring                   -- minimum size 35
        structure
        {
            Logicalname       octetstring
            Attribute_id      unsigned
        }
    }
```

IP_message_content	A configurable attribute that contains contents of the IP message send when a PDP context is established. • It shall be possible to define one value for the content of the message • It shall be possible to configure the Equipment Identifier as the content of the IP message. It is allowed to make a reference to the cosem object containing the Equipment Identifier • For each value, it is possible to refer to any other COSEM objects in the meter • For each value, it is possible to insert a string with a minimum length of 35 characters • The IP message does not have a DLMS protocol envelope. The message is an ASCII string
IP_message_target_address	A configurable attribute that defines the address of the receiver of the IP message, which is send after establishing PDP context, This attribute will contain: • A fully qualified domain name (minimal 35 characters or normal IP address)

	• ~~A Port number (the port number is different from the port number used for the DLMS/COSEM protocol because the IP message does not contain the DLMS/COSEM protocol)~~
GPRS_operation_mode	Enum ::= {always_on == 1, triggered == 2)
~~PDP_context_reconnect_count~~	~~The number times that the meter will try to setup a PDP context after a connection has prematurely ended or the initial attempt to set up a PDP context fails.~~
~~PDP_context_reconnect_interval~~	~~The initial time interval in which the meter, at a random moment, will try to re-establish the PDP context.~~
~~PDP_context_reconnect_multiplyer~~	~~The factor by which the interval between tries will be multiplied after each unsuccessful attempt to re-establish the PDP context.~~
~~Control_mode_Ebreaker~~	~~All changes to this attribute will be copied to the control mode in the Disconnect control (OBIS code 0-0:96.3.10.255 attribute 4).~~

Flags

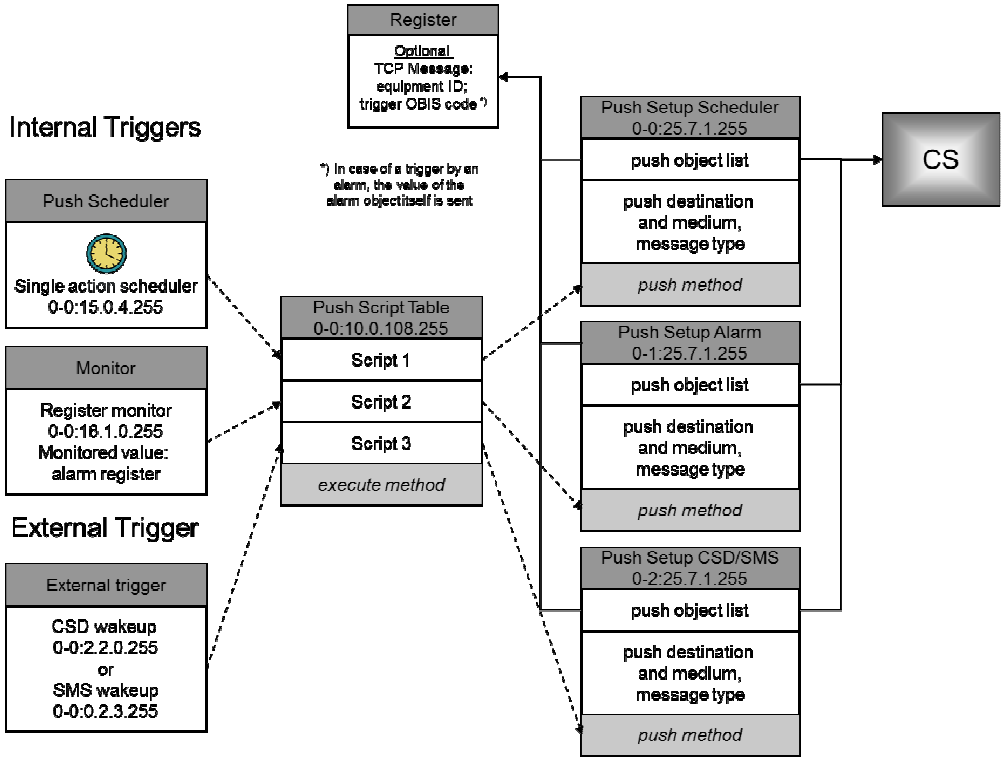
discover_on_open_cover (bit 0)	Indicates whether the M-Bus discovery process (see P2 companion standard) is started when the cover of the M-Bus connections on the electricity meter is removed.
discover_on_power_on (bit 1)	Indicates whether the M-Bus discovery process (see P2 companion standard) is started when the power to the electricity meter is switched on.
dynamic_mbus_address (bit 2)	Indicates whether the M-Bus device should use dynamic or static addressing. Dynamic addressing entails that the primary address of the M-Bus device is reset to 0 if the device is decommissioned.
~~send_comm_notification (bit 3)~~	~~A configurable attribute that indicates whether the device shall send an IP message after establishing a PDP context.~~
~~send_power_up_notification (bit 4)~~	~~Indicates whether the Electricity meter issues an alarm to the CS after the device is powered on. The device automatically connects to the network and sends the message after a successful logon if send_comm_notification == true.~~
P0_enable (bit **3**~~5~~)	Indicates whether communication via P0 is enabled or not.
