

Lastenheft

MUC – Multi Utility Communication

Version 1.01 - 04. Juli 2011 [Arbeitsfassung]



© Forum Netztechnik/Netzbetrieb im VDE (FNN)

Bismarckstraße 33, 10625 Berlin

Telefon: +49 (0)30 3838687 0

Fax: +49 (0)30 3838687 7

E-Mail: fnn@vde.com

Internet: www.vde.com/fnn

Ausgabe: **04. Juli 2011**

[Arbeitsfassung]

Historie

Version	Datum	Kommentar	Verantwortlich
0.0x	April 2007	1. Initiale Version	M. Wisy
0.02	09.05.2007	1. Umbenennung von „Smart Metering and Control“ zu „Multi Utility Communication“. 2. Überarbeitung des Kapitels „Bezug“ sowie Anpassung des Strukturbilds zur Referenzplattform.	Projektgruppe
0.03	Juli 2007	1. Kapitel Begriffslexikon eingeführt. 2. Kapitel Einleitung an MUC-Eckpunktepapier angepasst. 3. Kapitel zur Organisation entfernt. 4. Kapitel mit Eckpunkten ergänzt. 5. Kapitel Systemumfeld überarbeitet. 6. Kapitel Schnittstellen mit erstem Inhalt befüllt.	M. Wisy
0.04	30.07.2007	1. Integration der Anforderungsmatrix. 2. Kapitel zu den Einsatzszenarien (Struktur der Liegenschaften) entfernt. 3. Kapitel mit allgemeinen Anforderungen (basierend auf dem Positionspapier der RE) eingefügt.	Projektgruppe
0.05	08-09 / 2007	1. Einarbeitung der Fragen aus Arbeiten zu Open Metering. 2. Erweiterung der Vorgaben zum MUC – Metering Unified Container (siehe Kapitel Kodierung).	M. Wisy
0.06	24.09.2007	1. Beantwortung des Fragenkatalogs	Projektgruppe
0.07	09-10 / 2007	1. Übernahme der Hinweise aus AG-2 der WG Open Metering zu den vorgeschlagenen SML-Messages. 2. Erweiterung der Vorgaben zu MUC-Kodierung (Identifizier, Signatur). 3. Ergänzung des Kapitels zur Anbindung von eHZ per SML an den MUC-Controller. 4. Ergänzung des Kapitels zur Spezifikation der Kommunikation per SML / IP-Telemetrie über die Weitverkehrs-schnittstelle.	M. Wisy
0.08	06.11.2007	1. Konsolidierung des vorgelegten Entwurfs. 2. Übernahme der Vorschläge zur funkbasierten Kommunikation. 3. Restrukturierung der Kapitel-Anordnung mit dem Ziel einer klaren Trennung in / Zuordnung auf die unterschiedlichen Bereiche „Funk“, „Draht“ und „PLC“. 4. Ergänzungen von Detailforderungen an den MUC-Controller.	Projektgruppe
0.09	19.11.2007	1. Aktualisierung der Anhänge mit den Vorschlägen zur funkbasierten Kommunikation.	M. Wisy
0.10	21.01.2008	1. Kapitel 3.2 zur Migrationsphase wurde ergänzt. 2. Aktualisierung der Anhänge mit den Vorschlägen zur funkbasierten Kommunikation.	Projektgruppe
0.11	09.03.2008	1. In Bild 2 wurde bei Variante C das Applikations-Protokoll „SML“ ergänzt.	M. Wisy

		<ol style="list-style-type: none"> Zur Datenbeschaffung per SML von mehreren Datenpunkten wurde das Kommando SML_GetMucList' eingeführt (siehe Kapitel 5.3.1.1 ff.). Die SML-Nachrichten zur Tarifumschaltung wurden definiert (siehe Kapitel 5.3.3.2). Die Festlegungen zum Protokollstapel für die Nahfunkstrecke wurden getroffen (siehe Kapitel 5.4.5). Als Alternative zu ‚Mini-USB‘ bei eHZ wurde RJ10 nach SyM² vorgeschlagen (siehe Kapitel 5.4.7.2) 	
0.12	18.03.2008	<ol style="list-style-type: none"> Änderung des Titels von ‚MUC-Referenzplattform‘ zu ‚MUC-Lastenheft‘. Ergänzung der Festlegungen zur Verwendung von S-Mode und / oder T-Mode bei Nahfunk. Festlegung der direkten Kopplung zwischen eHZ und MUC via CL. Anpassung der Verbindung zu eHZ auf eHZ nach Lastenheft 1.03 (dort wird festgelegt werden, dass diese eHZ nur SML und kein IEC-1107 mehr verwenden). Einführung des Kapitels zur Vermaschung von MUC. Einführung des Kapitels zur Beschreibung simpler Anwendungsfälle. Entfernung der Anhänge zu M-Bus Datenpunkten. 	Projektgruppe (Arbeits- ergebnisse der Besprechung vom 18.03.2008)
0.13	14.04.2008	<ol style="list-style-type: none"> Umstellung des Vorschlags zur direkten Kopplung zwischen eHZ und MUC von CL auf eine reduzierte RS232. Zusammenführung der beiden SML-Message-Varianten ‚SML_GetMuc‘ und ‚SML_GetMucList‘ zu einer einzigen, allgemein formulierten SML-Message ‚SML_GetList‘ (siehe Kapitel 5.3.1.1). 	Übernahme von Hinweisen aus dem VNB-Kreis
0.14	14.05.2008	<ol style="list-style-type: none"> Umstellung auf BDEW-Layout. Berichtigung, Lastgangmessung Gas, s. Tab. 2. 	M. Wisy
0.15	02.06.2008	<ol style="list-style-type: none"> Integration des Vorschlags zur adaptiven Erweiterung von OBIS-Kennzahlen (siehe Kapitel 3.2.1). Präzisierung der Anbindung von eHZ an MUC per RS232 (siehe Kapitel 5.4.7.2). 	Übernahme von Hinweisen aus dem VNB-Kreis
0.16	09.06.2008	<ol style="list-style-type: none"> Präzisierung der zwingend für einen MUC-Controller geforderten Schnittstellen. Einführung der lokalen Erweiterungsschnittstelle in der Ausführungsvariante Ethernet mit PoE. Überarbeitung offener Punkte. 	Projektgruppe (Arbeits- ergebnisse der Besprechung vom 09.06.2008)
0.17	13.06.2008	<ol style="list-style-type: none"> Definition der für einen MUC-Controller maximal zulässigen Breite (siehe Kapitel 4.1.1.4). Redaktionelle Überarbeitung der Liste herangezogener Normen. 	M. Wisy
0.60	18.09.2008	<ol style="list-style-type: none"> Nach Übernahme in die Projektgruppe MUC des FNN freigegebene Arbeitsversion. 	Projektgruppe
0.61	05.11.2008	<ol style="list-style-type: none"> Integration generisch formulierter Prozesse zu Installation und Aktualisierung von MUC-Controllern und Sensoren / Aktoren. 	Projektgruppe

0.62	01.12.2008	<ol style="list-style-type: none"> Ergänzung der Datenstrukturen zur Parametrierung / Ansteuerung des MUC-Controllers. Die übergangsweise im MUC-Lastenheft getroffenen Festlegungen zu SML wurden in die Version 1.03 zur SML-Spezifikation verschoben. 	Projektgruppe
0.63	16.12.2008	<ol style="list-style-type: none"> Definition zum Statuswort eingeführt. 	M. Wisy
0.64	16.01.2009	<ol style="list-style-type: none"> Datenstrukturen zum Firmware-Upload bearbeitet. Konzept zum Datenspiegel von Sensoren eingeführt. 	Projektgruppe
0.65	28.01.2009	<ol style="list-style-type: none"> Festlegung ergänzender Definitionen zu Benutzernamen, Passwörtern und Zugriffsrechten. Abschluss der Arbeiten an den Datenstrukturen (bis auf die Vergabe der benötigten Kennzahlen). Einarbeitung redaktioneller Hinweise. 	Projektgruppe
0.66	29.01.2009	<ol style="list-style-type: none"> Redaktionelle Überarbeitung. 	M. Wisy
0.67	31.03.2009	<ol style="list-style-type: none"> Adressierung auf Verwendung der IMEI erweitert. 	M. Wisy
0.98	11.05.2009	<ol style="list-style-type: none"> Redaktionelle Ergänzung der fehlenden Kennzahlen zu OBIS-T. In diesem Zusammenhang wurden bei einigen Datenstrukturen strukturierende Listen ergänzt. Einarbeitung der im Zuge des abschließenden Freigabeprozesses eingegangenen Hinweise. Die Kodierung von IP-Adressen für die Varianten IPv4 und IPv6 wurde präzisiert. Datenstruktur zur Abfrage der Zeitinformation ergänzt Kommando zum Entfernen der Datenspiegel eingeführt. Datenstruktur zur Parametrierung der Kundenschnittstelle aufgenommen. 	M. Wisy
0.99	25.06.2009	<ol style="list-style-type: none"> Integration der zwischenzeitlich eingegangenen Rückmeldungen. Tabelle zur Kodierung der Geräteidentifikation erweitert. 	M. Wisy
0.99a	26.06.2009	<ol style="list-style-type: none"> Fehlerkorrektur in der Tabelle zur Kodierung der Geräteidentifikation. 	M. Wisy
1.0	28.07.2009	<ol style="list-style-type: none"> Von FNN-Projektgruppe „eHZ/MUC“ freigegebene Version 	Projektgruppe
1.01	26.01.2010	<p>Beginn der Integration notwendiger Anpassungen / Ergänzungen in Folge neuer Anforderungen aus dem Markt.</p> <ul style="list-style-type: none"> Datenstrukturen und Regeln zur Benutzung von Zeitstempel auf Basis der gesetzlichen Zeit sowie zur Abbildung des Gas-Tages eingeführt. 	Projektgruppe
1.01	11.05.2010	<ul style="list-style-type: none"> Erweiterung der ein-eindeutigen Adressierung auf den Einsatz von Sub-Adressen (siehe Tab. 7). 	Projektgruppe
1.01	03.06.2010	<ul style="list-style-type: none"> Ergänzung der Datenstrukturen zur Notation von Aktionsschaltprogrammen. 	Projektgruppe
1.01	22.06.2010	<ul style="list-style-type: none"> Erweiterung der Referenzzeitpunkte bei Aktionsschaltprogrammen um die Variante ‚Erster Werktag im Montag‘. 	Projektgruppe
1.01	05.07.2010	<ul style="list-style-type: none"> Erweiterung der Datenstrukturen, hauptsächlich mit dem Ziel der EDL40-Integration. 	Projektgruppe
1.01	28.09.2010	<ul style="list-style-type: none"> Klärung zum funktionalen Verhalten an der Kundenschnittstelle. Erweiterung der Attribute eines Logbuch-Eintrags. 	Projektgruppe

		<ul style="list-style-type: none"> • Überbeitung der Datenstrukturen zu PLC. • Ergänzung einer Liste zu allgemeinen Schnittstellen-Eigenschaften. • Ergänzung der Parameter zur EDL-Zähler-Schnittstelle. • Präzisierung zur konstruktiven Festlegung für die Variante DIN-Schienen-Montage. 	
1.01	26.10.2010	<ul style="list-style-type: none"> • Ergänzung der Informationen im Betriebslogbuch. • Überbeitung der Datenstrukturen zu PLC. 	Projektgruppe
1.01	03.11.2010	<ul style="list-style-type: none"> • Überbeitung der Datenstrukturen zur Dateiübertragung per Broadcast. 	Projektgruppe
1.01	25.11.2010	<ul style="list-style-type: none"> • Anpassung kleinerer Details an den Datenstrukturen zu den Aktionsschaltprogrammen. • Ergänzung mit Datenstrukturen zur Beschreibung von Parametrier-Schablonen („Templates“). 	Projektgruppe
1.01	01.12.2010	<ul style="list-style-type: none"> • Ergänzung weiterer Erläuterungen zum Einsatz der Datenstrukturen für die Dateiübertragung per Broadcast (siehe Bild 8 und Bild 9). • Erweiterung der Datenstrukturen zur Beschreibung von Parametrier-Schablonen („Templates“). 	Projektgruppe
1.01	06.12.2010	<ul style="list-style-type: none"> • Ergänzung fehlender Kennzahlen zum Aktionsschaltprogramm sowie zu den Parametrier-Schablonen • Präzisierung im Kapitel zu SML hinsichtlich des Zugriffs auf Datenstrukturen • Datenstruktur zur Beschreibung überarbeitet 	Expertenteam
1.01	16.02.2011	<ul style="list-style-type: none"> • Präzisierung zum Aktionsschaltprogramm • Ergänzung der Kodierung zur Geräteidentifikation 	Expertenteam
1.01	03.03.2011	<ul style="list-style-type: none"> • Fehler in Tab. 47 behoben 	M. Wisy
1.01	09./10.03.11	<ul style="list-style-type: none"> • Anzahl der maximal zulässigen Elemente für die M-Bus-OBIS-Umsetzung von 300 auf 1000 geändert • Begriff „Herstellen der Betriebsbereitschaft“ definiert • Präzisierung zum Aktionsschaltprogramm • Parameter ergänzt, um das Zeitfenster bis zur erneuten Nachführung der Systemuhr einstellen zu können • Tab. 48 zur Beschreibung von Regeln erweitert • Kapitel „MUC und EDL21-Zähler im EDL40-System“ zur Beschreibung der Funktionsprozesse im EDL40-System als Vorschlag erarbeitet • Kapitel zu Status- und Parametrierung der Erweiterungsschnittstelle eingefügt. 	Expertenteam
1.01	12./13.05.11	<ul style="list-style-type: none"> • Kapitel „MUC und EDL21-Zähler im EDL40-System“ zur Beschreibung der Funktionsprozesse im EDL40-System vpm 09./10.03.11 angenommen • Handhabung von TCP-Sockets präzisiert. • Definitionen zur Gangreserve der Systemuhr und dem Verhalten bei Spannungsausfall nach Gangreserve geändert • Hysterese zur Auswertung von Schwellwerten bei Eingangsgrößen für das Aktionsschaltprogramm ergänzt • Datenstrukturen zum Tunneln von Fremdprotokollen 	Expertenteam

-
- Der Aufzeichnungs-Prozess zum Datensammler wurde präzisiert.
-
-

Inhaltsverzeichnis

I	Bildverzeichnis	10
II	Tabellenverzeichnis.....	11
III	Abkürzungsverzeichnis	13
IV	Normen und referenzierte Dokumente.....	14
V	Begriffe	15
1	Bezug.....	19
2	Eckpunkte	20
3	Systemumfeld.....	21
3.1	Allgemeine Anforderungen, Beschreibung des Soll-Zustandes	21
3.1.1	Zählerebene.....	21
3.1.2	Ebene MUC-Controller	23
3.1.3	Grundsätzliches	23
3.2	Migrationsphase	23
3.2.1	Festlegungen zur Variante B (Migrationsphase)	24
3.3	Anforderungsmatrix	26
3.4	Prozesse	30
4	Systemkomponenten.....	37
4.1	MUC-Controller	37
4.1.1	Grundstruktur zum MUC-Controller	37
4.1.1.1	Betriebssicherung	38
4.1.1.2	Zeitbasis / Zeitserver / Betriebssekundenzähler.....	38
4.1.1.3	Schnittstelle zum Endkunden	39
4.1.1.4	Konstruktive Festlegungen	40
5	Schnittstellen.....	41
5.1	Anforderungen an die Protokolle zur Anbindung der Sensorik / Aktorik.....	43
5.2	Vermaschung von Sensoren / MUC-Controllern.....	43
5.3	Anwendungsebene.....	44
5.3.1	MwC – Messwert-Container	44
5.3.1.1	MwC, Messwert-Container, Kodierung.....	45
5.3.2	MUC, optionale Signatur.....	45
5.3.2.1	Krypto-Algorithmus	45
5.3.2.2	Signaturbildung.....	45
5.3.3	MUC, Datenstrukturen	46
5.3.3.1	Datenstrukturen zur Zählerstandsabfrage / Abfrage von Datenlisten	46
5.3.3.2	Datenstrukturen zur Tarifumschaltung.....	46
5.3.3.3	Datenstrukturen zur Ansteuerung von Sensoren / Aktoren.....	46
5.3.3.4	Benutzerverwaltung / Berechtigungskonzept	46
5.4	Transportebene	48

5.4.1	Verwendung von TCP/IP bei MUC	48
5.4.2	Verwendung von SML bei MUC	49
5.4.3	Weitverkehrsschnittstelle	50
5.4.4	SML als Application-Layer für Nahfunk- oder PLC-Systeme in MUC-Anwendungen	51
5.4.5	MUC, Funkschnittstelle an der Mess-Einrichtung.....	51
5.4.5.1	Ergänzende Anforderungen für hybride Geräte	52
5.4.6	Vermaschtes Funknetz	52
5.4.7	MUC, Leitungsgebundene Schnittstelle, lokale Erweiterungsschnittstelle	52
5.4.7.1	MUC, Einsatz bei Mess-Einrichtungen mit M-Bus-Schnittstelle	52
5.4.7.2	MUC, Einsatz bei Mess-Einrichtungen mit Stromzählern und EDL-Zähler-Schnittstelle ..	52
6	Anwendungsfälle.....	54
6.1	Inbetriebnahmeprozess / Installationsprozess.....	54
7	Anhang	55
7.1	Statuswert	55
7.2	MUC, einheitlicher Adressierung.....	56
7.2.1	Data-Matrix-Code.....	59
7.3	MUC-Datenstrukturen zur Parametrierung	60
7.3.1	Datenstrukturen zur Parametrierung der MUC-Controller-Funktionen.....	60
7.3.1.1	Direkte Parameter.....	60
7.3.1.2	Datenstruktur zur Parametrierung der Sommerzeit- / Winterzeitwechsel	61
7.3.1.3	Datenstruktur zur Parametrierung der Rollen / Benutzerrechte	61
7.3.1.4	Datenstruktur zum Lesen/Setzen der Parameter für die Endkundenschnittstelle.....	63
7.3.1.5	Datenstruktur für dynamischen Eigenschaften der Endkundenschnittstelle	64
7.3.1.6	Datenstruktur zum Lesen/Setzen der Parameter für die Erweiterungsschnittstelle	64
7.3.1.7	Datenstruktur für dynamischen Eigenschaften der Endkundenschnittstelle	64
7.3.1.8	Datenstruktur allgemeiner Schnittstellen-Eigenschaften	64
7.3.1.9	Datenstruktur zur Abfrage des WAN Status	65
7.3.1.10	Datenstruktur zum Lesen/Setzen der WAN Parameter.....	66
7.3.1.11	Datenstruktur zum Lesen/Setzen der GSM Parameter	66
7.3.1.12	Datenstruktur zur Abfrage des IPT Status.....	67
7.3.1.13	Datenstruktur zur Lesen/Setzen der IPT Parameter	68
7.3.1.14	Datenstruktur zur Lesen/Setzen von AT-Hayes-Strings.....	69
7.3.1.15	Datenstruktur zum Lesen / Setzen der Provider-abhängigen GPRS-Parameter	70
7.3.1.16	Datenstruktur zum Lesen / Setzen der zulässigen GSM/GPRS-Betreiber.....	70
7.3.1.17	Datenstruktur zur Abfrage dynamischer GPRS/GSM-Betriebsparameter.....	71
7.3.1.18	Datenstruktur zum Lesen / Setzen der LAN/DSL-Parameter.....	72
7.3.1.19	Datenstruktur zur Abfrage dynamischer LAN/DSL- Betriebsparameter	73
7.3.1.20	Datenstruktur zum Lesen / Setzen der PLC-Parameter	74
7.3.1.21	Datenstruktur zur Abfrage dynamischer PLC- Betriebsparameter	75
7.3.1.22	Datenstruktur zum Lesen / Setzen der W-MBUS-Parameter.....	76
7.3.1.23	Datenstruktur zum Lesen des W-MBUS-Status	77
7.3.1.24	Datenstruktur zum Lesen / Setzen der EDL-Zähler-Schnittstellen-Parameter.....	78