~~HLS_3_4_on_p3_enable (bit 4~~**6**~~)~~	~~Indicates whether authentication via HLS method 3 and 4 is enabled on P3.~~
HLS_3_on_P3_enable (bit 4)	**Indicates whether authentication via HLS method 3 is enabled on P3 (disabled == 0, enabled ==1)**
HLS_4_on_P3_enable (bit 5)	**Indicates whether authentication via HLS method 4 is enabled on P3 (disabled == 0, enabled ==1)**
HLS_5_on_P3_enable (bit 6)	**Indicates whether authentication via HLS method 5 is enabled on P3 (disabled == 0, enabled ==1)**
HLS_3_on_P0_enable (bit 7)	**Indicates whether authentication via HLS method 3 is enabled on P0 (disabled == 0, enabled ==1)**

HLS_4_on_P0_enable (bit 8)	**Indicates whether authentication via HLS method 4 is enabled on P0 (disabled == 0, enabled ==1)**
HLS_5_on_P0_enable (bit 9)	**Indicates whether authentication via HLS method 5 is enabled on P0 (disabled == 0, enabled ==1)**

29 In section 8.6 changed text for attribute 5 to make it more clear:

Definable load profile (Class ID: 7)				**P**	**M**	**Pr**
1	Logical name	Octet-string	**0-1:94.31.6.255**		R	
2	buffer	array	The buffer must be filled monotonously, i.e. no irregular entries are allowed = exactly one entry per capture period		R	
3	capture_objects	Array	{8,0-0:1.0.0.255,2,0}; clock		RW	
4	capture_period	double-long-unsigned	86400 (daily)		RW	
5	sort_method	enum	3 (**1 =** unsorted (FIFO) or **3 =** sorted (largest))		RW	
6	sort_object	object definition	{8,0-0:1.0.0.255,2,0} (sorted by clock), other obis codes to be added during operation.		RW	
7	entries_in_use	double-long-unsigned			R	
8	profile_entries	double-long-unsigned	40 (40 days)		RW	
	Specific methods	***m/o***				
1	reset ()	m			X	
2	capture ()	m			X	
3	Reserved from previous versions					
4	Reserved from previous versions					

30 Replaced chapter 9 with the following:



The “Push setup” interface mainly contains a list of references of the object (attributes) to be pushed to the CS as well as the push destination and the communication medium to be used. A trigger (scheduler, monitor, wake-up call, etc.) calls a script entry in the push script table object which then invokes the push method of the related “Push setup” object. This method at the end handles the sending of the push data to the CS using the communication channel defined. The “Push setup” interface class also defines the communication time windows and the handling of retries for a push operation.

In the DSMR specification the “Push setup” interface is not used to send data to the CS but to initiate a connection. The Central System will retrieve the relevant data. However, as part of the communication initiation a message can be send to the CS which is defined in the push object list.

The following triggers can call a script entry in a push script table object are possible:

- Push scheduler.
  This is used for regular communication (see figure 1 in the DSMR GPRS Companion Standard).
- Monitor.
  This is used for alarming
- External trigger.
  This is used for CSD or SMS wake up mechanisms initiated from the CS

These triggers call a script entry in the Push Script Table object which then invokes the push method of the related “Push setup” object.

The push setup object takes care for connection management:

- starting the GPRS session
- setting up the PDP context
- optional sending the TCP message configured in the push object list containing the equipment identifier and the OBIS code of the trigger (see figure 2 in the DSMR GPRS Companion Standard)
- and preparing the TCP connection (open a server socket in listener mode) for the DLMS data transfer (initiated by the CS)

Note that the TCP connection, PDP context and GPRS attach are not destroyed after sending the objects defined in the push object list but only after all connections on the server socket are closed or timed out.

Some objects in this section use time windows (listening-window, calling-window) . The following settings shall be taken into account:

- An never active window is configured by setting both the start_time and end_time to 'not specified',
- An always active window is configured by setting the start_time to a valid date and time in the past and the end_time to 'not specified'

9.1 Triggers

9.1.1 Internal triggering by Push Scheduler

The meter can be triggered on a regular basis to setup a GPRS connection.. On the regular time, indicated by the Execution time (attribute 4) the associated script in attribute 2 is executed. This script is executed and the Push Setup object will trigger the connection management. In case this functionality is not used, the Execution time will be empty.

Single action scheduler (Class ID: 22, version=0)					**P**	**M**	**Pr**
1	logical_name	(static)	Octet string	0-0:15.0.4.255		R	
2	Executed_script	(static)	script	Contains the script which contains the push method invocation of the Push Setup Scheduler object.		R/W	
3	Type	(static)	enum	Default value 1 is used.		R/W	
4	Execution time	(static)	array	Contains the trigger moment.		R/W	
	Specific methods		***m/o***				

9.1.2 Internal triggering by Alarms

The meter can be triggered by an alarm to setup a GPRS connection. An event can cause an alarm (depending on the alarm filter). If an alarm (monitored value) is raised the associated script in attribute 4 is executed. This script is executed and the Push Setup object will trigger the connection management. In case this functionality is not used, the Actions array will be empty.

Register monitor (Class ID: 21, version=0)					**P**	**M**	**Pr**
1	logical_name	(static)	Octet string	0-0:16.1.0.255		R	
2	Thresholds	(static)	array	Value of the alarm register when there are no alarms (this is 0). Note that the alarm register contains only filtered events.		R	
3	Monitored_value	(static)	array	Obis code of the alarm register {1,0-0:97.98.0.255,2}		R	
4	Actions	(static)	array	The action_up contains the script which contains the push method invocation of the Push Setup Alarm object		R/W	
	Specific methods		***m/o***				

9.1.3 External trigger: wakeup using CSD call

The meter can be triggered by a Circuit Switched Data (CSD) call to setup a GPRS connection. The CSD call will be answered with a call reject. The listening window is active in case CSD wakeup is used. If the calling number is in the list_of_callers_and_actions, the associated script is executed. This script is executed and the Push Setup object will trigger the connection management.
See annex C for the Class definition.

Auto answer

Auto Answer (Class ID: 28, version=1)					**P**	**M**	**Pr**
1	logical_name	(static)	Octet string	0-0:2.2.0.255		R	
2	Mode	(static)	enum	Value: (200) Manufactory specific Mode: CSD call is used to triggger GPRS connection.		R	
3	Listening_window	(static)	array	In case CSD wakeup is not used: Listening_window is never active. In case CSD wakeup is used: Listening window is always active		R/W	
4	Status	(dyn.)	enum			R	
5	number_of_calls	(static)	unsigned	Default: (0) No limit		R	

6	number_of_rings (static)	nr_rings_type	Default: Call is answered (rejected) after 1 ring. This means: nr_rings_in_window=1 nr_rings_out_of_window=1.		R	
7	list_of_allowed_callers (static)	array	empty		R	
8	list_of_callers_and_actions (static)	array	Array of callers with associated scripts. The script contains the push method invocation of the Push Setup CSD/SMS object		R/W	
	Specific methods	***m/o***				

9.1.4 External trigger: wakeup using SMS message

The meter can be triggered by a SMS message to setup a GPRS connection. The listening window is always active in case SMS wakeup is used. The content of the SMS message is empty. If the calling number is in the list_of_allowed_senders_and_actions, the associated script is executed. This script is executed and the Push Setup object will trigger the connection management.
See annex C for the Class definition

Message Handler

message Handler (Class ID: 60, version=0)					**P**	**M**	**Pr**
1	logical_name	(static)	Octet string	0-0:0.2.3.255		R	
2	Listening_window	(static)	array	In case SMS wakeup is used: Listening_window is always active. In case SMS wakeup is not used: Listening window is never active.		R/W	
3	list_of_allowed_senders	(static)	array	empty		R	
4	list_of_senders_and_actions	(static)	array	Array of senders with associated scripts. The script contains the push method invocation of the Push Setup CSD/SMS object.		R/W	
	Specific methods		***m/o***				

9.2 Push script table

The Push script table holds scripts to activate the push operation. There are 3 entries in the script array. Every entry in the array of scripts calls the push method of one “Push setup” object instance.

Script table (Class ID: 9, version=0)					P	M	Pr
1	logical_name	(static)	Octet string	0-0:10.0.108.255		R	
2	Scripts	(static)	array	Contains the script which contains the push method invocation of the Push Setup object n: Script 1 invokes the push method of Push Setup scheduled object. Script 2 invokes the push method of Push Setup Alarm object. Script 3 invokes the push method of Push Setup CSD/SMS object.		R/W	
	Specific methods		***m/o***				
1	**Execute(data)**		**m**	Data contains the entry in the script table (1, 2 or 3)		X	

9.3 Push Setup

The push_object_list contains the information that is sent in the TCP message as mentioned in figure 2 of the GPRS Companion Standard.
If the push_object_list is empty, no TCP message is sent to the CS.

In the objectlist two objects are defined. The first object will point to a object that contains an octetstring. The second object points to the logical name of the triggering object (scheduler, register monitor or SMS/CSD messagehandler). As a TCP message the values are sent as ASCII string separated by a comma (,): for example:
XXXXX123456789012,0-0:15.0.4.255

The number of retries and repetitions_delay attributes determine if and when a retry is executed if the actions defined in the push object are not successful. This means that retries are started when one of the following actions were not successful:

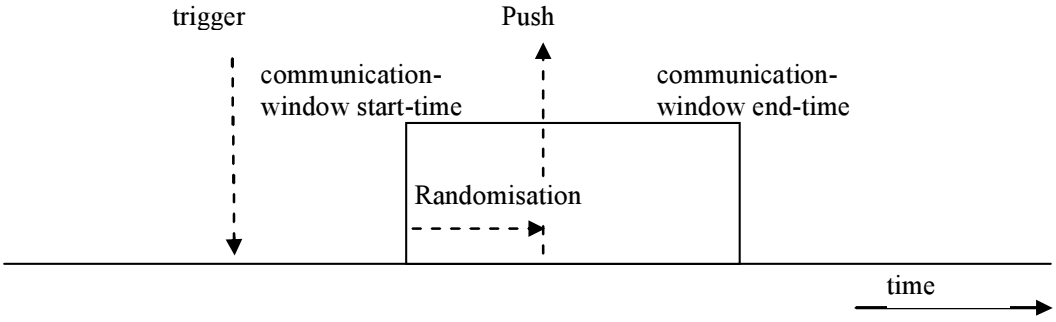
- GPRS attach
- PDP context activation
- sending push information (if configured by the Push object via the push_object_list)
  - TCP client socket activation
  - sending message
  - receiving ACK
- TCP server socket activation.

The repetitions and repetitions_delay attributes are used to guard the activation of PDP context and additionally guard the reception of an ACK of a send TCP message (if configured by the Push object via the push_object_list).

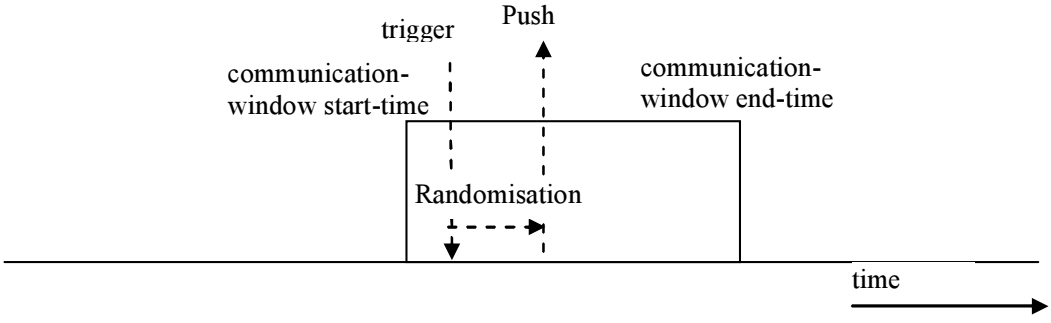
If multiple push objects are activated by multiple triggers at almost the same time then the first setup push method shall be finalized before the next push setup activation is executed. Finalized means in this case that communication on all the TCP sockets stopped and the meter closed PDP, TCP and GPRS connections and is 'idle' with regard to active communication sessions on P3.

When a trigger for a Push action is received before the start of a communication window, the randomisation time will be determined from the start time of the window. See figure below. When the trigger for a Push action is received after the start-time of the communication window, then the randomisation will be determined from the start of the trigger. When the determined randomisation time falls outside the communication-window, then a new randomisation time will be determined in the next communication window.
The reason for this behaviour is that it shall be prevented that many meters will start a Push action at the same time. In the above described way a power-up of many meters at the same time (after a power-down) during a communication-window will not result in many Push actions at the same time.

Trigger before communication-window:



Trigger within communication-window:



.