7.3.1.25	Liste möglicher Push-Quellen.....	78
7.3.1.26	Datenstruktur zum Transport der Eigenschaften von Push-Vorgängen.....	79
7.3.1.27	Zeitverhalten periodischer Push-Vorgänge	80
7.3.2	Liste möglicher Push-Dienste	81
7.3.2.1	Datenstruktur zur Adressierung einer bestimmten Push-Quelle	82
7.3.2.2	Datenstruktur zum Transport der Installationsparameter	82
7.3.2.3	Datenstruktur zum Transport der Eigenschaften von Regelsätzen.....	83
7.3.2.4	Datenstruktur zum Transport der Eigenschaften von Saisonprogrammen	84
7.3.2.5	Datenstruktur zum Transport der Eigenschaften von Tagesprogrammen	85
7.3.2.6	Datenstruktur zum Transport der Eigenschaften von Regeln mit Aktionslisten	88
7.3.2.7	Datenstruktur zum Transport der Liste aller Aktionsbeschreibungen.....	88
7.3.2.8	Datenstruktur zum Transport der Eigenschaften von ‚Register On Off‘	89
7.3.2.9	Datenstruktur zum Transport der Eigenschaften von ‚Register intern‘	89
7.3.2.10	Datenstruktur zum Transport der Eigenschaften von ‚Register extern‘	89
7.3.2.11	Datenstruktur zur Aufzeichnung in einem Datenspiegel.....	90
7.3.2.12	Datenstruktur zum Transport der Eigenschaften von ‚Zustand ausgeben‘	91
7.3.2.13	Datenstruktur zum Transport der Eigenschaften von ‚Impuls ausgeben‘	91
7.3.2.14	Datenstruktur zum Transport der Eigenschaften von ‚SML-Datei ausgeben‘	92
7.3.2.15	Datenstruktur zum Transport der Liste aller ‚Regeln‘	93
7.3.2.16	Datenstruktur mit Liste der sichtbaren Sensoren / Aktoren.....	96
7.3.2.17	Datenstruktur mit Liste der aktiven Sensoren / Aktoren	97
7.3.2.18	Datenstruktur zum Transport neuer sichtbarer Sensoren / Aktoren.....	98
7.3.2.19	Datenstruktur zum Transport verschwundener, zuvor sichtbarer Sensoren / Aktoren.....	99
7.3.2.20	Datenstruktur zum Transport deaktivierter Sensoren / Aktoren	100
7.3.2.21	Datenstruktur zum Transport aktivierter Sensoren / Aktoren	101
7.3.2.22	Datenstruktur zum Ergänzen eines aktiven Sensors / Aktors	102
7.3.2.23	Datenstruktur zum Deaktivieren eines Sensors / Aktors	102
7.3.2.24	Datenstruktur zum Entfernen / Löschen eines Sensors / Aktors.....	102
7.3.2.25	Datenstruktur zum Auftrag ‚Kommando Reset ausführen‘	103
7.3.2.26	Datenstruktur zur Abfrage der Geräte-Identifikation.....	103
7.3.2.27	Datenstruktur zur Antwort der Geräte-Identifikation	103
7.3.2.28	Datenstruktur zum remote Firmware-/Datei-Download (Übertragung).....	105
7.3.2.29	Datenstruktur zum Auftrag ‚Firmware/Datei aktivieren‘	106
7.3.2.30	Auftrag/Antwort mit GET-Kontrollstruktur zur Statusabfrage des Firmware-/Datei-Downloads	108
7.3.2.31	Auftrag mit SET-Kontrollstruktur zum Start des Firmware-Datei-Downloads.....	108
7.3.2.32	Auftrag/Antwort zur Statusabfrage des Multicast-Firmware-Download.....	109
7.3.2.33	Auftrag/Antwort zur Datenübertragung des Multicast-Firmware-Download	110
7.3.2.34	Datenstruktur zur Abfrage dynamischer Multicast-Firmware-Download-Betriebsparameter	110
7.3.2.35	Datenstruktur zur Übertragung des Hersteller-spezifischen Ereignis-Logbuches.....	111
7.3.2.36	Datenstruktur zur Abfrage der NTP-Parameter	111
7.3.2.37	Datenstruktur mit Antwort / zum Setzen der NTP-Parameter	111

7.3.2.38	Datenstruktur zur Abfrage der Zeitinformation	112
7.3.2.39	Datenstruktur mit Antwort der Zeitinformation	112
7.3.2.40	Datenstruktur zum Leeren eines Datensammlers	113
7.3.2.41	Datenstruktur der Eigenschaften eines Datenspiegels.....	113
7.3.2.42	Liste vordefinierter Kennzahlen zur Datensammler-Identifikation	117
7.3.2.43	Datenstruktur zum Setzen / Lesen der Eigenschaften eines Datensammlers	117
7.3.2.44	MUC und EDL21-Zähler im EDL40-System	119
7.3.2.45	Datenstruktur zum Setzen / Lesen der Eigenschaften von Schwellwerten	123
7.3.2.46	Datenstruktur zu Alarm-Meldungen bei Überschreitung von Schwellwerten eines internen Datensammlers.....	123
7.3.2.47	Datenstruktur zum Schalten von Relaisausgängen (Relais-Attribute)	123
7.3.2.48	Datenstruktur zum Schalten von Relaisausgängen (Steuerungs-Attribute)	124
7.3.2.49	Datenstruktur zur Beschreibung von Parametrier-Schablonen	124
7.3.2.50	Datenstruktur zur Beschreibung von Parametrier-Schablonen	128
7.3.2.51	Datenstruktur zum Tunneln von Fremdprotokollen (3rd-Party-Protrocol-Request).....	128
7.3.2.52	Datenstruktur zum Tunneln von Fremdprotokollen (3rd-Party-Protrocol-Response).....	129
7.4	Binär-Tunnel für andere Protokolle	131
7.5	Aktionsschaltprogramm.....	132
7.5.1	Grundsätzliche Festlegungen zum Funktionsverhalten.....	132
7.6	Betriebslogbuch.....	133
7.7	Betriebslogbücher zu Datenspiegeln	138
7.8	Verwendete MUC-spezifische Fehlernummern	138

I Bildverzeichnis

Bild 1:	Systemumfeld	21
Bild 2:	Variationen möglicher Systemkonzepte	24
Bild 3:	Installations- und Aktualisierungsprozess, MUC-Controller	30
Bild 4:	Installations- und Aktualisierungsprozess, Sensor oder Aktor, Variante 1	31
Bild 5:	Installations- und Aktualisierungsprozess, Sensor oder Aktor, Variante 2	32
Bild 6:	Zugriff auf einen Sensor (Arbeitsweise der Datenspiegel; Daten vorhanden)	33
Bild 7:	Zugriff auf einen Sensor (Arbeitsweise der Datenspiegel; Daten fehlend)	34
Bild 8:	Dateiübertragung (Unicast)	35
Bild 9:	Dateiübertragung (Multicast)	36
Bild 10:	MUC-Controller, Grundform	37
Bild 11:	MUC-Controller, konstruktive Festlegung.	40
Bild 12:	Schnittstellen und Systemumfeld	41
Bild 13:	Konzept zum MWC – Messwert-Container	44
Bild 14:	Standards im MWC – Messwert-Container	45
Bild 15:	Berechtigungskonzept	48
Bild 16:	Kodierung von IPv4-Adressen	49
Bild 17:	Kodierung von IPv6-Adressen	50
Bild 18:	PIN-Belegung zum RJ10	53
Bild 19:	Kodierung für die Adressierungs-Variante E DIN 43863-5:2010-02	58
Bild 20:	Kodierung für die Adressierungs-Variante E DIN 43863-5:2010-07	59
Bild 21:	Kodierung einer Datenzeile im Data-Matrix-Code	59
Bild 22:	Zeitverhalten periodischer Push-Vorgänge	80
Bild 23:	Berechnung zum Jitter bei periodischen Pushvorgängen	81
Bild 24:	Struktur von Aktionsschaltprogrammen	83
Bild 25:	Datenspiegel / Datensammler (Parametrierung)	114
Bild 26:	Datensammler (Zugriff zur Datenbeschaffung)	115
Bild 27:	Datensammler (Beispiel-Zuordnung von EDL40-Informationen)	120
Bild 28:	Konzept von Relaisausgang und Steuerquelle	123
Bild 29:	Verhalten bei Definition mehrerer Saisonprogramme.	133
Bild 30:	MUC-Controller, Arbeitsweise zum Logbuch für Betriebsereignisse	133

II Tabellenverzeichnis

Tab. 1:	Liste ergänzend benötigter Dokumente	17
Tab. 2:	Anforderungsmatrix	29
Tab. 3:	Durch MUC berührte Schnittstellen	42
Tab. 4:	Betriebsart für Nahfunkstrecke (S-Mode/ T-Mode)	52
Tab. 5:	Parameter zur Ausführung der Datenschnittstelle für EDL-Anbindung	53
Tab. 6:	Statuswort zum MUC-Controller	56
Tab. 7:	Kodierung zur ein-eindeutigen Adressierung der Sensoren / Aktoren	58
Tab. 8:	Parameter für den direkten Zugriff	60
Tab. 9:	Parameter für die Festlegung der Sommerzeit- / Winterzeitwechsel	61
Tab. 10:	Parameter zur Einstellung der Zugriffsrechte	62
Tab. 11:	Parameter zur Einstellung der Endkundenschnittstelle	64
Tab. 12:	Status der Endkundenschnittstelle	64
Tab. 13:	Allgemeine Schnittstellen-Eigenschaften.	65
Tab. 14:	Parameter zum WAN-Status	65
Tab. 15:	Datenstruktur mit WAN-Parametern	66
Tab. 16:	Datenstruktur mit GSM-Parametern	67
Tab. 17:	Datenstruktur zum IP-T-Status	67
Tab. 18:	Datenstruktur mit IP-T-Parametern	69
Tab. 19:	Datenstruktur zu AT-Hayes Parametern	69
Tab. 20:	Datenstruktur mit GPRS-Provider-Parametern	70
Tab. 21:	Datenstruktur für GPRS-Betreiber-Parameter	70
Tab. 22:	Datenstruktur für dynamische GSM/GPRS-Betriebsparameter	72
Tab. 23:	Datenstruktur mit LAN/DSL-Parametern	73
Tab. 24:	Datenstruktur für dynamische LAN/DSL-Betriebsparameter	74
Tab. 25:	Datenstruktur für PLC-Parameter	74
Tab. 26:	Datenstruktur für dynamische PLC-Betriebsparameter	76
Tab. 27:	Datenstruktur für Parameter zum Wireless M-Bus	77
Tab. 28:	Datenstruktur zum Wireless M-Bus Status	77
Tab. 29:	Datenstruktur mit den EDL-Zähler-Schnittstellen-Parametern.	78
Tab. 30:	Datenstruktur für mögliche Push-Quellen	79
Tab. 31:	Datenstruktur für Transport-Parameter von Push-Vorgängen	80
Tab. 32:	Datenstruktur für Push-Dienste	81
Tab. 33:	Datenstruktur zur Adressierung einer Push-Quelle	82
Tab. 34:	Datenstruktur zum Transport der Installationsparameter	82
Tab. 35:	Datenstruktur zur Auflistung aller Regelsätze	83
Tab. 36:	Datenstruktur für Parameter von Regelsätzen	84
Tab. 37:	Datenstruktur für Parameter von Saisonprogrammen	85
Tab. 38:	Datenstruktur für Parameter von Tagesprogrammen	87
Tab. 39:	Datenstruktur für Parameter mit Regeln und Aktionslisten	88
Tab. 40:	Datenstruktur für Liste aller Aktionsbeschreibungen mit deren Namen (Kennzahlen)	88

Tab. 41:	Datenstruktur für Parameter zur Aktion ‚Register on Off‘	89
Tab. 42:	Datenstruktur für Parameter zur Aktion ‚Register intern‘	89
Tab. 43:	Datenstruktur für Parameter zur Aktion ‚Register extern‘	90
Tab. 44:	Datenstruktur für Parameter zur Aufzeichnung in einem Datenspiegel	90
Tab. 45:	Datenstruktur für Parameter zur Aktion ‚Zustand ausgeben‘	91
Tab. 46:	Datenstruktur für Parameter zur Aktion ‚Impuls ausgeben‘	91
Tab. 47:	Datenstruktur für Parameter zur Aktion ‚SML-Datei ausgeben‘	92
Tab. 48:	Datenstruktur für Liste aller ‚Regeln‘	96
Tab. 49:	Datenstruktur für die Liste der sichtbaren Sensoren / Aktoren	97
Tab. 50:	Datenstruktur für die Liste der aktivierten Sensoren / Aktoren	98
Tab. 51:	Datenstruktur für die Liste neu erkannter Sensoren / Aktoren	99
Tab. 52:	Datenstruktur für die Liste „verschundener“ Sensoren / Aktoren	100
Tab. 53:	Datenstruktur für die Liste deaktivierter Sensoren / Aktoren	101
Tab. 54:	Datenstruktur für die Liste aktivierter Sensoren / Aktoren	101
Tab. 55:	Datenstruktur zum Aktivieren eines Sensors / Aktors	102
Tab. 56:	Datenstruktur zum Deaktivieren eines Sensors / Aktors	102
Tab. 57:	Datenstruktur zum Entfernen eines Sensors / Aktors	102
Tab. 58:	Datenstruktur zum Auslösen von Reset	103
Tab. 59:	Datenstruktur zur Abfrage der Geräte-Identifikation	103
Tab. 60:	Datenstruktur mit Antwort zur Geräteidentifikation	105
Tab. 61:	Datenstruktur zum Übertragen einer Firmware / Datei	106
Tab. 62:	Datenstruktur zum Aktivieren einer Firmware / Datei	107
Tab. 63:	Datenstruktur für Abfrage zum Stand der Firmware / Datei-Übertragung	108
Tab. 64:	Datenstruktur zum Start der Firmware / Datei-Übertragung	109
Tab. 65:	Datenstruktur zur Statusabfrage für Multicast-Firmware-Download	109
Tab. 66:	Datenstruktur zur Übertragung beim Multicast-Firmware-Download	110
Tab. 67:	Datenstruktur zur Abfrage des Status beim Multicast-Firmware-Download	111
Tab. 68:	Datenstruktur zur Abfrage des Hersteller-spezifischen Logbuchs	111
Tab. 69:	Datenstruktur zur Abfrage der NTP-Parameter	111
Tab. 70:	Datenstruktur für Zugriff auf die NTP-Parameter	112
Tab. 71:	Datenstruktur für Abfrage der Zeitinformation	112
Tab. 72:	Datenstruktur für Zugriff auf die Zeitinformation	113
Tab. 73:	Datenstruktur zum Leeren eines Datensammlers	113
Tab. 74:	Datenstruktur für Parameter zum Datenspiegel	116
Tab. 75:	Liste vordefinierter Kennzahlen zur Datensammler-Identifikation	117
Tab. 76:	Datenstruktur mit Parametern zum Datensammler	118
Tab. 77:	Datenstruktur mit Parametern zu Relais-Ausgängen	124
Tab. 78:	Datenstruktur mit Parametern zu Relais-Steuerquellen	124
Tab. 79:	Datenstruktur mit Liste aller Parametrier-Schablonen	128
Tab. 80:	Kommando zur manuellen Anwendung von Parametrier-Schablonen	128
Tab. 81:	Kommando zum Tunneln von Fremdprotokollen in Senderichtung	129
Tab. 82:	Kommando zum Tunneln von Fremdprotokollen in Empfangsrichtung (Polling-Request)	130

Tab. 83:	Kommando zum Tunneln von Fremdprotokollen in Empfangsrichtung (Polling-Response)	131
Tab. 84:	Liste vordefinierter Identifier zu Fremdprotokollen	132
Tab. 85:	Betriebslogbuch, Liste der je Ereignis immer vorhandenen Zusatzinformationen.	134
Tab. 86:	Betriebslogbuch, Liste der Ereignis spezifischen Zusatzinformationen.	135
Tab. 87:	Betriebslogbuch, Kodierung der Einträge	138
Tab. 88:	MUC-spezifische Fehlernummern	139

III Abkürzungsverzeichnis

Einheiten:

Hinsichtlich physikalischer Messgrößen und Einheiten gelten die im SI (siehe DIN 1301, Teil 1 [1]) getroffenen Vereinbarungen.

Relevante Abkürzungen:

Den nachfolgenden Abkürzungen können arabische Ziffern nachgestellt werden, um mehrfach auftretende Ausprägungen derselben Funktion / desselben Signals unterscheiden zu können.

AL	⇔ Application Layer
CCITT	⇔ Comité Consultatif International Télégraphique et Téléphonique
DIN	⇔ Deutsches Institut für Normung e.V.
DKE	⇔ Deutsche Elektrotechnische Kommission im DIN
DLMS	⇔ Device Language Message Specification
(E) DIN	⇔ Entwurf einer Norm des DIN
EN	⇔ Europäische Norm,
FNN	⇔ Forum Netztechnik/ Netzbetrieb im VDE
ID	⇔ Identifikationsnummer,
IEC	⇔ International Electrotechnical Commission
IEEE	⇔ Institute of Electrical and Electronics Engineers
IP	⇔ Internet Protocol
IP-T	⇔ IP-Telemetrie
ISO	⇔ Internationale Organisation für Normung
LAN	⇔ Local Area Network
LSB	⇔ Least Significant Bit, niederwertigstes Bit
MUC	⇔ Multi Utility Communication, ⇔ Metering Unified Container
MSB	⇔ Most Significant Bit, höchstwertigstes Bit
NTP	⇔ Network Time Protocol
OBIS	⇔ Objekt-Identifikations-System
OBIS-T	⇔ OBIS Telemetrie
OMS	⇔ Open-Metering in FIGAWA/ZVEI
OSI	⇔ Open Systems Interconnection Reference Model
RS232	⇔ Serielle Schnittstelle
SML	⇔ Smart Message Language
SyM ²	⇔ Synchronous Modular Meter
TAG	⇔ Merkmal / Kennzeichnung / Auszeichnung bei der Kodierung von Datenelementen
TCP	⇔ Transmission Control Protocol
WAN	⇔ Wide Area Network
XML	⇔ Extensible Markup Language
ZVEI	⇔ Zentralverband Elektrotechnik- und Elektronikindustrie

IV Normen und referenzierte Dokumente

[1]	DIN 1301, Teil 1	10.2002	Einheiten, Teil 1: Einheitennamen, Einheitenzeichen
[2]	EN 13757-1	--.--	Communication system for meters and remote reading of meters - Part 1: Data Exchange
[3]	EN 13757-2	--.--	Communication system for meters and remote reading of meters - Part 2: Physical and Link Layer
[4]	EN 13757-3	--.--	Communication system for meters and remote reading of meters - Part 3: Dedicated Application Layer
[5]	EN 13757-4	--.--	Communication system for meters and remote reading of meters - Part 4: Wireless Meter Readout
[6]	E DIN 43863-4	06.2009	Zählerdatenkommunikation – IP-Telemetrie (bis zur Verabschiedung der Norm ist die jeweils aktuelle Entwurfssfassung heranzuziehen)
[7]	DIN EN 62056-61	01.2003	Messung der elektrischen Energie – Zählerstandsübertragung, Teil 61: OBIS Objekt Identification System
[8]	IEC 62056-62	2002	Messung der elektrischen Energie – Zählerstandsübertragung, Tarif- und Laststeuerung - Teil 62: Interface-Klassen (IEC 62056- 62:2002)
[9]	DIN EN 62056-62	01.2003	
[10]	CCITT-CRC16	--.--	Standard der CCITT zur Prüfsummenberechnung
[11]	ISO 8859-15	03.1999	Informationstechnik - 8-Bit-Einzelbyte-codierte Schriftzeichensätze - Teil 15: Lateinisches Alphabet Nr. 9
[12]	prEN 62056-5-8	--.--	SML container services (SML, Smart Message Language) (bis zur Verabschiedung der Norm ist die jeweils aktuelle Entwurfssfassung heranzuziehen)
[13]	SyM ²	--.--	Synchronous Modular Meter – Spezifikation im tLZ-Projekt, siehe „www.sym2.org“
[14]	EDL	05.2011	FNN: Lastenheft EDL (Elektronische Haushaltszähler - Funktionale Merkmale und Protokolle); Version 1.1
[15]	Open-Metering in FIGAWA/ZVEI	--.--	Festlegungen zum Einsatz von M-Bus / Wireless-M-Bus bei Smart Metering Anwendungen, siehe „www.openmetering.org“.
[16]	RFC 1305	--.--	NTP (V3)
[17]	RFC 3927	03.2005	Dynamic Configuration of IPv4 Link-Local-Address („Zero- Config“)

Tab. 1: Liste ergänzend benötigter Dokumente

V Begriffe

Betriebsbereitschaft:

Die an verschiedenen Stellen im Lastenheft genutzte Formulierung "*Mit Herstellen der Betriebsbereitschaft*" definiert folgenden Zustandswechsel:

- Die Energieversorgung des Geräts wurde hergestellt.
- Der Startprozess zu einem möglicherweise in dem Gerät vorhandenen Betriebssystem ist abgeschlossen (⇔ alle zum Betrieb der Applikation benötigten Funktionen sind betriebsbereit).
- Der Startprozess der Applikation (⇔ hier konkret der MUC-Firmware) ist abgeschlossen.
- Die Applikation beginnt, soweit erforderlich, externe Kommunikationsverbindungen (WAN, Kundenschnittstelle, Anbindung von Sensoren / Aktoren, ...) aufzubauen. Das "*Herstellen der Betriebsbereitschaft*" erfordert nicht, dass dieser Aufbau externer Kommunikationsverbindungen auch abgeschlossen ist.

Heizkostenverteiler:

Zur Präzisierung wird festgelegt, zwischen den Begriffen „Heizkostenverteiler“ und „Wärmezähler“ (s.u.) zu unterscheiden.

Der Begriff „Heizkostenverteiler“ wird für Mess-Einrichtungen verwendet, die an dezentraler Stelle, üblicherweise den Heizkörpern, angeordnet sind, und die je Mess-Stelle dimensionslose Verbrauchseinheiten erfassen.

Wärmezähler:

Zur Präzisierung wird festgelegt, zwischen den Begriffen „Wärmezähler“ und „Heizkostenverteiler“ (s.o.) zu unterscheiden.

Der Begriff „Wärmezähler“ wird für Mess-Einrichtungen verwendet, die an zentraler Stelle vergleichbar einem Stromzähler angeordnet sind, und die Summe der von einem Abnehmer vom Versorger bezogenen Wärmemenge erfassen.

1 Bezug

Der sich abzeichnende Wechsel in das Zeitalter elektronischer Messtechnik und Technologien für die Erfassung von Versorgungsgrößen bei Privatkunden (Wasser, Wärmemengen, Strom und Gas) bietet u. a. die Chance, ein standardisiertes System zur Erfassung, Übertragung und Weiterverarbeitung der Messwerte einzuführen. Die Realisierung solcher Konzepte wird durch die in den letzten Jahren bereits etablierten Standards öffentlicher Kommunikationsnetze, allen voran das Internet, unter wirtschaftlichen Gesichtspunkten ermöglicht.

Zeitgleich treiben politische Zielsetzungen wie Energieeffizienz und aktive Informationsgestaltung gegenüber Privatkunden diese Umsetzung.

Um künftig die Anforderungen der verschiedenen Rollen in den liberalisierten Branchen (Erzeugung, Verteilung, Handel, Mess-Stellenbetreiber, Messdienstleister, Privatkunden, ...) wirtschaftlich tragbar abdecken zu können, wird eine weitgehende Standardisierung der Schnittstellen (Hardware, Software, Prozesse) benötigt.

Der Antrieb, diesen Standard zu schaffen, wird von der Erkenntnis getrieben, der Einführung vieler neuer Schnittstellen und damit verbundener Mehrkosten nur durch optimierte Prozesse (Datenbeschaffung, Bereitstellung, flexible Tarifierung, ...) begegnen zu können.

Optimierte Prozesse benötigen perfekt aufeinander abgestimmte standardisierte Schnittstellen, so dass einzelne Marktteilnehmer je nach eigener Zielsetzung singulär Komponenten austauschen können, ohne das Gesamtsystem und damit die Teilprozesse anderer Marktteilnehmer zu beeinflussen.

Mit dem Lastenheft Multi Utility Communication (MUC) wird das Anliegen verfolgt, diesen Standard im Sinne der Integration bestehender Normen und Konzepte sowie Festlegung ergänzend benötigter Vorgaben zu schaffen. Es ist erklärtes Ziel, dabei wo immer technisch sinnvoll möglich, auf Dokumente zu verweisen, deren Umsetzung frei von Lizenzen ist.

Das MUC-Lastenheft definiert ein Konzept aus Sicht der Applikation, Systemkomponenten und deren Schnittstellen. Wann immer möglich, wird das Verhalten der Systemkomponenten sowie die Ausprägung von Schnittstellen durch Verweise auf heute bereits akzeptierte und erprobte Normen oder Spezifikationen bestimmt. Nur dann, wenn die vorhandenen Festlegungen nicht ausreichen oder im Sinne einer Reduktion denkbarer Varianten eingeschränkt werden müssen, trifft das MUC-Lastenheft eigene Vorgaben.

2 Eckpunkte

Zentrale Eckpunkte des Vorhabens sind:

- Beschreibung des Standards zur automatisierten Messdatenerfassung bei Privatkunden über offene Netze.
- Abdeckung aller Sparten (Gas, Wasser, Wärme, Strom).
- Einhaltung eichrechtlicher Vorgaben im Umgang mit abrechnungsrelevanten Messwerten.
- Bereitstellung von erfassten Messdaten für Aufgaben der Betriebsführung (Smart Grid / Smart Generation / ...).
- Bereitstellung der Informations-Infrastruktur für künftige Dienste (Smart Customer, ...).
- Direkter Zugriff durch Privatkunden auf die Messwerte über In-House-Communication mittels standardisierter Technik (PC/Laptop, TCP/IP, Web-Server, ...) oder durch Kundeninformationseinheiten („MUC-Display“) über etablierte Standards (KNX, ...).
- Technische Umsetzung relevanter Hinweise aus Studien zur effizienten Energienutzung (z. B. E-Energy, ...).
- Technische Entkopplung der Systembereiche Messtechnik und Weitverkehrskommunikation durch eine zentrale Instanz in der Liegenschaft (MUC-Controller) zum langfristigen Investitionsschutz (langlebige Messtechnik versus innovative Kommunikationstechnik).

3 Systemumfeld

Das Systemumfeld kann aus den geltenden gesetzlichen Rahmenbedingungen und im Markt stattfindenden Entwicklungen wie folgt skizziert werden:

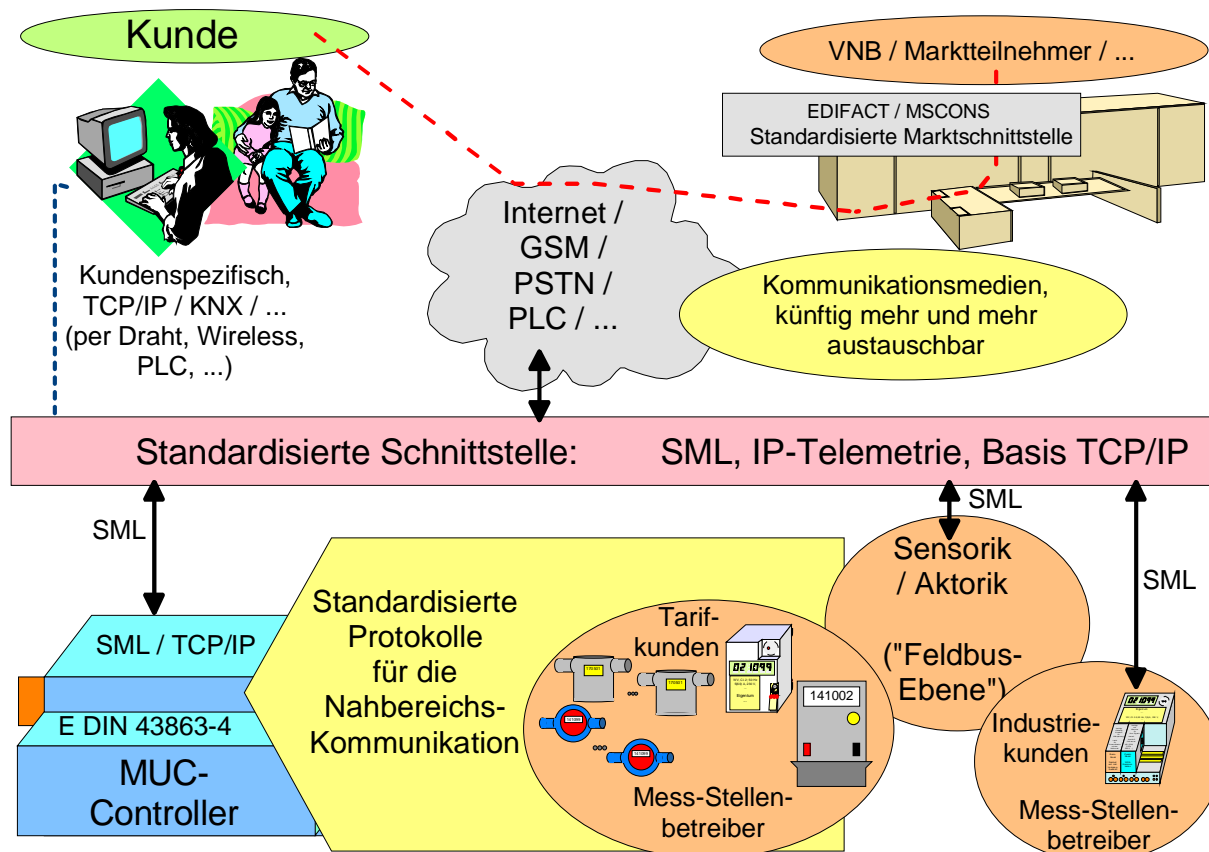


Bild 1: Systemumfeld

Unabhängig der mit Bild 1 skizzierten Mischung aus Markttrollen, Geräten und Protokollen wird ein funktionsfähiger Standard Lösungen für die verschiedensten, im Feld anzutreffenden, Installationsvarianten anbieten müssen. Auch wenn diese Lösungen – schon aus ökonomischer Überlegung – niemals 100 % der Varianten abdecken können, muss der Standard mindestens 80 – 90 % erreichen.

3.1 Allgemeine Anforderungen, Beschreibung des Soll-Zustandes

3.1.1 Zählerebene

Die Kommunikationsprotokolle sollen auf bestehenden, möglichst europäischen oder internationalen Standards aufbauen.

Die gewählten Kommunikationsprotokolle dürfen nicht spartenspezifisch ausgeführt werden.

Die Interoperabilität zwischen Zählern verschiedener Hersteller muss gewährleistet sein.

Es gibt Zähler, die per Drahtschnittstelle kommunizieren und es gibt Zähler, die über Nahfunk (z. B. 868 MHz) kommunizieren.

Die Zählerkommunikation folgt dem ISO/OSI-7-Schichtenmodell.

Es sind einfache, genormte Steckverbindungen (z. B. RJ 45) bei der drahtgebundenen Datenkommunikation vorzusehen, gegebenenfalls in entsprechender IP-Ausführung.

Es werden keine nachgebildeten Zählerstände (z. B. über Impulsausgänge) übertragen, sondern nur originäre Zählerstände (real data) verwendet.

Abhängig vom Einsatz sind zwei Arten von Zählern vorzusehen:

- Der einfache Zähler sendet automatisch in kurzen Abständen oder auf Anfrage seitens des MUC-Controller:
 - Zähleridentifikation,
 - Zählerstand (real data),
 - Einheit,
 - OBIS-Kennzahl,
 - Status.
- Der lastgangfähige Zähler sendet automatisch in kurzen Zeitabständen oder auf Anfrage seitens des MUC-Controller:
 - Zähleridentifikation,
 - Zählerstand (real data),
 - Einheit,
 - OBIS-Kennzahl,
 - Status.
- Darüber hinaus automatisch in größeren Zeitabständen oder auf Anfrage seitens des MUC-Controller:
 - Zähleridentifikation,
 - „Daily profile“ des Vortages als Zählerstandsgang (real data) mit Zeitstempeln,
 - Der Zählerstandsgang basiert auf Stundenwerten,
 - Einheit,
 - OBIS-Kennzahl,
 - Status.

Um die Zähler in Betrieb zu nehmen ist vor Ort keine Parametrierung notwendig.

Dies gilt sowohl bei erstmaliger Inbetriebnahme einer Verbrauchsstelle als auch bei einem planmäßigen oder außerplanmäßigen Zählerwechsel.

Die Zähler haben einen niedrigen Energiebedarf, so dass sie ihre jeweilige Eichgültigkeit und mindestens eine Verlängerungstichprobe unter den in diesem Dokument genannten Voraussetzungen ohne Batteriewechsel durchstehen, sollten sie für den Betrieb auf eine Batterie angewiesen sein.

Lastgangfähige Zähler dürfen auf eine Spannungsversorgung angewiesen sein.

3.1.2 Ebene MUC-Controller

Der MUC-Controller stellt eine von einem Zähler logisch und funktionell unabhängige Einheit dar.

Der MUC-Controller ist der „Master“ für alle Zähler im Anschlussobjekt, die mit dem MUC-Controller kommunizieren.

Der MUC-Controller kommuniziert über Drahtschnittstelle und / oder über Nahfunk mit den Zählern, abhängig von der Ausstattung der Zähler.

Für lastgangfähige Zähler stellt der MUC-Controller den Timeserver zur Zeitsynchronisation dar, für einfache Zähler ergänzt der MUC-Controller die vom Zähler kommenden Daten mit dem jeweils aktuellen Zeitstempel.

Der MUC-Controller darf keine Eigenschaften aufweisen, die eine Eichpflicht begründen würden.

Der MUC-Controller steuert, mit entsprechenden Sicherheitsroutinen, die von einer Zentrale kommenden Abschaltbefehle und die Befehle, die die Aufhebung der Abschaltung anordnen.

Der MUC-Controller stellt die IP-basierte, bidirektionale Kommunikation mit einer oder mehrerer Zentralen sicher.

Länger andauernde Unterbrechungen der Datenkommunikation zwischen dem MUC-Controller und den Zählern sowie Fehlermeldungen der Zähler werden kurzfristig an die Zentrale gemeldet.