Push Setup Scheduler (Class ID: 40, version=0)				P	M	Pr
1	logical_name (static)	Octet string	0-0:25.9.1.255		R	
2	push_object_list (static)	array	This contains the TCP message that has to be sent: equipment ID and OBIS code of Push Scheduler {1,0-0:96.1.1.255,2,0 } {22,0-0:15.0.4.255,1,0} If the array is empty (array[0]), no TCP message is sent		RW	

3	send_destination_and_method (static)	structure	Defines the destination of the TCP message that is optionally sent to the CS
			
			send_destination_and_method ::= structure
			{
			service service_type
			destination octet-string,
			message message_type
			}
			
			“service” is TCP, value 0
			“Destination” contains the destination address (DNS) including port number
			“message” is not used and is 128 (i.e. first free manufacturer specific value: ASCII String).
					RW	

4	communication_window (static)	array	If the calling_window is empty (array[0]) this means the communication is possible without any limitations. If no starttime is specified then the communication will never start.		RW	
5	randomisation_start_interval (static)	long-unsigned	in seconds (default value = 0). An interval of 0 means no randomisation; i.e. immediate start at the beginning of the first communication window		RW	
6	Number_of_retries (static)	unsigned	The maximum number of retries in case of unsuccessful push attempts. A value of 0 means no repetitions		RW	
7	repetition_delay (static)	long-unsigned	The time delay expressed in seconds until an unsuccessful		RW	

			push attempt can be repeated			
	Specific methods	***m/o***				
1	push (data)	**m**			X	

Push Setup Alarm (Class ID: 40, version=0)				**P**	**M**	**Pr**
1	logical_name (static)	Octet string	0-1:25.9.1.255		R	
2	push_object_list (static)	array	This contains the TCP message that has to be sent: equipment ID and the value of the Alarm object {1,0-0:96.1.1.255,2,0} {1,0-0:97.98.0.255,2,0} If the array is empty (array[0]), no TCP message is sent		RW	

3	send_destination_and_method (static)	structure	Defines the destination of the TCP message that is optionally sent to the CS
			
			send_destination_and_method ::= structure
			{
			service service_type
			destination octet-string,
			message message_type
			}
			“service” is TCP, value 0
			“Destination” contains the destination address (DNS) including port number
			“message” is not used and is 128 (i.e. first free manufacturer specific value: ASCII String).
			
			When “destination” is empty(= octet-string[0]) then the function is disabled.
					RW	

4	commiunication_window (static)	array	If the calling_window is empty (array[0]) this means the communication is possible without any limitations. If no starttime is specified then the communication will never start.		RW	
5	randomisation_start_interval (static)	long-unsigned	in seconds (default value = 0). An interval of 0 means no randomisation; i.e. immediate start		RW	

			at the beginning of the first communication window			
6	number of retries (static)	unsigned	The maximum number of retries in case of unsuccessful push attempts. A value of 0 means no repetitions		RW	
7	repetition_delay (static)	long-unsigned	The time delay expressed in seconds until an unsuccessful push attempt can be repeated		RW	
	Specific methods	***m/o***				
1	push (data)	**m**			X	

Push Setup CSD/SMS (Class ID: 40, version=0)				**P**	**M**	**Pr**
1	logical_name (static)	Octet string	0-2:25.9.1.255		R	
2	push_object_list (static)	array	This contains the TCP message that has to be sent: equipment ID and OBIS code of CSD wakeup (auto answer) or {1,0-0:96.1.1.255,2,0} {28,0-0:2.2.0.255,1,0} or equipment ID and OBIS code of SMS wake (message handler): {1,0-0:96.1.1.255,2,0} {60,0-0:0.2.3.255,1,0} If the array is empty (array[0]), no TCP message is sent		RW	

3	send_destination_and_method (static)	structure	Defines the destination of the TCP message that is optionally sent to the CS
			
			send_destination_and_method ::= structure
			{
			service service_type
			destination octet-string,
			message message_type
			}
			“service” is TCP, value 0
			“Destination” contains the destination address (DNS) including port number
			“message” is not used and is 128 (i.e. first free manufacturer specific value: ASCII String).
					RW	

			When “destination” is empty(= octet-string[0]) then the function is disabled.			
4	communication_window (static)	array	If the calling_window is empty (array[0]) this means the communication is possible without any limitations. If no starttime is specified then the communication will never start.		RW	
5	randomisation_start_interval (static)	long-unsigned	in seconds (default value = 0). An interval of 0 means no randomisation; i.e. immediate start at the beginning of the first communication window		RW	
6	number_of_retries (static)	unsigned	The maximum number of retries in case of unsuccessful push attempts. A value of 0 means no repetitions		RW	
7	repetition_delay (static)	long-unsigned	The time delay expressed in seconds until an unsuccessful push attempt can be repeated		RW	
	Specific methods	***m/o***				
1	push (data)	**m**			X	

31 Updated Annex B. See end of document.

6. IN DSMR V4.0.4 GPRS CHANGES

This section lists all the changes incorporated in the Dutch Smart Meter Requirements v4.0.4 GPRS companion standard.

1. In section 2.2 changed requirement 4.2.7. to:

Description	If the communication unit receives a GSM call or SMS messages during a GPRS connection with active PDP context it shall proceed with the GPRS connection.
Rationale	In a normal network speech has priority over a data connection. In this ~~case~~ **metering network** the data connection shall have priority~~. Actions defined for the communication unit when receiving a GSM call or SMS message when no GPRS connection is active can be neglected.~~
Fit criterion	If the communication unit receives a GSM **voice** call ~~or SMS messages~~ during a GPRS connection with active PDP context it shall proceed with the GPRS connection and ~~the no further actions related to the GSM call or SMS message~~ **no actions related to the voice call** are executed.

History	18 Nov 2008	Origin	TST	Port	P3	Applicable	Comm. Unit

2. In section 3.1 added the following line of text :

- 'Always on' communication: Meter continuously has PDP context in the mobile network. There can be data exchange 'at any time'.
- Internal trigger (no continuous PDP context): Meter must be 'woken up' before data exchange over IP can take place. An internal trigger is generated by the meter itself using an internal (randomized) clock or by any event for which the meter is configured to wake up. After being woken up meters have PDP context. (See figure 3 and 4)
- External trigger (no continuous PDP context): Meter must be 'woken up' before data exchange over IP can take place. Triggers are coordinated by the CS. External triggers can be CSD (or voice call if secure), SMS or Network Initiated PDP context. After being woken up meters have PDP context. (See figure 1 and 2)

In all wake-up scenario's the meter will wait for TCP establishment from the CS during a configurable time (inactivity_time_out of the TCP-UDP Setup (ClassID 41)).