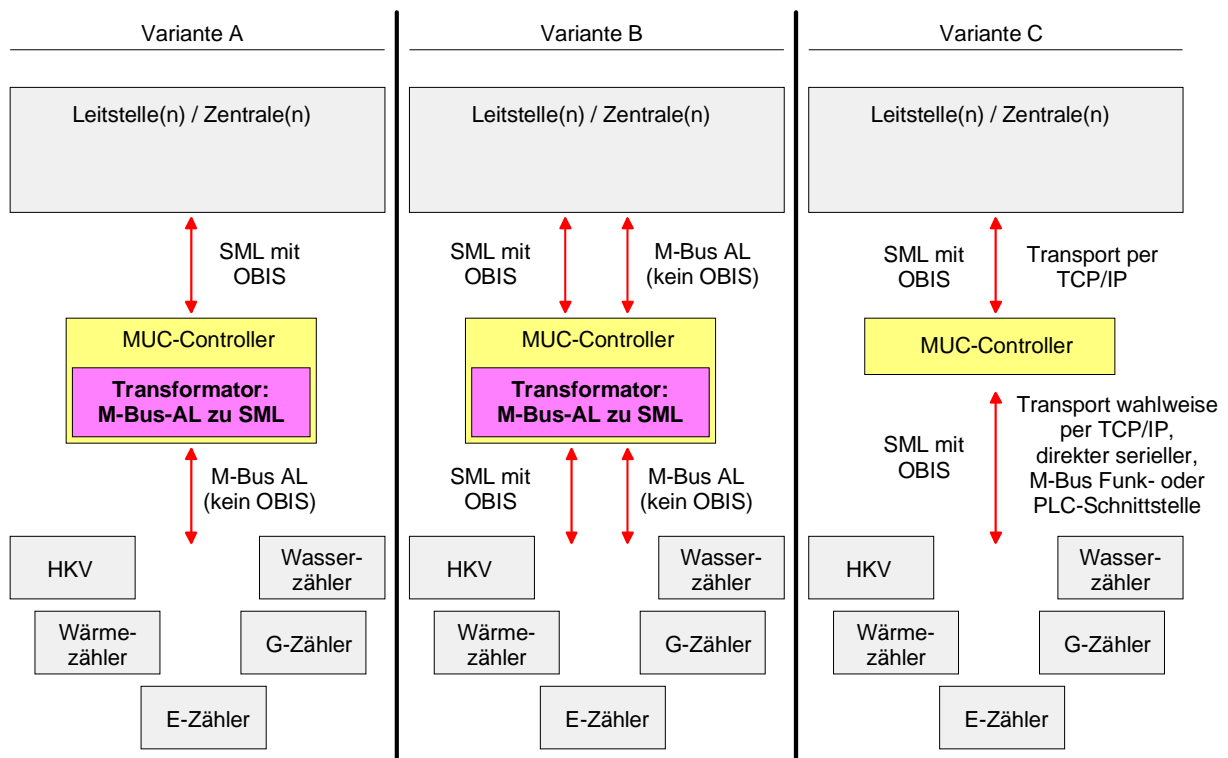
3.1.3 Grundsätzliches

Grundsätzlich ist das Gesamtsystem nur dann massentauglich, wenn ein sehr hoher Automationsgrad bei der Implementierung erreicht werden kann. Dies bedingt, dass eine Selbstanmeldung eines neuen Zählers nicht auf der Ebene MUC-Controller endet, sondern sich über die technische Messdatenverwaltung bis in das zugehörige Abrechnungssystem zieht.

Das Gesamtsystem ist nach anerkannten Regeln der Technik gegen die missbräuchliche Benutzung Unbefugter zu sichern.

3.2 Migrationsphase

Mit Blick auf die in 2008 im Feld anzutreffenden Systeme und deren Vergleich zu den Zielsetzungen des MUC-Lastenhefts werden vom Konzept her unterschiedliche Systemlösungen erkennbar:



Alle Varianten basieren auf M-Bus Physical- und Link-Layer für die Nahfunkstrecke.
Für E-Zähler werden in der Variante B zwei Anbindungen (eHZ optisch und M-Bus Funk) diskutiert.
Variante C sollte vier Alternativen bieten: TCP/IP, eHZ optisch, M-Bus Funk oder PLC.
Solange PLC nicht standardisiert ist, soll TCP/IP als Harnisch um PLC-Sub-Systeme genutzt werden.

Bild 2: Variationen möglicher Systemkonzepte

Variante A wurde der Vollständigkeit wegen aufgenommen und wird nicht weiter verfolgt.

3.2.1 Festlegungen zur Variante B (Migrationsphase)

Um den in einem MUC-Controller benötigten Transformator (M-Bus-AL zu SML) möglichst selten im Feld per remote Firmware-Download aktualisieren zu müssen, ist die zu dessen Funktion benötigte Übersetzungsliste der OBIS-Kennzahlen zum M-Bus-Identifikationssystem adaptiv zu gestalten. Konkret wird dazu gefordert:

- Die Übersetzungsliste ist bei Lieferung auf die üblicherweise benötigten Kennzahlen zu initialisieren. Zu diesem Zweck enthält die Übersetzungsliste Einträge aus M-Bus-Identifikatoren und OBIS-Kennzahlen.
- Künftig neue Produkte müssen die Einträge dieser Liste dynamisch im Sinne einer adaptiven Vorgehensweise erweitern können, so dass Kennzahlen für neue Inhalte durch die Produkte selber an die Liste angefügt werden und eine Parametrierung im Feld nicht notwendig ist.
- MUC-Controller müssen den lesenden wie schreibenden (auch löschenden) Zugriff¹ auf die Übersetzungsliste an der WAN-Schnittstelle bereitstellen.

¹ Diese Funktion ist nicht für den Regelbetrieb sondern ausschließlich zu Wartungszwecken vorgesehen.

- Kann die Übersetzungsliste nicht erweitert werden (beispielsweise will die List bereits voll ist), ist eine Fehlermeldung zu erzeugen und im Betriebslogbuch abzuspeichern.
- Die Übersetzungsliste muss Platz für mindestens 1000 M-Bus-Identifikatoren² enthalten.
- Die in der Übersetzungsliste vorhandenen M-Bus- Identifikatoren müssen eindeutig sein. Ein MUC-Controller muss eine Fehlermeldung erzeugen, falls ein Eintrag hinzugefügt werden soll, dessen M-Bus- Identifikator bereits in der Liste vorhanden ist.

² Wegen der im M-Bus vorliegenden Struktur zur Identifikation von Datenpunkten können verschiedene M-Bus-Identifikatoren in der Übersetzungsliste auf dieselbe OBIS-Kennzahl verweisen.

3.3 Anforderungsmatrix

Für die Ausarbeitung der mit den folgenden Kapiteln definierten Details wurde eine Anforderungsmatrix zu Grunde gelegt, deren Inhalt Ziele und Prioritäten wie folgt in Verbindung bringt:

Pos.	Merkmal / Ziel	Priorität			Kommentar / Hinweis
		A ³	B ⁴	C ⁵	
1	Allgemeine Anforderungen				
1.1	Ein System für alle Medien, Kompatibilität für alle Sparten	X			Ein System bedeutet: Vom Zähler bis in die Leitstelle dieselben Konzepte.
1.2	Einheitliche Datenprotokolle	X			Auf den einzelnen Teilstrecken (primär, sekundär und tertiär Kommunikation) werden je Teilstrecke dieselben Protokolle für alle Sparten verwendet.
1.3	Erfassung originärer Zählerstände	X			Der originale Zählerstand kommt aus dem Mess-Sensor.
1.4	Herstellerunabhängigkeit	X			
1.5	Optional drahtlose Verbindung im Haus	X			
1.6	Datensicherheit, Zugriffsschutz	X			Eine Verschlüsselung der Übertragung ist nicht notwendig.
1.7	Durchgehendes Informationsmanagement in ein System (Leitstelle)	X			
1.8	Langlebigkeit der Systeme	X			Die Langlebigkeit wird nicht im Sinne der Eichgültigkeit verstanden, sondern im Sinne der Standzeit des Systems (Protokolle, Datenstrukturen, ...).
1.9	Nutzung Massentechnik, keine Sonderlösung	X			Vorhandene Technologien (z.B. TCP/IP) sind anzuwenden.
1.10	Automatische Selbstanmeldung	X			
1.11	Keine Nutzung von Infrastruktureinrichtungen der Endkunden	X			Für die Weitverkehrskommunikation
1.12	MUC-Controller ist nicht eichpflichtig	X			

³ Priorität A ⇔ Muss erfüllt sein.

⁴ Priorität B ⇔ Sollte erfüllt sein.

⁵ Priorität C ⇔ Falls einfach / zu geringen Kosten möglich zu realisieren („nice to have“).

Pos.	Merkmal / Ziel	Priorität			Kommentar / Hinweis
		A ³	B ⁴	C ⁵	
1.13	Ein-eindeutige und einheitliche Identifikation der Mess-Sensoren	X			
2	Kommunikation				
2.1	Kommunikationsfähigkeit für alle Zähler	X			
2.2	Applikation unabhängig von Übertragungssystemen (Draht, Funk)	X			
2.3	Adressierung nach TCP/IP	X			Hierunter ist die Adressierung bis zum MUC-Controller, nicht bis zum Sensor zu verstehen.
2.4	Datensammler als eigenständiges Gerät ⇔ arbeitet als MUC-Controller / Verbindung zur Zentrale	X			
2.5	E-Zähler integriert den Datensammler ⇔ arbeitet als MUC-Controller / Verbindung zur Zentrale			X	Wird im Grundsatz abgelehnt, da anderenfalls die mit dem MUC-Lastenheft angestrebte Flexibilität nicht erreicht wird. Falls die Einheiten integriert werden, muss die Funktionalität MUC-Controller von der des Sensors getrennt werden
2.6	Tägliche Weiterleitung des Zählerstands an die Zentrale Individueller Zugriff durch Berechtigte	X			Regelbetrieb für Abrechnung und Bilanzierung Erfolgt auf Anforderung mit einer Latenz von: - Strom: 5 min - Gas: 5 min - Wasser: 5 min - Wärme: 5 min - HKV: 5 min
2.7	Bidirektionale Nahkommunikation Messeinrichtung Strom	X			Die Sensoren können in erster Stufe auch unidirektional arbeiten.
2.8	Bidirektionale Nahkommunikation Messeinrichtung Gas	X			Die Sensoren können in erster Stufe auch unidirektional arbeiten.
2.9	Bidirektionale Nahkommunikation Messeinrichtung Wasser	X			Die Sensoren können in erster Stufe auch unidirektional arbeiten.
2.10	Bidirektionale Nahkommunikation Messeinrichtung Wärme	X			Die Sensoren können in erster Stufe auch unidirektional arbeiten.
2.11	Bidirektionale Nahkommunikation Messeinrichtung Heizkostenverteiler	X			Die Sensoren können in erster Stufe auch unidirektional arbeiten.

Pos.	Merkmal / Ziel	Priorität			Kommentar / Hinweis
		A ³	B ⁴	C ⁵	
2.12	Bidirektionale Nahkommunikation Messeinrichtung Unterzähler (Wasser)	X			Die Sensoren können in erster Stufe auch unidirektional arbeiten.
2.13	Bidirektionale Fernkommunikation	X			Einsatzfall: Beispielsweise Ansteuerung von Aktoren zur Abschaltung / Funktionalität bestehender Rundsteuerungsanlagen.
2.14	Direkter Kundenzugriff auf seine Verbrauchsdaten mit einfachen Mitteln	X			
2.15	Anbindung an Hausautomatisierung/Gebäudeinformation	X			
2.16	Anbindung Fernwirktechnik		X		Verwendung der Messwerte zu Aufgaben im Sinne von Smart-Grid. Die konkreten Anforderungen sind noch mit den Projektgruppen zu Smart-Grid abzustimmen.
2.17	Fernschaltung im Netz	X			
2.18	Anbindung von Unterzählern	X			Siehe oben: Heizkostenverteiler
2.19	Optimiertes Datenvolumen	X			
3	Betriebliche Anforderungen				
3.1	Einfache Montage vor Ort, keine Parametrierung an der Mess-Stelle	X			
3.2	Problemloser Zählerwechsel	X			
3.3	Niedriger Energiebedarf der Smart Meters (Gas, Wasser, Wärme, Heizkostenverteiler)	X			
3.4	Standzeit der Zähler im Netz, wie heute	X			Falls Batterien eingesetzt werden, sollen diese zu wechseln sein, ohne den Zähler aus dem Feld nehmen zu müssen. Bei Wasserzählern können auch nicht wechselbare Batterien verwendet werden.
4	Funktionen vor Ort				
4.1	Fernabschaltung / Leistungsbegrenzung Strom	X			
4.2	Fernabschaltung / Leistungsbegrenzung Gas		X		

Pos.	Merkmal / Ziel	Priorität			Kommentar / Hinweis
		A ³	B ⁴	C ⁵	
4.3	Fernabschaltung / Leistungsbegrenzung Wasser			X	
4.4	Fernabschaltung / Leistungsbegrenzung Wärme			X	
4.5	Tariffbildung vor Ort im Zähler		X		Die Ansteuerung der jeweils aktiven Register erfolgt wahlweise über eine interne Tarifschaltuhr oder per Kommando über die Datenschnittstelle.
4.6	Tarifregister vor Ort in der Messeinrichtung		X		Die Registrierung mehrerer Tarife muss bei Bedarf im Zähler vor Ort optional möglich sein.
4.7	Lastgangbildung vor Ort in der Messeinrichtung bei Strom	X			In eichtechnisch nutzbarer Ausprägung
4.8	Lastgangbildung vor Ort in der Messeinrichtung bei Gas			X	In eichtechnisch nutzbarer Ausprägung
4.9	Lastgangbildung vor Ort in der Messeinrichtung bei Wasser			X	In eichtechnisch nutzbarer Ausprägung
4.10	Lastgangbildung vor Ort in der Messeinrichtung bei Wärme		X		In eichtechnisch nutzbarer Ausprägung
4.11	Lastgangbildung vor Ort in der Messeinrichtung bei Heizkostenverteilern				Ohne Bewertung.
4.12	Netzkenngößen	X			Betrifft alle Sparten mit unterschiedlichen Anforderungen.
4.13	Fehlererkennung / Manipulation	X			Betrifft alle Sparten mit unterschiedlichen Anforderungen.
5	Weiteres				
5.1	Handling von großen Datenmengen	X			
5.2	Kundenbindung / Neue Produkte und Marktanteile	X			Das System muss in der Lage sein, derartige Informationen von den Händlern an den Endkunden zu leiten (als Dienstleistung des Netzbetreibers an den Händler sowie zur Information an den Endkunden).

Tab. 2: Anforderungsmatrix

3.4 Prozesse

Das MUC-Lastenheft legt nachstehend skizzierte Prozesse für die Definition der Datenstrukturen / Funktionen eines MUC-Controllers zu Grunde. Dabei wird davon ausgegangen, dass die Aufgaben des Zentral-Systems sowohl automatisiert als auch manuell durch Mitarbeiter vor Ort wahrgenommen werden können:

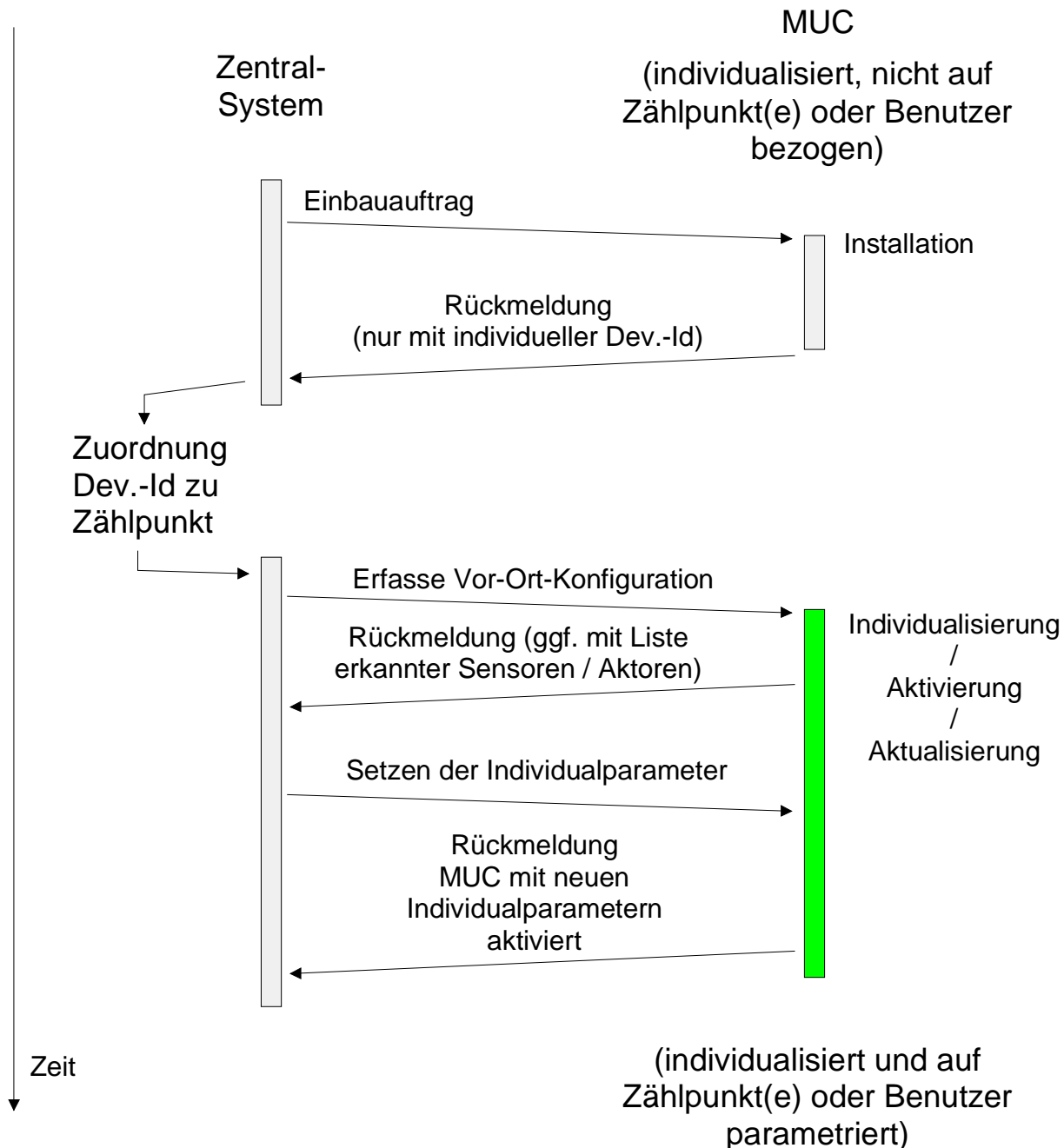


Bild 3: Installations- und Aktualisierungsprozess, MUC-Controller

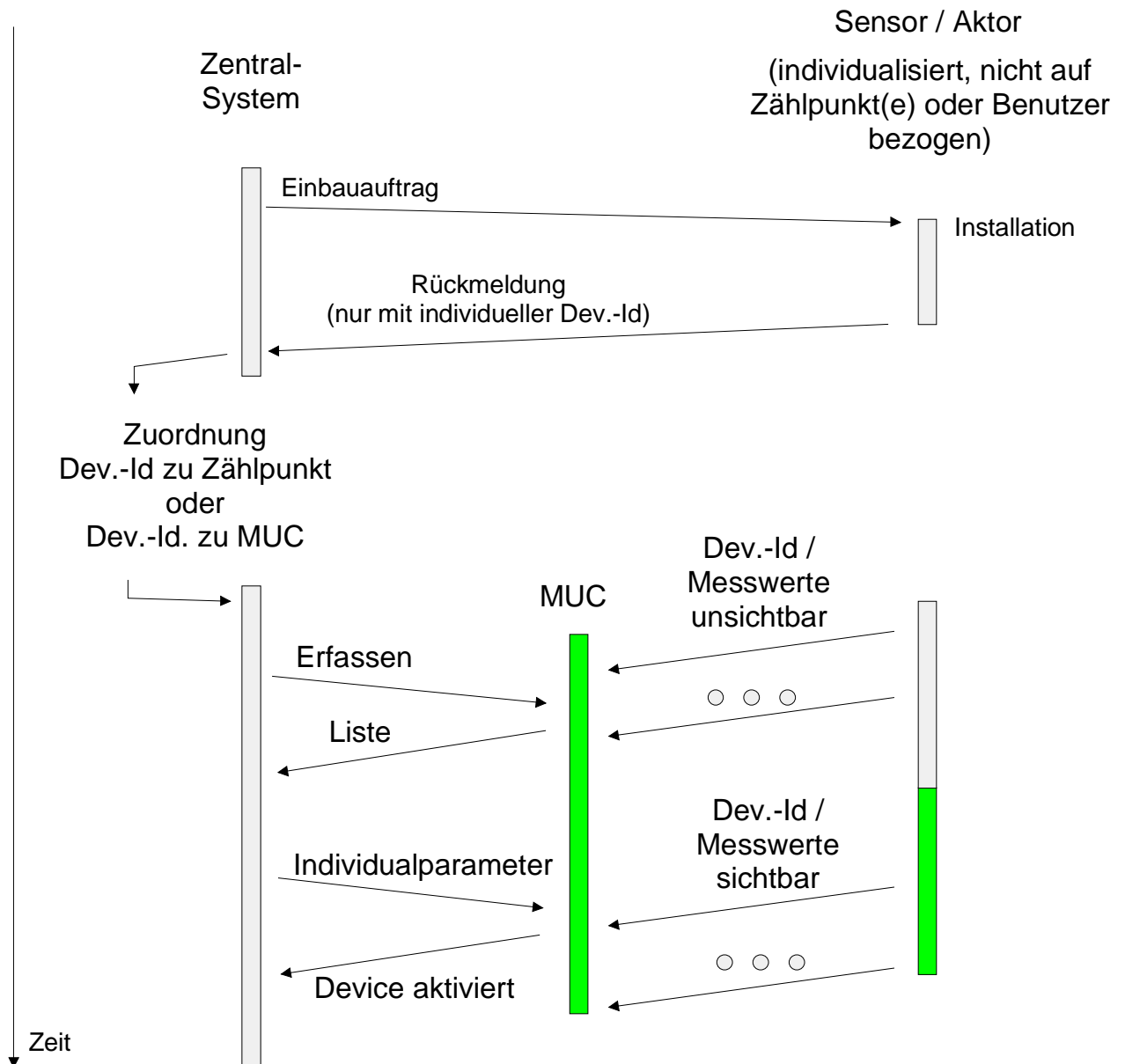


Bild 4: Installations- und Aktualisierungsprozess, Sensor oder Aktor, Variante 1

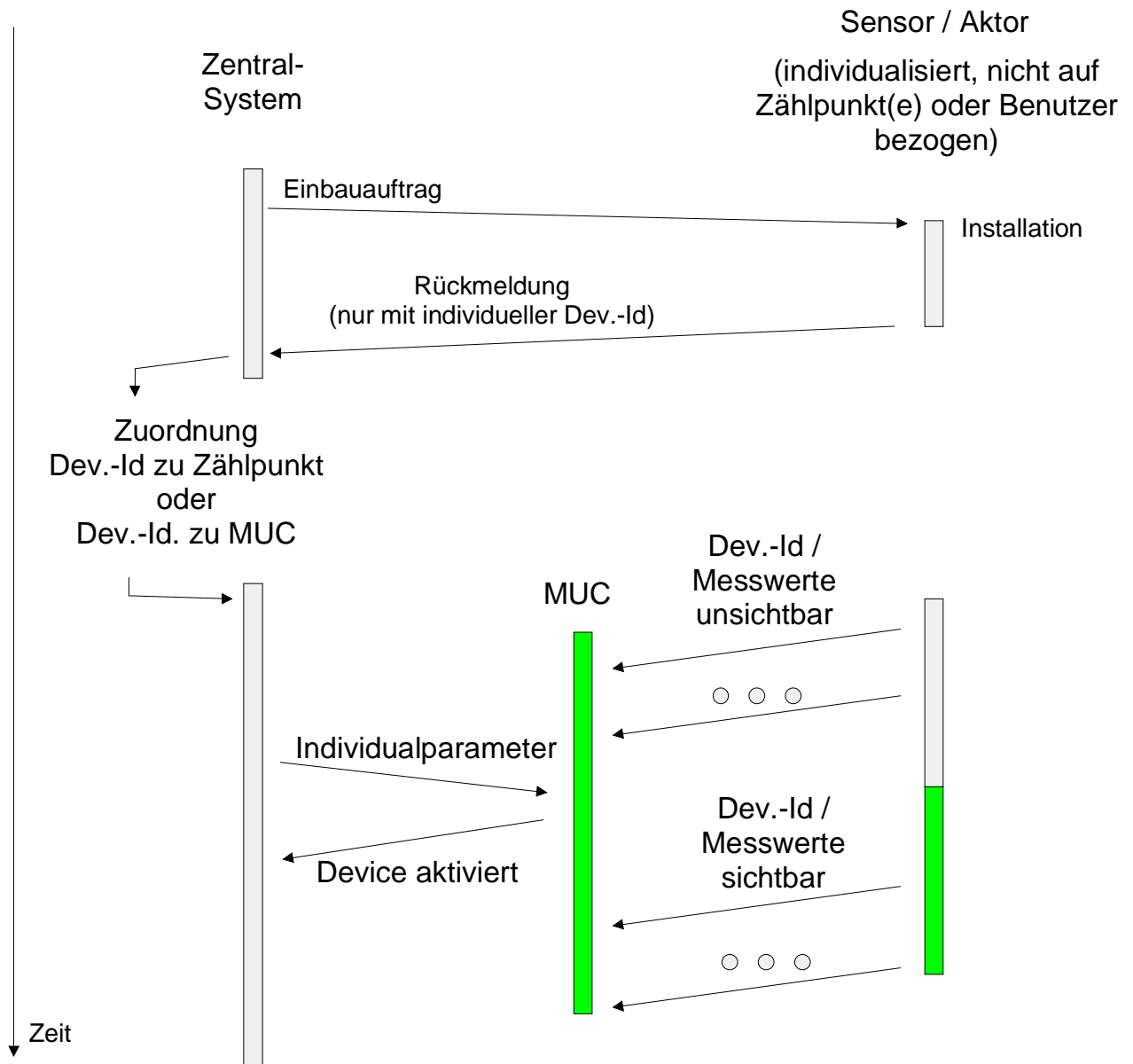


Bild 5: Installations- und Aktualisierungsprozess, Sensor oder Aktor, Variante 2

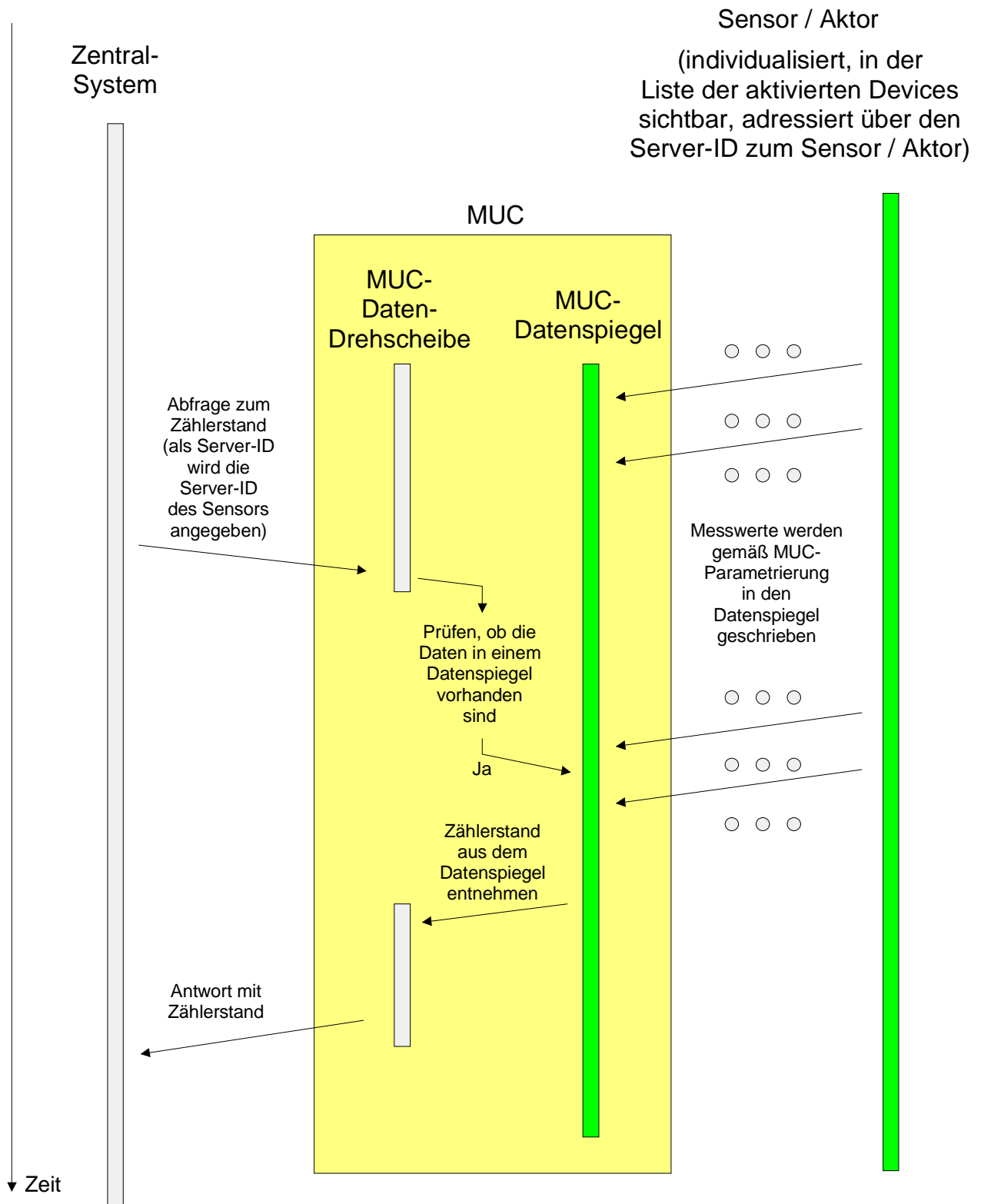


Bild 6: Zugriff auf einen Sensor (Arbeitsweise der Datenspiegel; Daten vorhanden)

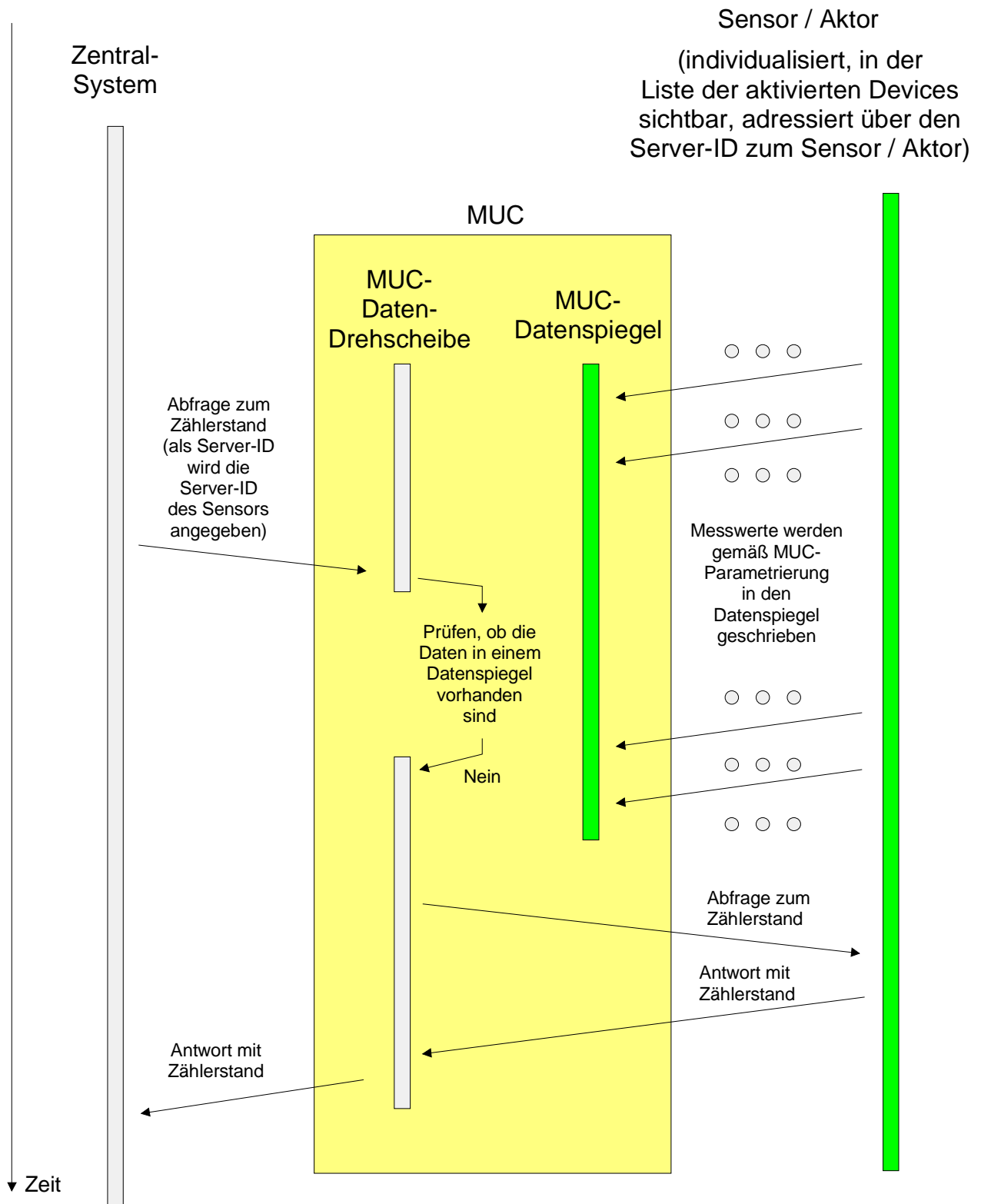


Bild 7: Zugriff auf einen Sensor (Arbeitsweise der Datenspiegel; Daten fehlend)

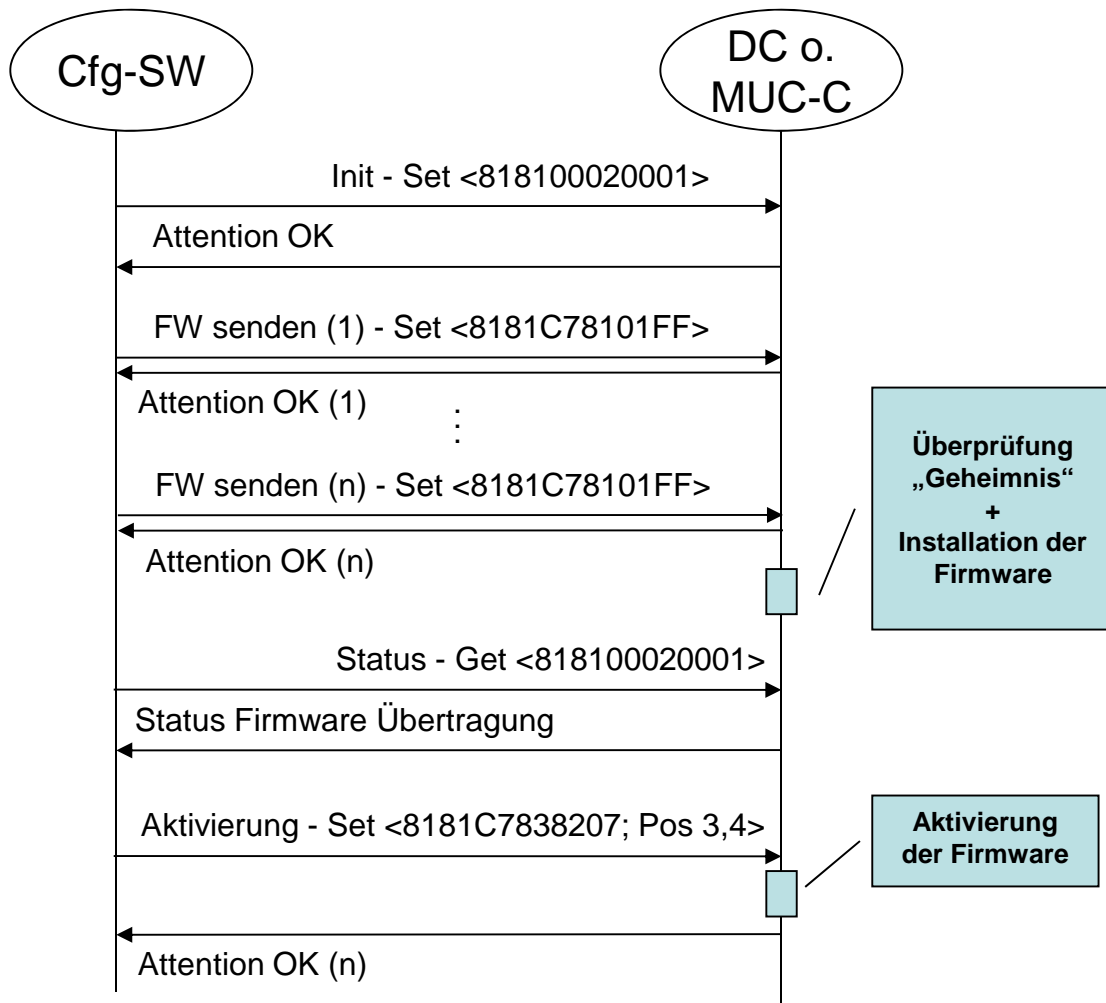


Bild 8: Dateiübertragung (Unicast)

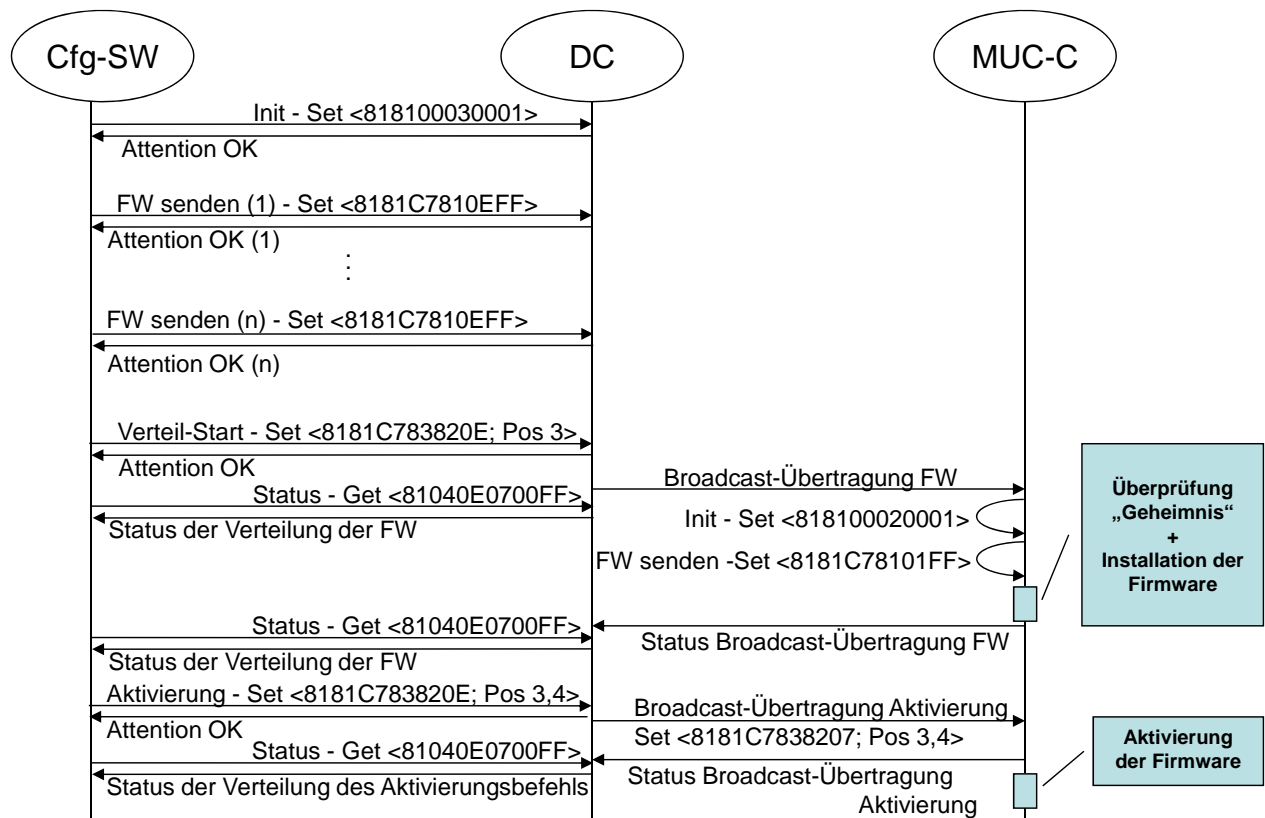


Bild 9: Dateiübertragung (Multicast)

4 Systemkomponenten

4.1 MUC-Controller

Der MUC-Controller verbindet als zentrale Einheit die Endgeräte (Zähler, Sensoren, Aktoren) mit der Weitverkehrsschnittstelle sowie der Schnittstelle zu Visualisierungseinheiten beim Endkunden.

Das MUC-Lastenheft beschreibt in diesem Sinne das funktionelle Verhalten zu einem MUC-Controller; es lässt dabei bewusst offen, ob ein MUC-Controller als eigenständiges Gerät oder als verteilte Funktionseinheit über mehrere Geräte hinweg verteilt ausgeführt wird.

4.1.1 Grundstruktur zum MUC-Controller

Die vorstehenden Kapitel bewirken von der logischen Struktur her folgende Grundaussprägung des MUC-Controllers:

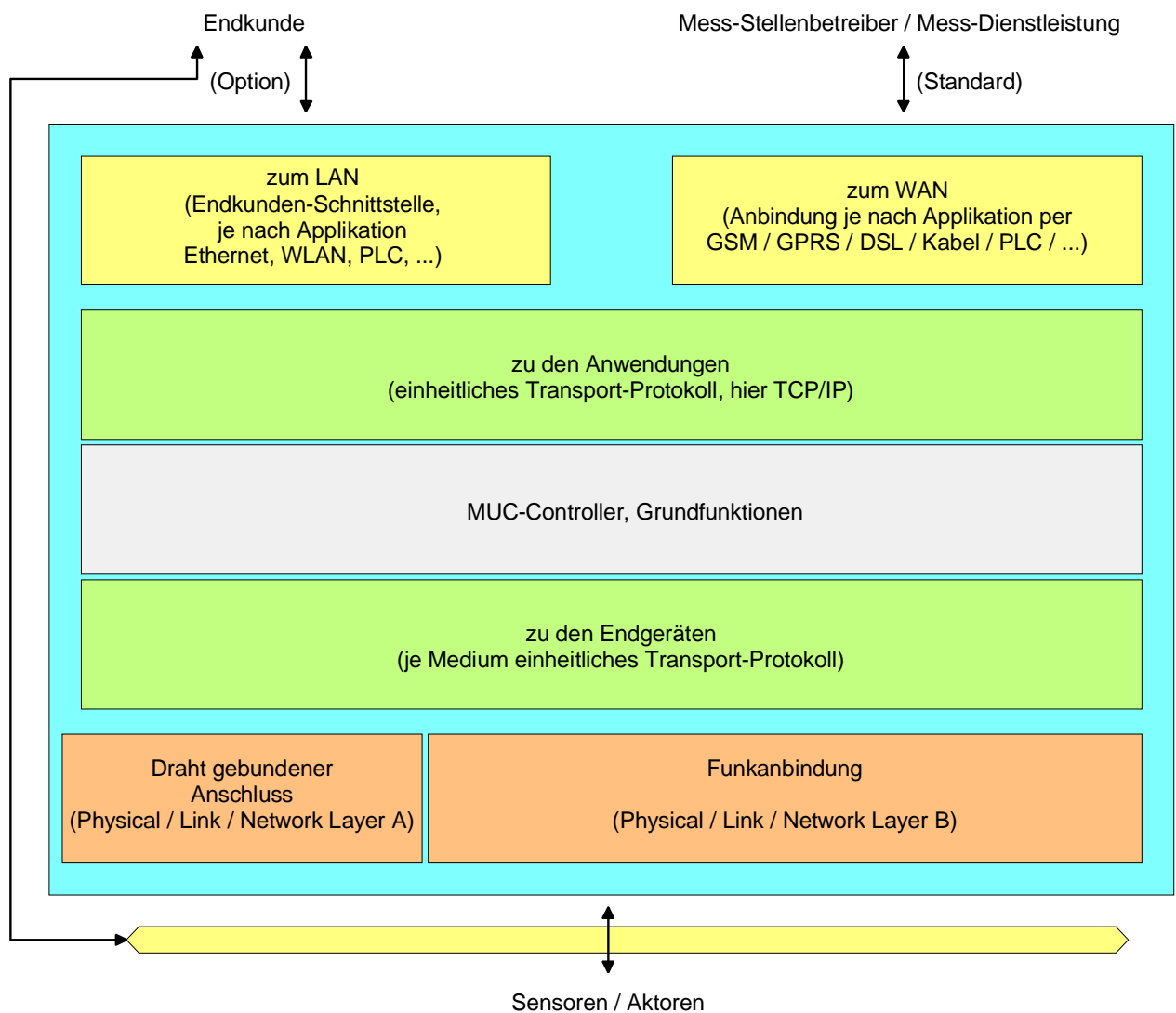


Bild 10: MUC-Controller, Grundform

4.1.1.1 Betriebssicherung

Der MUC-Controller ist mechanisch derart aufzubauen, dass eine Sicherung per Betriebsplombe ermöglicht wird⁶. Diese Sicherung muss bis auf die Schnittstelle zum Endkunden alle Elemente des MUC-Controllers abdecken.

4.1.1.2 Zeitbasis / Zeitserver / Betriebssekundenzähler

Ein MUC-Controller stellt im Sinne eines Servers die Zeitbasis für die an ihn angeschlossenen Sensoren / Aktoren bereit. Zu diesem Zweck muss er die Zeitbasis auf eine gesicherte Grundlage stellen (NTP⁷, DCF, GPS, oder qualitativ vergleichbare Führungsgröße).

Die Zeitbasis wird in UTC geführt.

Die Ganggenauigkeit ist mit mindestens 20 ppm zu gewährleisten.

Im MUC-Controller wird zusätzlich die Information zur Zeitzone (als Delta in Minuten) abgelegt.

Ergänzend wird im MUC-Controller der Sommerzeit-Offset vorgehalten.

Die aktuelle Zeitinformation, immer bestehend aus UTC, Zeitzone und Sommerzeit-Offset, kann jederzeit über die Datenschnittstellen abgefragt werden⁸.

Alle von einem MUC-Controller gebildeten Zeitstempel (z.Bsp. Logbuch, ...) bestehen immer aus dem Tripel UTC, Zeitzone und Sommerzeit-Offset.

- Die Interpretation der Bezugszeit für Datenspiegel und Push-Vorgänge wird für jeden Datenspiegel einmal definiert. Alle den Datenspiegel betreffenden zeitbezogenen Aktionen benutzen diese Bezugszeit. Die Bezugszeit kann folgende Varianten annehmen:
 - - UTC,
 - - UTC plus Zeitzone,
 - - UTC plus Zeitzone plus saisonaler Offset (\Leftrightarrow gesetzliche / lokale Zeit).

Zur Aufzeichnung von Datensammlern wird auf Abschnitt 7.3.2.43 verwiesen.

Zusätzlich zur Zeitbasis in UTC hält ein MUC-Controller einen Betriebssekundenzähler bereit (dieser wird beispielsweise zur Markierung der Logbucheinträge verwendet). Dieser wird, beginnend mit dem ersten Anlegen der Betriebsspannung im Herstellungsprozess bis zur Entsorgung des Gerätes fortlaufend jede Sekunde um 1 inkrementiert (streng monoton wachsendes Verhalten).

Die Zeitbasis muss per Gangreserve über mindestens 8 Stunden nach Wegfall der Versorgungsspannung aufrechterhalten werden. Bei Spannungswiederkehr nach Ausfall der Gangreserve werden ...

A. ... die Zeitbasis mit dem um mindestens 1 und maximal 300 Sekunde(n) gegenüber dem Wert bei Ausfall der Versorgungsspannung inkrementierten Zeitwert fortgeführt und im Statuswort das Merkmal

⁶ Wird der MUC-Controller als verteilte Funktionseinheit über mehrere Geräte realisiert, ist diese Forderung analog für die einzelnen Geräte anzuwenden.

⁷ Siehe [16].

⁸ Damit haben angeschlossene Sensoren / Aktoren die Option, selber zu entscheiden, welche der drei Zeitvarianten (rein UTC, UTC plus Zeitzone oder gesetzliche / lokale Zeit inkl. Sommer- / Winterzeit-Anteil) von ihnen benutzt werden.

‚Zeitbasis unsicher‘ gesetzt. Kann die Zeitbasis über die als Bezug verwendete Führungsgröße neu aktualisiert werden, wird das Merkmal ‚Zeitbasis unsicher‘ im Statuswort zurückgesetzt.

B. ... der Betriebssekundenzähler mit dem um mindestens 1 gegenüber dem Wert bei Ausfall der Versorgungsspannung inkrementierten Zahlenwert fortgeführt.

Erkennt ein MUC-Controller, dass die Zeitbasis nicht mehr gesichert⁹ bereitgestellt werden kann (etwa, weil die zu Grunde gelegte Führungsgröße über einen längeren Zeitraum nicht verfügbar war), wird das Merkmal ‚Zeitbasis unsicher‘ solange im Statuswort gesetzt, bis die Aktualisierung auf die Führungsgröße wieder erfolgt ist. Solange der Zustand ‚Zeitbasis unsicher‘ besteht, ist die Nachführung der Zeit in den an einem MUC-Controller angeschlossenen Sensoren / Aktoren zu unterbinden. Weiterhin sind in diesem Zustand alle neu aufgezeichneten Zeitstempel – nur bei der durch den MUC zu ergänzenden Zeitinformatio n – mit dem Wert „0xFFFFFFFF“ für das Element UTC zu markieren.

4.1.1.3 Schnittstelle zum Endkunden

Im Sinne einer Grundanforderung wird festgelegt:

- Der MUC-Controller enthält eine Schnittstelle zum Endkunden.
- Die Schnittstelle zum Endkunden ist rückwirkungsfrei auszuführen, so dass Manipulationsversuche nicht zu einer Funktionsbeeinträchtigung der Strecke Zähler / Sensoren / Aktoren zum WAN führen können.
- Die Schnittstelle zum Endkunden ist als RJ45, Ethernet mit Protokoll TCP/IP auszuführen.
- Die Schnittstelle zum Endkunden verfügt über zwei IP-Adressen die quasi gleichzeitig erreichbar sein müssen.
- Die erste der beiden IP-Adressen wird wahlweise wie folgt vergeben:
 - Automatischer Bezug der Netzwerkparameter per DHCP
(MUC-Controller arbeitet als DHCP-Client);
 - Automatische Vergabe der Netzwerkparameter per DHCP
(MUC-Controller arbeitet als DHCP-Server);
 - Manuelle Vergabe der IP-Adresse / Netzwerkparameter per Parametrierung.
- Die zweite der beiden IP-Adressen / Netzwerkparameter wird per Parametrierung fest vergeben.
- Die Schnittstelle verhält sich protokolltechnisch / inhaltlich im Applikationsprotokoll identisch zur WAN-Schnittstelle. Der vom Endkunden sichtbare Funktionsumfang wird über die Authentifizierung eingeschränkt.
- Der Datentransport erfolgt ohne IP-T direkt per TCP und ist mit SSL zu schützen.
- Optional kann in den MUC-Controller ein Webserver zum Zugriff / zur Parametrierung der Endkundenschnittstelle vorgesehen werden.
Der Zugriff auf den Webserver muss per Username / Passwort geschützt sein und über das Protokoll https ausgeführt werden.

⁹ Die Zeitbasis gilt als unsicher, wenn die Führungsgröße für mehr als 48 Stunden nicht zur Nachführung der Zeitbasis verwendet worden ist. Diese Zeitdauer kann per Parameter verändert werden (siehe Tab. 8).

4.1.1.4 Konstruktive Festlegungen

MUC-Controller, die zur Montage auf einer Standard-DIN-Schiene vorgesehen werden, sollten eine maximale Breite von 4 PLE (eine PLE entspricht 17,9 mm) aufweisen.

SIM-Karte:

Die SIM-Karte muss einführbar und entnehmbar sein, ohne das Gehäuse zu öffnen. Die mechanische Position wird nicht festgelegt, darf aber nicht im kundenzugänglichen Bereich liegen.

Belegung der Anschlußklemmen:

A = Phase der Versorgung

B = Null der Versorgung

C = Schalterausgang, intern einseitig verbunden mit B

D = Steuereingang, intern einseitig verbunden mit A

Anschlußquerschnitte der Anschlüsse A...D:

0,75...2,5 mm² ohne Aderendhülse

Antennenbuchsen:

M-BUS: Fakra, C, Signal blue (W-M-BUS 868MHz)

GSM: Fakra, D, Bordeaux Violett, (GSM / GPRS)

Mechanische Vorgaben für die Variante DIN-Schiene:

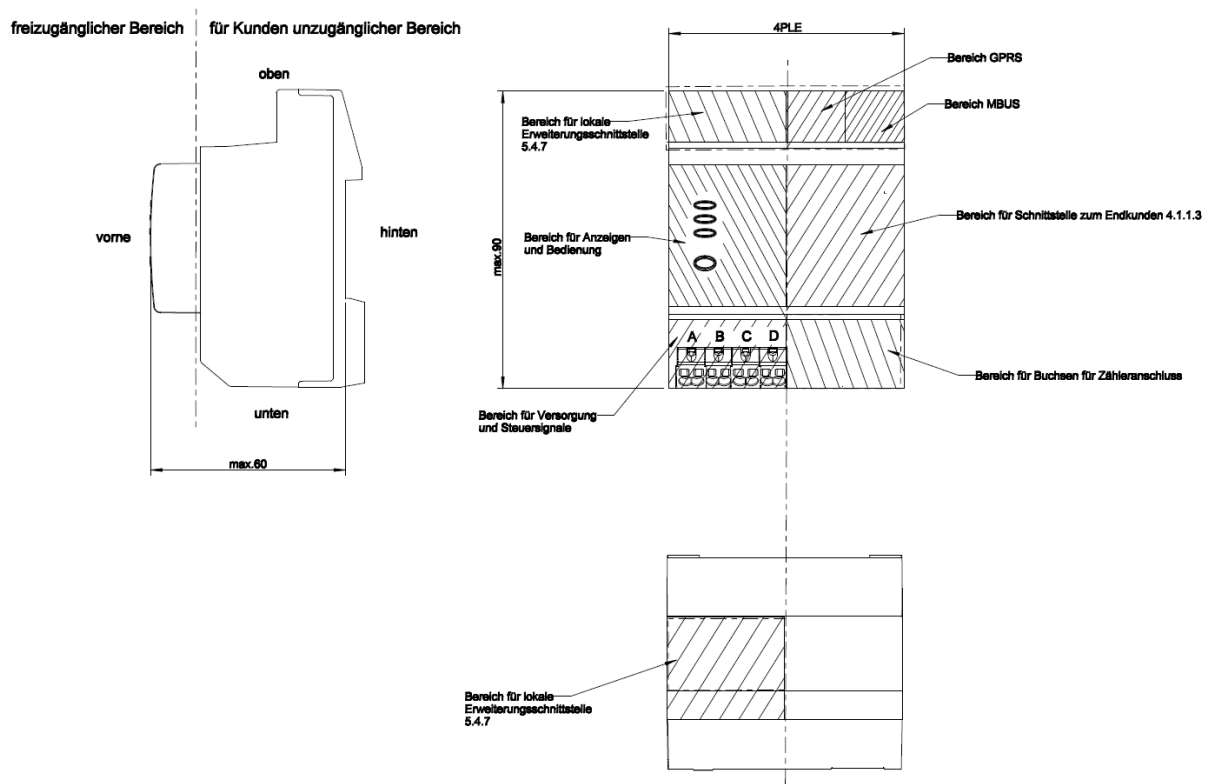


Bild 11: MUC-Controller, konstruktive Festlegung.