3. In section 3.1 added the following figures and explanation before requirement DSMR-G4.3.1

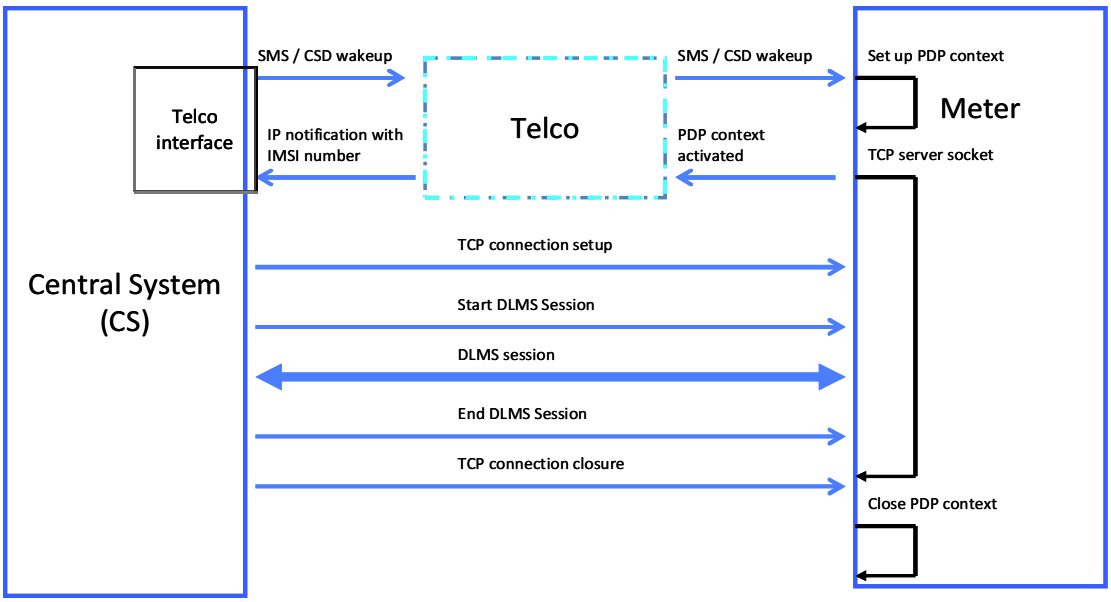


Figure 1 : Wake Up Mechanism 1

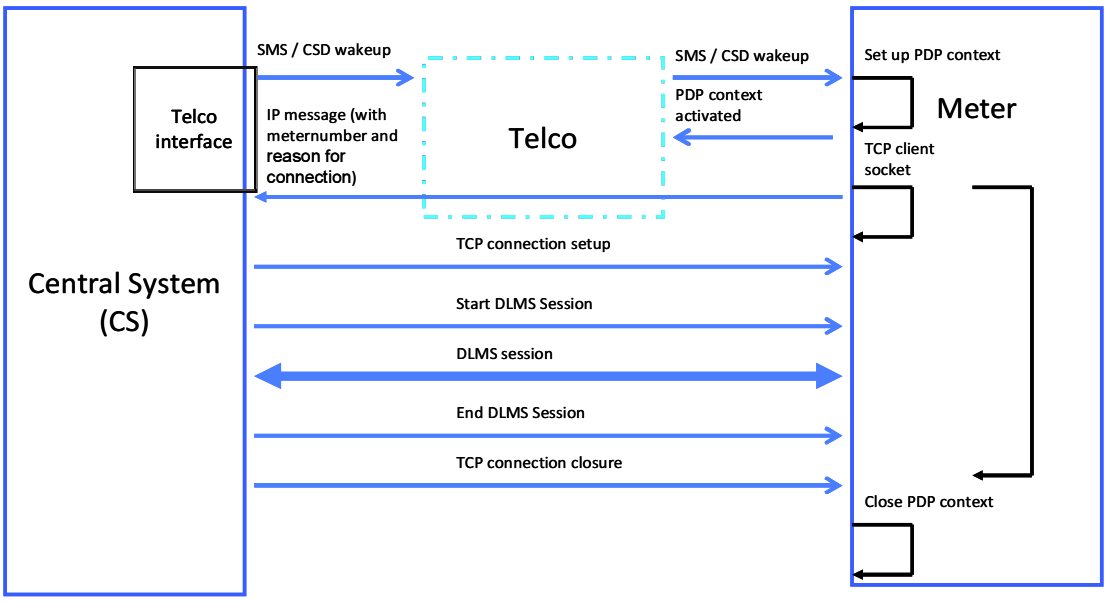


Figure 2: Wake Up Mechanism 2

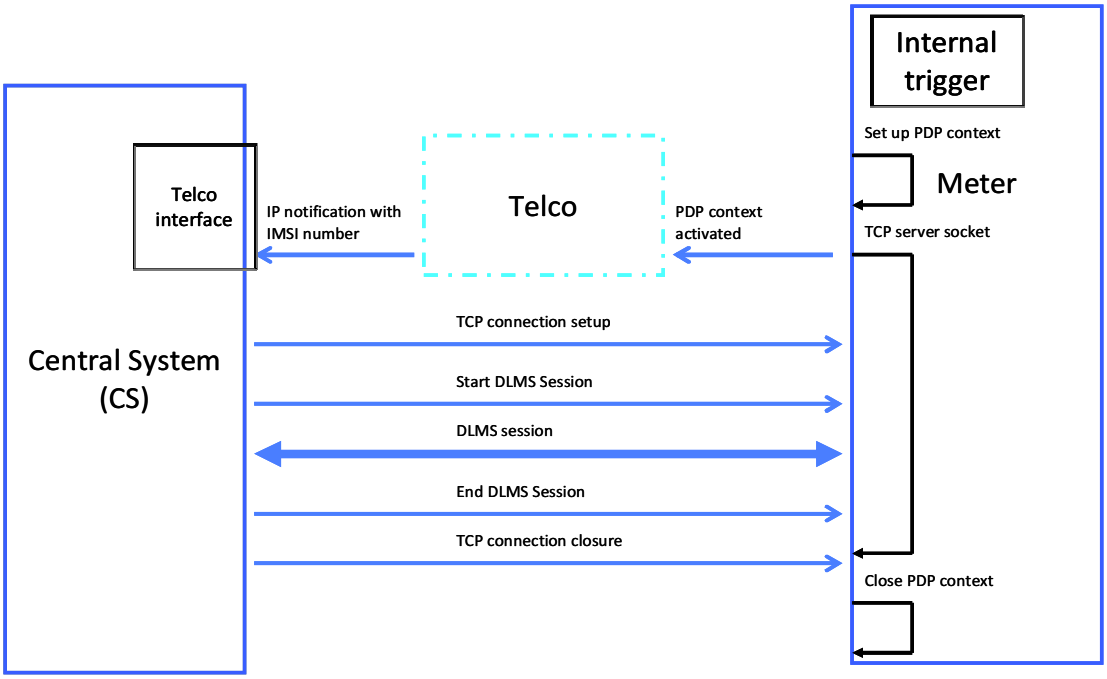


Figure 3: Meter initiated wake-up mechanism 3

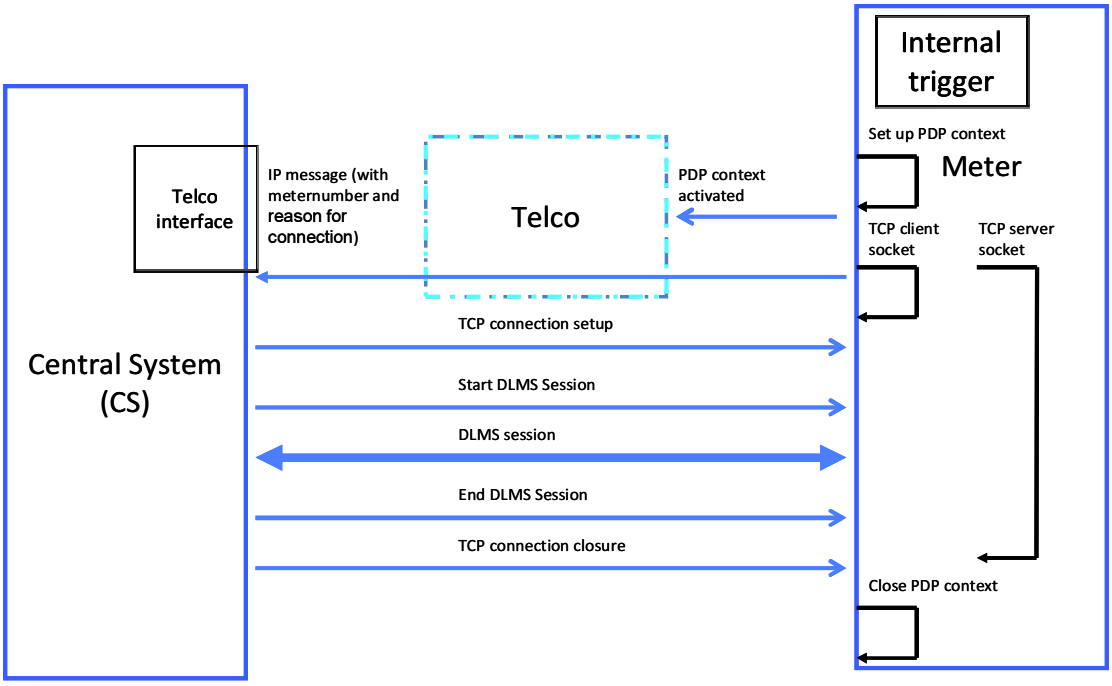


Figure 4: Meter initiated wake-up mechanism 4

Figure 2 and 5 show two TCP sockets.

1. Client socket to send the optional TCP message.
2. Server socket to listen to the incoming DLMS TCP connection.

The Server socket shall be opened at the same time the client socket is opened to prevent missing TCP connections.

Every connection from figure 2 thru 5 has the following steps:

1. Setup of PDP context
2. Setup of TCP server connection and optional:
   - 2a. Setup of TCP client connection
   - 2b. Send TCP message
   - 2c. Closure of the TCP client connection
3. Start DLMS Session
4. DLMS Message exchange
5. End DLMS Session
6. Closure of TCP server connection
7. Closure of PDP context

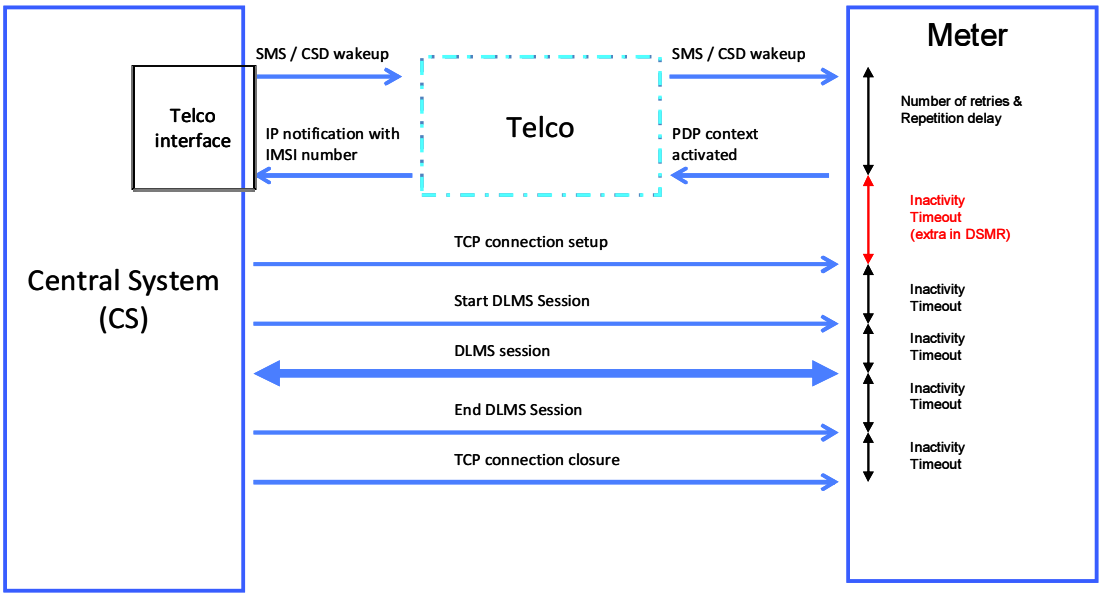


Figure 5: Inactivity time out for Mechanism 1 and 3

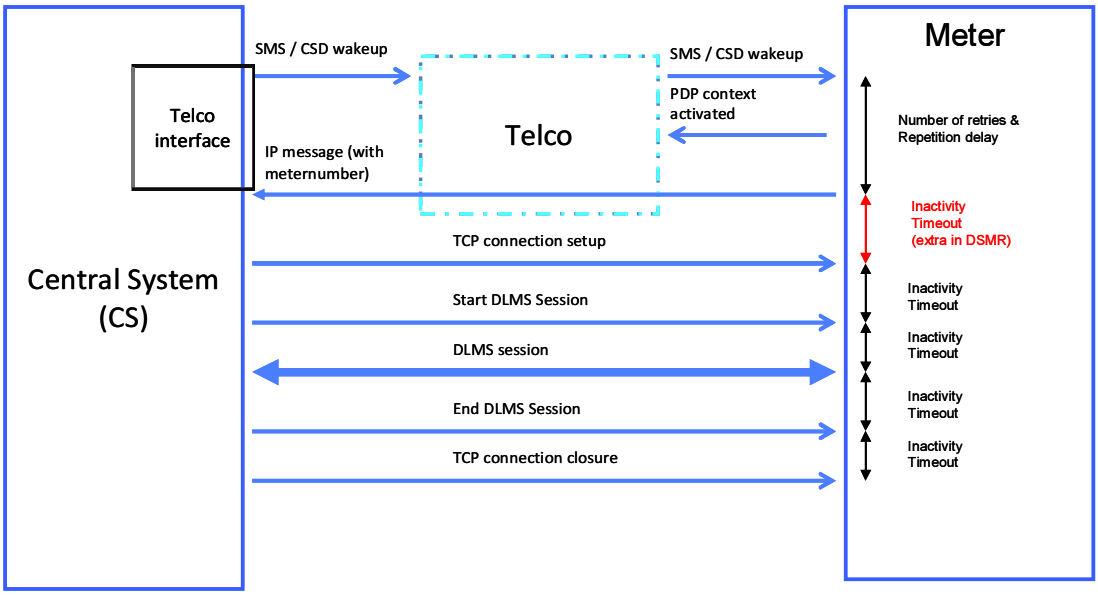


Figure 6: Inactivity time out for Mechanism 2 and 4

The DLMS specifies that the inactivity timeout is used to check whether a frame is received on a TCP connection. In DSMR we also use this time-out to check whether a TCP connection is set up for DLMS communication.

Annex B: Standard Readout Object List

STANDARD READOUT OBJECT LISTS and Display Definitions		Auto scroll mode			Manual scroll mode			Testmode			During M-bus installation mode of E meter		P1 output
		Display in Auto scroll (5 seconds per Obis code)			Manual scroll of the display. Return to auto scroll after 30 seconds not pushing a button.			Manual scroll of the display. Only available in case of removed terminal cover. Return to auto scroll only after installed terminal cover.					Continuously when request line is activated
		General display read out Numeric row 0-0:21.0.1.255	General display read out Alpha-numeric row 0-0:21.0.1.255	Symbols / other parts of the display	Alternate display read out Numeric row 0-0:21.0.2.255	Alternate display read out Alpha-numeric row 0-0:21.0.2.255	Symbols / other parts of the display	Service display read out Numeric row 0-0:21.0.3.255	Service display read out Alpha-numeric row 0-0:21.0.3.255	Symbols / other parts of the display	Numeric row	Symbols / other parts of the display	General local port read out 0-0:21.0.0.255
OBIS Code	**Description**												
n.a.	Display test	1	1	1	1	1	1						n.a.
n.a.	Manufacturer specific header												1
1-3:0.2.8.255	Version information				2								2
0-0:1.0.0.255	Actual date and time							1					3
0-0:96.1.1.255	Equipment identifier												4
1-0:1.8.1.255	Positive active energy Tariff 1 (A+)	2		+T01	3		+T01 and kWh symbol	2		+T01 and kWh symbol			5
1-0:1.8.2.255	Positive active energy Tariff 2 (A+)	3		+T02	4		+T02 and kWh symbol	3		+T02 and kWh symbol			6
1-0:2.8.1.255	Negative active energy Tariff 1 (A-)	4		-T01	5		-T01 and kWh symbol	4		-T01 and kWh symbol			7
1-0:2.8.2.255	Negative active energy Tariff 2 (A-)	5		-T02	6		-T02 and kWh symbol	5		-T02 and kWh symbol			8
0-0:96.14.0.255	Tariff indicator Electricity (actual tariff information)			T01 or T02			T01 or T02			T01 or T02			9
1-0:1.7.0.255	Instantaneous Active Power (A+, sum of all phases)												10
1-0:2.7.0.255	Instantaneous Active Power (A-, sum of all phases)												11
1-0:15.7.0.255	Instantaneous Active Power (abs(QI+QIV)+abs(QII+QIII)) (see note 5)		2,3,4,5	-P <- or -> +P and KW symbol		2,3,4,5	-P <- or -> +P and kW symbol		2,3,4,5	-P <- or -> +P and kW symbol			
0-1:24.1.0.255	Device type Channel 1 (formatted as text, eg. "Gas", "Water", "Warmte", "Koude", "Elektr.")					7a			7a				
0-1:24.1.0.255	Mbus Client Channel 1 serial number (see note 2)				8			7					
0-2:24.1.0.255	Device type Channel 2 (formatted as text)					8a			8a				
0-2:24.1.0.255	Mbus Client Channel 2 serial number (see note 2)				9			8					

0-3:24.1.0.255	Device type Channel 3 (formatted as text)					9a			9a				
0-3:24.1.0.255	Mbus Client Channel 3 serial number (see note 2)				10			9					
0-4:24.1.0.255	Device type Channel 4 (formatted as text)					10a			10a				
0-4:24.1.0.255	Mbus Client Channel 4 serial number (see note 2)				11			10					
1-0:0.2.0.255	Active FW Ver Core							11	0.0.2.0				
1-1:0.2.0.255	Active FW Ver Module							12	1.0.2.0				
1-0:21.7.0.255	Instantaneous Active Power L1 (A+)							13	0.21.7.0	kW symbol			
1-0:22.7.0.255	Instantaneous Active Power L1 (A-)							14	0.22.7.0	kW symbol			