Andere konstruktive Ausführungen sind gleichermaßen denkbar.

5 Schnittstellen

Zur klaren Trennung der Zuständigkeiten definiert das MUC-Lastenheft eine Reihe von Schnittstellen. Schnittstellen, die aus Sicht der geforderten Eckpunkte zentrale Bedeutung für einen effizienten, operativen Betrieb haben, werden durch das MUC-Lastenheft im Detail festgelegt.

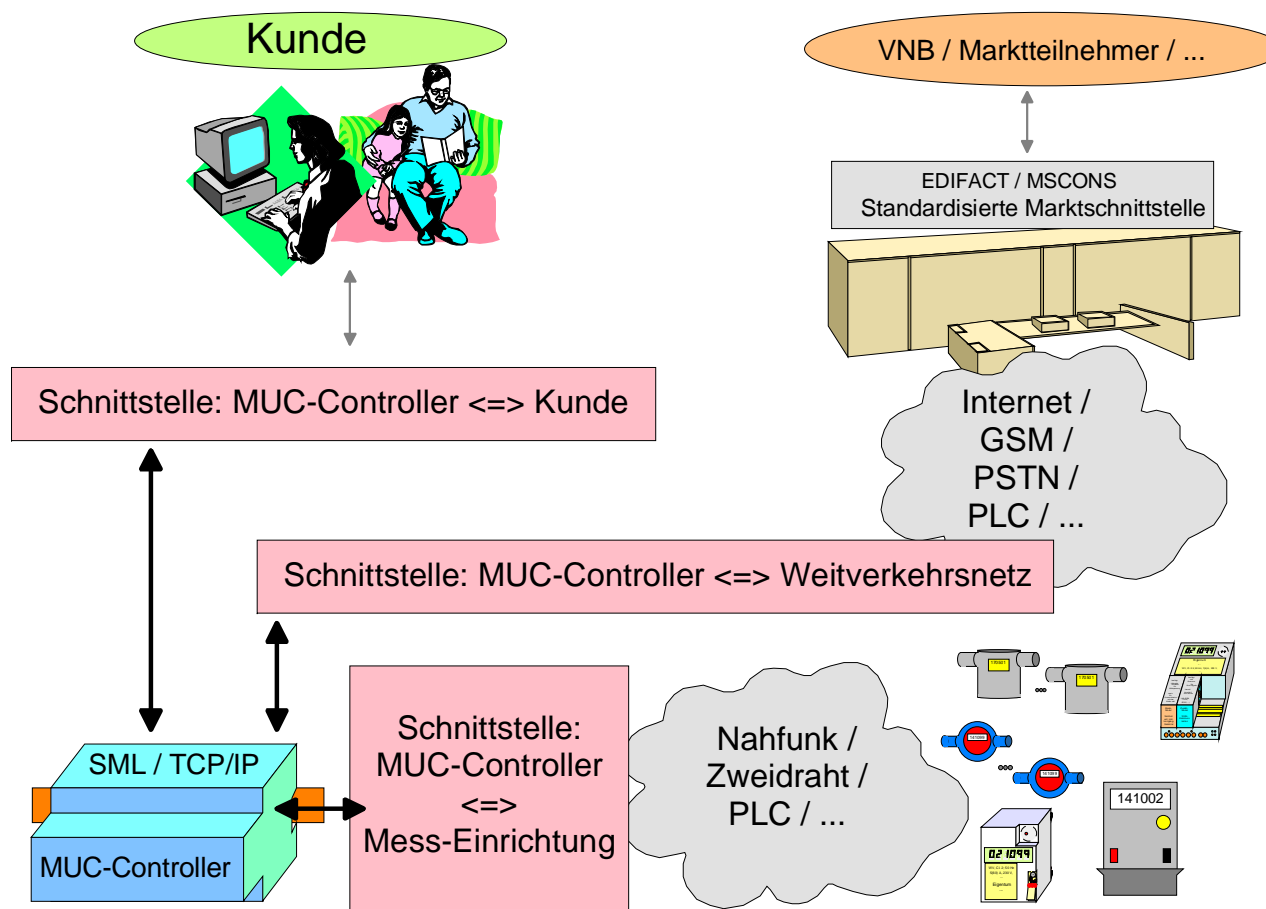


Bild 12: Schnittstellen und Systemumfeld

Pos.	Festlegung erfolgt in ...	Rubrik / Layer	Datenquelle	Datensenke
1	Kapitel 5.1 und 5.4	Application	Bereitstellung durch MUC-Controller	Eingang in der Zentrale
2	Kapitel 5.1	Presentation	Kodierung durch MUC-Controller	Interpretation durch Zentrale
3	Kapitel 5.4	Transport	Auslieferung durch MUC-Controller	Anlieferung in der Zentrale
4	Kapitel 5.1	Application	Bereitstellung durch Mess-Einrichtung	Eingang im MUC-Controller
5	Kapitel 5.1 und 5.1	Presentation	Kodierung durch Mess-Einrichtung	Interpretation durch MUC-Controller
6	Kapitel 5.1	Transport	Auslieferung durch Mess-Einrichtung	Anlieferung im MUC-Controller

Tab. 3: Durch MUC berührte Schnittstellen

5.1 Anforderungen an die Protokolle zur Anbindung der Sensorik / Aktorik

Um das in Bild 1 geforderte Verhalten der „einheitlichen“ Schnittstelle zur Anbindung der Zähler oder künftig ergänzend benötigter Sensorik / Aktorik an den MUC-Controller zu erreichen, müssen im Sinne des MUC-Lastenhefts folgende Rahmenbedingungen eingehalten werden:

- Die Protokolle sind so zu definieren, dass miteinander im Wettbewerb stehende Hersteller Geräte entwickeln, produzieren und liefern können, deren Austausch untereinander in ein und derselben Liegenschaft möglich wird, ohne die minimal geforderte Funktion des Gesamtsystems zu beeinträchtigen.
- Konkret wird damit die technische Kompatibilität für alle benötigten Bereiche des betroffenen Protokollstapels gefordert; der Einsatz proprietärer Lösungen für Physical / Link / Network / Transport Layer und die zu deren Einsatz benötigten Management Mechanismen wird abgelehnt.
- Das für die Anbindung der Geräte vor Ort zwischen Zähler / Sensor / Aktor zu definierende Nachrichten-Transportsystem muss in der Lage sein, aus Sicht MUC-Controller einheitlich Aufträge an die Endgeräte senden und deren Antwort ebenfalls wieder einheitlich entgegen nehmen zu können.
- Dabei erscheint es zulässig, zwischen zwei sich gegenseitig nicht störenden, aber im Detail unterschiedlich an die Erfordernisse der Bereiche Wasser / Wärme und Gas / Strom angepassten Ausführungen zu unterscheiden.

5.2 Vermaschung von Sensoren / MUC-Controllern

Die mit Version 1 zum MUC-Lastenheft primär verfolgte direkte Zuordnung von MUC-Controller und Wohneinheit kann nur als initialer Ansatz verstanden werden. Komplexere Installationen werden kurzfristig die Notwendigkeit mitbringen, zwischen den Sensoren und zwischen MUC-Controllern eine Art vermaschtes Netz zu verwenden. Die Funktionalität dieser Vermaschung wird an anderer Stelle auch als „Sekundär-Kommunikation“ bezeichnet.

Speziell zur Kostenreduktion in komplexen Installationen wird die Anzahl der MUC-Controller nicht Eins-zu-Eins mit der Anzahl der Wohneinheiten übereinstimmen. Hier wird man mehrere Wohneinheiten einem MUC-Controller zuordnen müssen.

Zum Betrieb derartiger Anordnungen sind zwei Aufgabenfelder zu definieren:

- Arbeitsweise / Protokoll zum vermaschten Netz;
- Handhabung der Benutzerverwaltung / Zugriffsrechte, so dass nur die Nutzer eine konkreten Wohneinheit auf die Messwerte dieser Wohneinheit zugreifen können.

Das MUC-Lastenheft soll auf diese Anordnungen erweitert werden, sobald erste Erfahrungen mit der Version 1 vorliegen.

5.3 Anwendungsebene

5.3.1 MwC – Messwert-Container

Zur Vereinheitlichung operativer Prozesse sowie zur Umsetzung rechtlicher Rahmenbedingungen im Umgang mit abrechnungsrelevanten Versorgungsgütern legt das MUC-Lastenheft einen standardisierten Datencontainer fest.

Dieser Datencontainer fasst im Sinne einer atomisierten Größe alle Informationen einer Mess-Einrichtung zusammen, die nur gemeinsam als Grundlage für die Weiterverarbeitung verwendet werden können. Er wird im Sensor gebildet und liegt damit unter der eichtechnischen Sicherung.

Der Datencontainer kann optional mit einer Signatur ausgestattet sein, um im Sinne der einleitend zitierten rechtlichen Rahmenbedingungen Authentizität und Integrität des Inhalts zu sichern.

Falls Applikationen zusätzlich die Privatsphäre des Inhalts gewährleisten müssen, sind ergänzende Schutzmechanismen (beispielsweise der Einsatz von VPN auf der Weitverkehrsstrecke) auf den jeweiligen Kommunikationsnetzen anzuwenden.

Die Zielsetzung kann dargestellt werden mit:

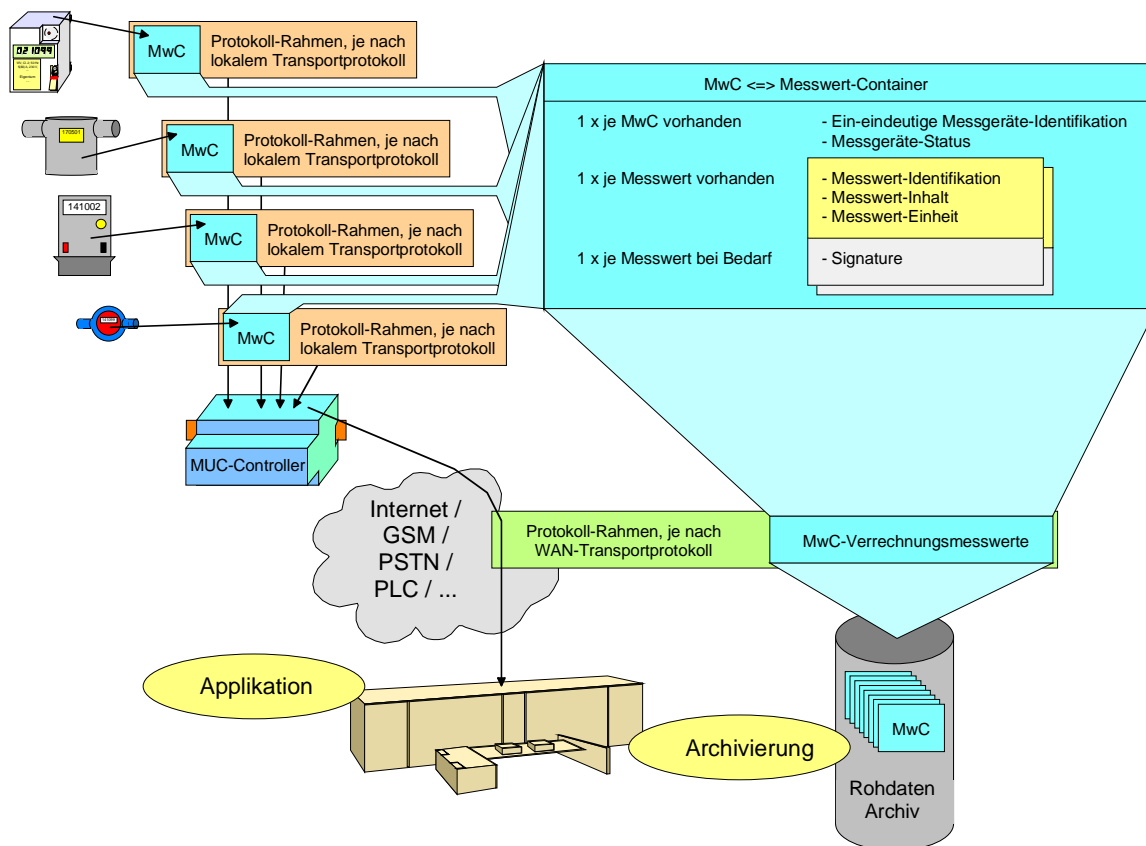


Bild 13: Konzept zum MwC – Messwert-Container

Der MwC standardisiert die in ihm zusammengefassten Inhalte wie folgt:

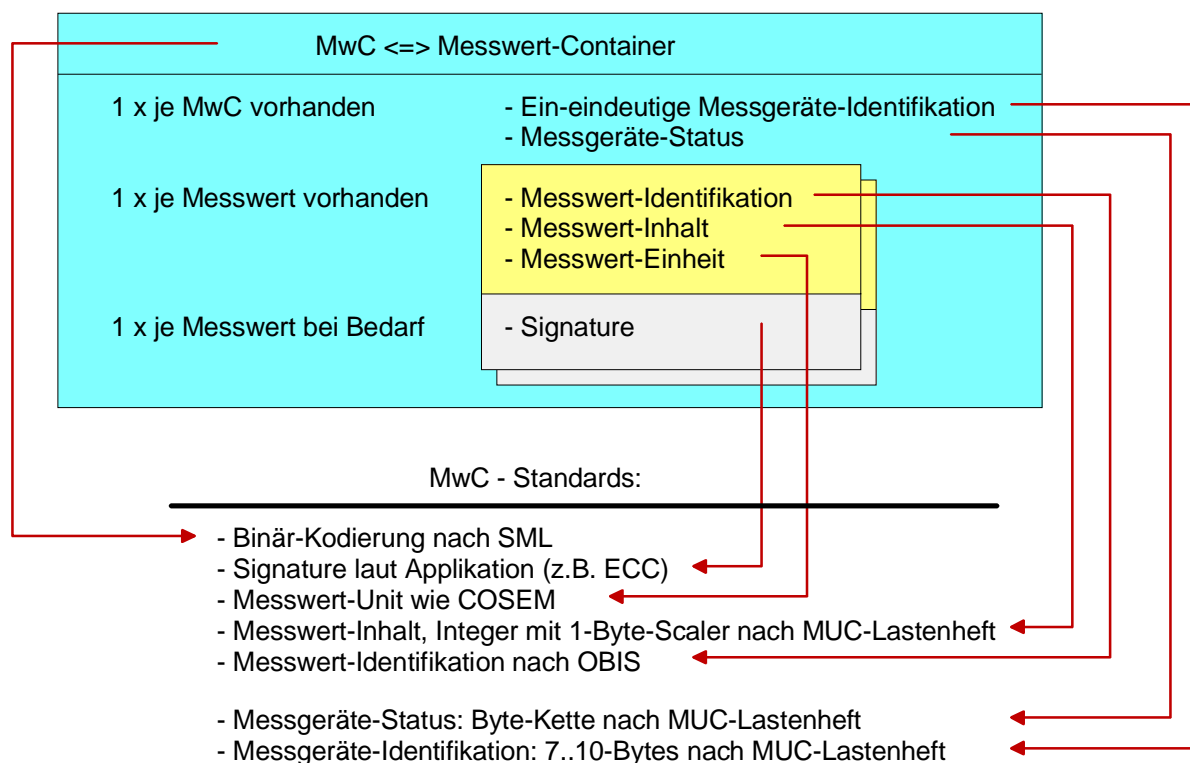


Bild 14: Standards im MwC – Messwert-Container

5.3.1.1 MwC, Messwert-Container, Kodierung

Der MwC wird mittels ‚SML_GetList‘ (siehe SML [12]) kodiert.

5.3.2 MUC, optionale Signatur

Das jeweils verwendete Verfahren soll mit den für jede Sparte spezifischen Detailspezifikationen zu den Sensoren / Aktoren definiert werden.

5.3.2.1 Krypto-Algorithmus

Für Stromzähler:

- Der zu verwendende Krypto-Algorithmus ist gemäß FNN-Lastenheft zum eHZ zu wählen.

Für Gaszähler:

- Künftig noch zu ergänzen

Für Wärmezähler:

- Künftig noch zu ergänzen

Für Wasserzähler:

- Künftig noch zu ergänzen

Für Heizkostenverteiler:

- Künftig noch zu ergänzen

5.3.2.2 Signaturbildung

Für Stromzähler:

- Die Signaturbildung ist gemäß FNN-Lastenheft zum EDL-Zähler zu wählen.

Für Gaszähler:

- Künftig noch zu ergänzen

Für Wärmezähler:

- Künftig noch zu ergänzen

Für Wasserzähler:

- Künftig noch zu ergänzen

Für Heizkostenverteiler:

- Künftig noch zu ergänzen

5.3.3 MUC, Datenstrukturen

- Ein MUC-Controller muss über die im Anhang beschriebenen Datenstrukturen parametrierbar sein.

5.3.3.1 Datenstrukturen zur Zählerstandsabfrage / Abfrage von Datenlisten**Auftrag / SML-Request:**

- Zur Abfrage des Zählerstands ist eine SML-Auftragsdatei mit folgenden SML-Nachrichten zu erzeugen:
- SML-Open-Request (siehe SML-Spezifikation);
- SML_GetList-Request (siehe SML-Spezifikation);
- SML-Close-Request (siehe SML-Spezifikation);

Antwort / SML-Response:

- Als Antwort zur Abfrage des Zählerstands ist eine SML-Antwortdatei mit folgenden SML-Nachrichten zu erzeugen:
- SML-Open-Response (siehe SML-Spezifikation);
- SML_GetList-Response (siehe SML-Spezifikation);
- SML-Close-Response (siehe SML-Spezifikation).