1-0:41.7.0.255	Instantaneous Active Power L2 (A+) (see note 1)							15	0.41.7.0	kW symbol			
1-0:42.7.0.255	Instantaneous Active Power L2 (A-) (see note 1)							16	0.42.7.0	kW symbol			
1-0:61.7.0.255	Instantaneous Active Power L3 (A+) (see note 1)							17	0.61.7.0	kW symbol			
1-0:62.7.0.255	Instantaneous Active Power L3 (A-) (see note 1)							18	0.62.7.0	kW symbol			
0-0:17.0.0.255	Active Threshold			LMT flag (see note 4)	7 (see note 4)		LMT flag (see note 4)		6 (see note 4)	LMT flag (see note 4)			12
0-0:96.3.10.255	Switch position electricity			Open or closed breaker symbol			Open or closed breaker symbol			Open or closed breaker symbol			13
0-0:96.7.21.255	Number of power failures in any phase												14
0-0:96.7.9.255	Number of long power failures in any phase												15
1-0:99.97.0.255	Power failure event log												16
1-0:32.32.0.255	Number of voltage sags in phase L1												17
1-0:52.32.0.255	Number of voltage sags in phase L2 (see note 1)												18
1-0:72.32.0.255	Number of voltage sags in phase L3 (see note 1)												19
1-0:32.36.0.255	Number of voltage swells in phase L1												20
1-0:52.36.0.255	Number of voltage swells in phase L2 (see note 1)												21
1-0:72.36.0.255	Number of voltage swells in phase L3 (see note 1)												21
0-0:96.13.1.255	Text message 8		6 (see note 6)										22
0-0:96.13.0.255	Text message 1024												23
0-1:24.1.0.255	M-Bus Client Channel 1 Device type (see note 2)												24

0-1:96.1.0.255	M-Bus Device ID 1 Channel 1 - Equipment Identifier (see note 2)												25
0-1:24.2.1.255	M-Bus Capture Time Channel 1												26
0-1:24.4.0.255	M-Bus Client Channel 1 Valve/Switch position (see note 3)												27
0-1:24.2.1.255	M-Bus Value 1 Channel 1												28
0-2:24.1.0.255	M-Bus Client Channel 2 Device type (see note 2)												29
0-2:96.1.0.255	M-Bus Device ID 1 Channel 2 - Equipment Identifier (see note 2)												30
0-2:24.2.1.255	M-Bus Capture Time Channel 2												31
0-2:24.2.1.255	M-bus Value 1 Channel 2												32
0-2:24.4.0.255	M-Bus Client Channel 2 Valve/Switch position (see note 3)												33
0-3:24.1.0.255	M-Bus Client Channel 3 Device type (see note 2)												34
0-3:96.1.0.255	M-Bus Device ID 1 Channel 3 - Equipment Identifier (see note 2)												35
0-3:24.2.1.255	M-Bus Capture Time Channel 3												36
0-3:24.2.1.255	M-bus Value 1 Channel 3												37
0-3:24.4.0.255	M-Bus Client Channel 3 Valve/Switch position (see note 3)												38
0-4:24.1.0.255	M-Bus Client Channel 4 Device type (see note 2)												39
0-4:96.1.0.255	M-Bus Device ID 1 Channel 4 - Equipment identifier (see note 2)												40
0-4:24.2.1.255	M-Bus Capture Time Channel 3												41
0-4:24.2.1.255	M-bus Value 1 Channel 3												42
0-4:24.4.0.255	M-Bus Client Channel 4 Valve/Switch position (see note 3)												43
1-0:32.7.0.255	Instantaneous voltage L1			L1			L1			L1 (see note 7)		L1 (see note 7)	
1-0:52.7.0.255	Instantaneous voltage L2 (see note 1)			L2			L2			L2 (see note 7)		L2 (see note 7)	
1-0:72.7.0.255	Instantaneous voltage L3 (see note 1)			L3			L3			L3 (see note 7)		L3 (see note 7)	
0-1:94.31.4.255	GPRS Network information			GPRS signal strength symbol			GPRS signal strength symbol			GPRS signal strength symbols			
0-1:94.31.0.255	Administrative In/Out status: Administrative Off ("Administratief Uit")			AU flag			AU flag			AU flag			
0-1:94.31.0.255	Administrative In/Out status: Default ("Standaard")			STD flag			STD flag			STD flag			
0-1:94.31.0.255	Administrative In/Out status: Administrative On ("Administratief In")			AI flag			AI flag			AI flag			

	Registered to GSM Netwerk					NW flag			NW flag			NW flag				
0-0:97.97.0.255	Error register (only in case of a critical error)											FF flag				
	M-bus serial numbers (at least 7 digits) of all M-bus devices found												M-bus serial nr's in order (see note 8)			

Notes:

1) Polyphase meters only
2) Only if device is installed
3) Only if device is installed and the device is equipped with a valve or switch
4) Only if limiter is activated (threshold unequal to 999999W)
5) Value is allways positive. Direction of power can be determined by the arrows of the actual power on the display
6) Visible untill button is pressed or empty message is sent from CS to meter
7) Blinking in case of registration of negative active energy as an indication of POSSIBLE wrong connection of "phase in" and "phase out" for that phase
8) Wireless M-bus devices: By pushing the button, the next serial number of a found M-bus device is shown. Selecting a device is done by pressing the button at least two seconds
   Wired M-bus devices: No selection necessary by means of a push button