5.3.3.2 Datenstrukturen zur Tarifumschaltung

Über den MUC-Controller können Aufträge / Abfragen zur Tarifumschaltung von der WAN-Schnittstelle an die Sensoren / Aktoren weitergeleitet werden.

Diese Aufträge / Abfragen sind gemäß SML zu kodieren und müssen in der vom konkreten Sensor / Aktor jeweils spezifischen Applikationsstruktur strukturiert werden.

5.3.3.3 Datenstrukturen zur Ansteuerung von Sensoren / Aktoren

Sollen künftig spezielle Sensoren / Aktoren über den MUC-Controller angebunden werden, werden die zu deren Einsatz benötigten Datenstrukturen hier zu definieren sein.

5.3.3.4 Benutzerverwaltung / Berechtigungskonzept

An die Benutzerverwaltung / das Berechtigungskonzept werden folgende, generell zu erfüllenden, Anforderungen gestellt:

- Die in den Datenstrukturen eines MUC-Controllers genannten Inhalte zu Passwörtern dürfen niemals durch irgendeinen Zugriff ausgelesen werden können. Das Auslesen ist als Ausnahme in verschlüsselter Form zulässig, falls ein Passwort verschlüsselt abgelegt ist und die Rückführung der verschlüsselten Darstellung auf den Klartext unmöglich ist.

- Die mit den Datenstrukturen definierten Elemente zu Benutzernamen und Passwort müssen durch den MUC-Controller derart verwaltet werden, dass Änderungen nur nach Freigabe durch eine jeweils berechnigte Instanz (die Berechnigung erfolgt üblicherweise durch die korrekte Eingabe von Benutzernamen und Passwort des für die Verwaltung / Parametrierung benutzten Zugangs) möglich sind.
- In der Grundeinstellung sind alle Zugriffe abzuweisen, es sei denn, der Zugriff wurde explizit per Parametrierung gemäß der nachfolgend beschriebene Strukturierung erlaubt.

Zur Strukturierung von Zugriffsberechtigungen definiert das MUC-Lastenheft 8 Rollen:

- (0) Gastzugang;
- (1) Endanwender;
- (2) MUC-Betreiber;
- (3) Mess-Stellenbetreiber;
- (4) Mess-Dienstleister;
- (5) Lieferanten;
- (6) reserviert für künftig benötigte Rollen;
- (7) Hersteller.

Die Strukturierung zur Vergabe von Zugriffsberechtigungen ist wie folgt gegeben:

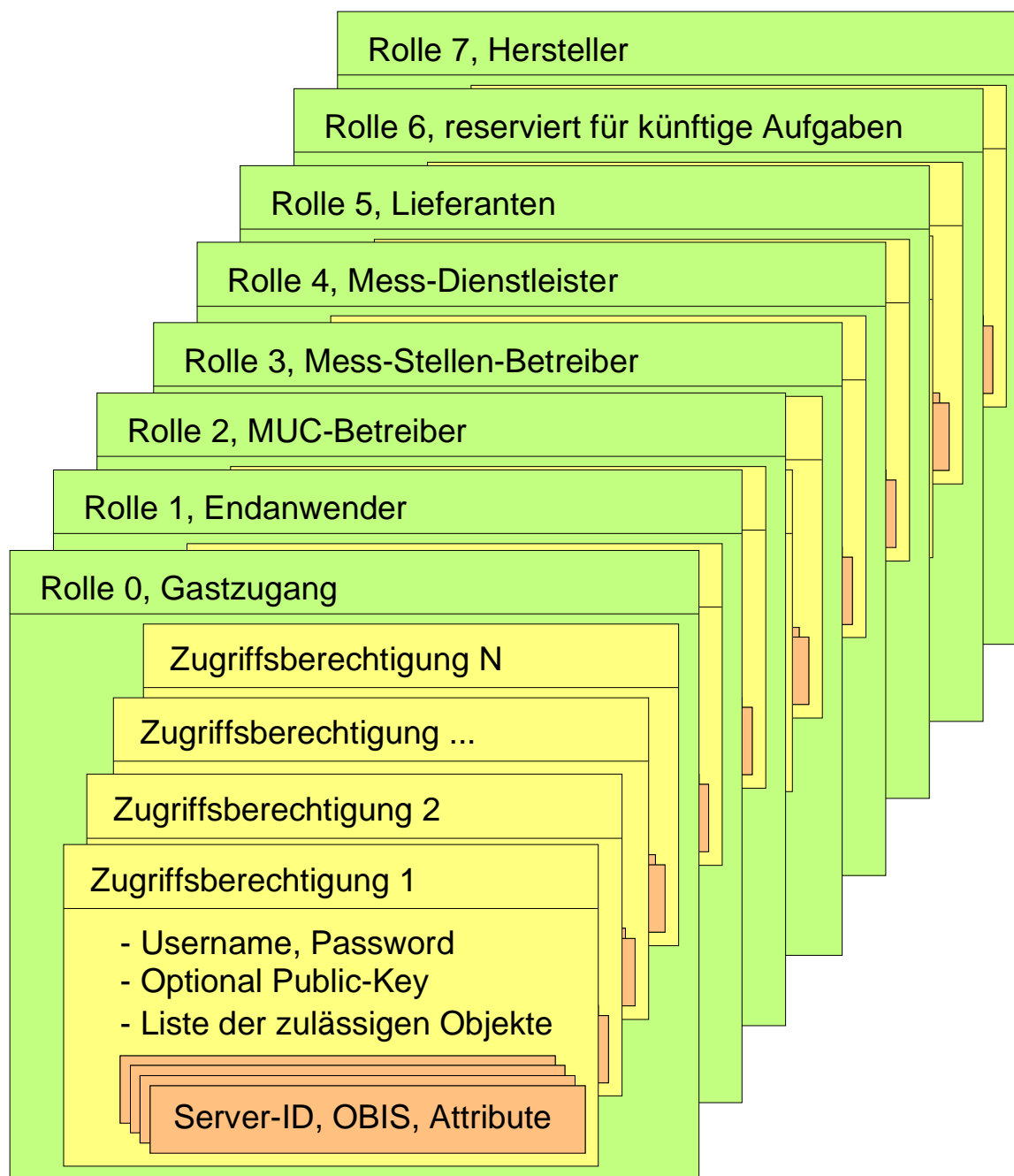


Bild 15: Berechtigungskonzept

5.4 Transportebene

5.4.1 Verwendung von TCP/IP bei MUC

Soweit TCP/IP als Protokolltechnologie für einen MUC-Controller herangezogen wird, ist IP in der Version IPv4 anzuwenden. Optional kann ein MUC-Controller zusätzlich auch IP in Version IPv6 anbieten.

Auf einer TCP/IP-Schnittstelle müssen mindestens 2 TCP-Sockets gleichzeitig zu öffnen sein.

Ein geöffneter TCP-Socket (MUC-Controller als TCP-Server) ist bei ausbleibendem Datentransport durch den MUC-Controller eigenständig zu schließen und für eine erneute Benutzung freizugeben. Als Zeiträume werden gefordert:

- Endkundenschnittstelle: 15 Minuten

- WAN-Schnittstelle: Derzeit keine Definition, da ein MUC-Controller an der WAN-Schnittstelle immer ein TCP-Client ist.
- Erweiterungsschnittstelle: Siehe SyM²

Einer Penetration an der TCP/IP-Schnittstelle (Denial of Service) ist durch geeignete Maßnahmen entgegenzuwirken.

5.4.2 Verwendung von SML bei MUC

Wegen der in SML gegebenen Mechanismen zur Unterscheidung von ‚Responses‘ und ‚Responses without Request‘ kann der MUC – für die umgebende Applikation vollkommen transparent – sowohl in Push- als auch in Pull-Betriebsarten verwendet werden. Im Vergleich zur Betriebsart Pull fällt bei Push lediglich die Notwendigkeit weg, eine Nachricht vom Typ ‚SML_GetList.Req‘ an die Datenquelle vorab zu senden. Beide Betriebsarten bewirken letztlich die Übertragung des MUC als Nachricht vom Typ ‚SML_GetList.Req‘ an die Applikation.

Als Server-ID wird die mit Kapitel 7.2 definierte Kodierung verwendet.

Falls bei SML-Nachrichten das Element ‚actTime‘ vorhanden ist, ist darin der Betriebssekundenzähler des MUC mit dem zum Zeitpunkt der Ausgabe aktuellen Wert einzutragen.

Werden IPv4-Adressen in SML transportiert, sind diese als Unsigned32 anzugeben. IPv6-Adressen werden als Octet-String in binär kodierter Form (also nicht als ASCII-String mit den Doppelpunkten als Trennzeichen) benutzt.

IPv4-Adressen, zu kodieren als Unsigned32:

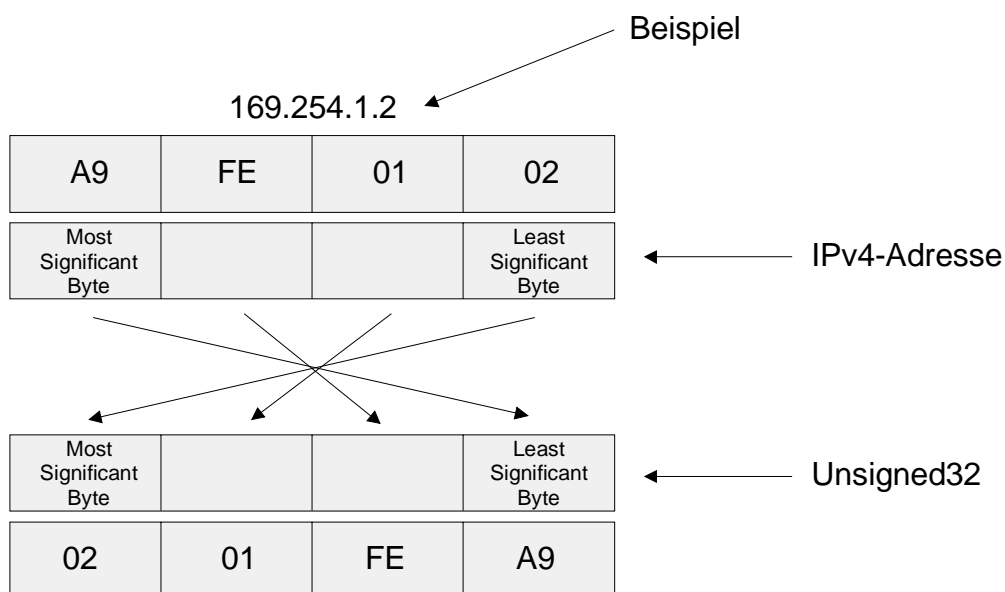


Bild 16: Kodierung von IPv4-Adressen

IPv6-Adressen, zu kodieren als Octet-String:

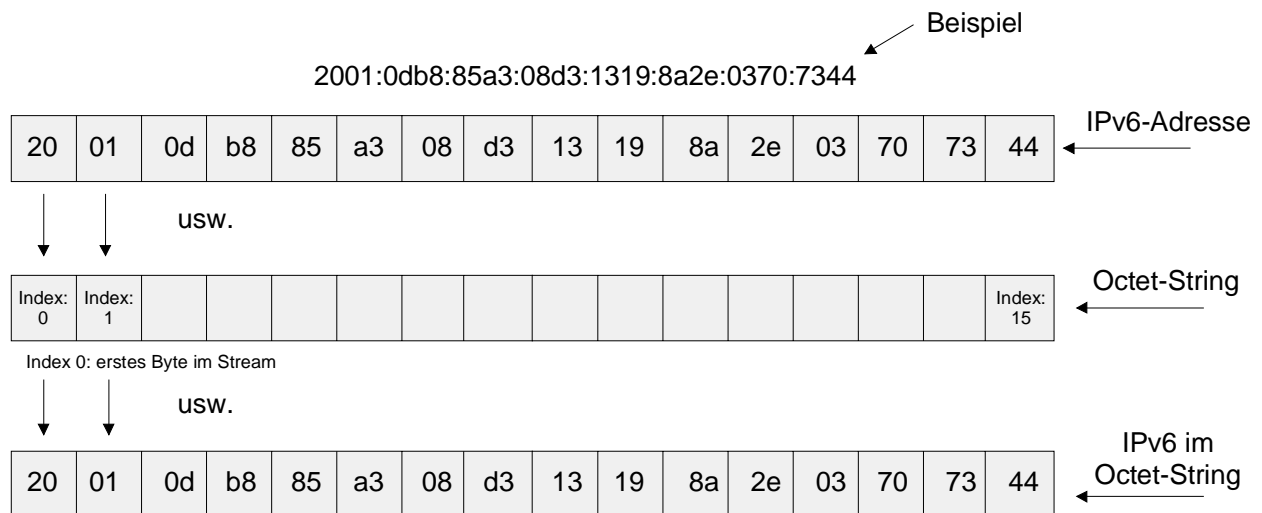


Bild 17: Kodierung von IPv6-Adressen

Bei Zugriff auf Datenstrukturen sind folgende Regeln anzuwenden:

- Beim lesenden Zugriff auf das Wurzel-Element einer Datenstruktur ist der Baum aller Subelemente auszugeben. Falls ein Subelement als Wirkung der Zugriffsrechte nicht gelesen werden darf, ist das Lesen der ganzen Datenstruktur mit ‚SML_Attention‘ abzulehnen.
- Beim schreibenden Zugriff auf das Wurzel-Element einer Datenstruktur können im Baum einzelne Subelemente weggelassen werden. Weggelassene Subelemente behalten ihre vorherigen Wert oder sind, falls der ganze Baum neu angelegt wird, mit ihrem Defaultwert zu initialisieren. Falls ein Subelement als Wirkung der Zugriffsrechte nicht geschrieben werden darf, ist der Schreibvorgang der ganzen Datenstruktur mit ‚SML_Attention‘ abzulehnen.
- Der schreibende Zugriff auf das Wurzelement einer Datenstruktur ohne Angabe von Subelemente führt zum Löschen / Entfernen der Datenstruktur (entsprechende Zugriffsrechte und Funktion vorausgesetzt). Anderenfalls ist eine Fehlermeldung zu erzeugen.

5.4.3 Weitverkehrsschnittstelle

Installationen, die den Vorgaben des MUC-Lastenhefts folgen, bieten an der Weitverkehrsschnittstelle zwischen MUC-Controller und Leitstelle folgende Mechanismen:

- Der MUC-Controller enthält eine Weitverkehrsschnittstelle.
- Als Transporttechnologie wird TCP/IP eingesetzt.
- Für die physikalischen Transportwege können alle für TCP/IP geeigneten Verfahren zum Einsatz kommen.
- Anwendungen, die besonderen Schutz gegen Manipulation benötigen, sollten auf die Konzepte zu VPNs (Virtual Private Networks) der jeweiligen Telekommunikations-Service-Provider zurückgreifen.
- Als Schnittstelle zwischen der allgemein verfügbaren Transporttechnologie und den von MUC speziell getroffenen Festlegungen wird TCP vereinbart.

- Der für den Einsatz von TCP zu verwendende Port wird durch den Anwender / Betreiber festgelegt.
- Zur Lösung der beim Einsatz von privaten wie auch dynamischen IP-Adressen entstehenden Problematik wird IP-Telemetrie (siehe E DIN 43863-4 [6]) angewendet.
- Aus IP-Telemetrie werden für die Übertragungen von SML-Dateien die Variante „IP-T-Link“ für ‚Pull‘ und „IP-T-Push“ für ‚Push‘ genutzt. SML-Dateien werden immer per Rahmen nach SML-Transportprotokoll der Version 1 gekapselt.
- Die zu verwendenden SML-Dateien genügen der mit SML [12] getroffenen Spezifikation inklusive der Detaillierung des MUC-Lastenhefts.

5.4.4 SML als Application-Layer für Nahfunk- oder PLC-Systeme in MUC-Anwendungen

Bei Einsatz als Application-Layer für Nahfunk- oder PLC-Systeme werden an Stelle vollständiger SML-Dateien nur SML-Nachrichten übertragen, falls ausschließlich SML-Nachrichten vom Typ „Get-List“ eingesetzt sind. In diesem Fall sind durch den umgebenden Link-Layer Anfang und Ende der SML-Nachricht eindeutig zu markieren. Sollen andere SML-Nachrichten benutzt werden, ist eine vollständige SML-Datei (inklusive SML-Open / SML-Close) zu benutzen.

Andererseits bietet diese Vorgehensweise verschiedene Vorteile:

- Auf die Transformation von Application-Layer-Inhalten kann verzichtet werden, da diese vom Sensor bis in die Leitstelle identisch kodiert werden.
- Innerhalb des MUC-Controllers müssen die von den Sensoren eingehenden SML-Nachrichten lediglich mit dem Rahmen einer SML-Datei ergänzt werden.
- Auf der Nahfunk- oder PLC-Strecke wird unnötiger Overhead reduziert.

5.4.5 MUC, Funkschnittstelle an der Mess-Einrichtung

Ein MUC-Controller enthält immer eine Nahfunkschnittstelle zur Anbindung lokaler Mess-Einrichtungen.

Der bei Einsatz batteriegestützter Funktechnik problematische Kompromiss zwischen Langzeitnutzung (≥ 10 Jahre), zulässigem Duty-Cycle für den Funkkanal und pro Zeiteinheit verfügbarer Energiemenge für die Datenübertragung erzwingt eine Lösung, die speziell für die Funkstrecke optimierte Datentelegramme / Konzepte anwendet.

Vor diesem Hintergrund wird festgelegt:

- Als Physical-Layer wird M-Bus (EN 13757-4 [5] / Open-Metering in FIGAWA/ZVEI [15]) verwendet.
- Als Link-Layer wird M-Bus (EN 13757-4 [5] / Open-Metering in FIGAWA/ZVEI [15]) verwendet.
- Die Auswahl der konkret zu verwendenden Betriebsart wird mit Tab. 4 definiert.
- Als Application-Layer wird SML in der mit Kapitel 5.4.4 definierten Ausprägung eingesetzt.
- Die über Funkschnittstellen transportierten Informationen sind zu verschlüsseln.

Zur Auswahl der für die Nahfunkstrecke zu verwendenden Betriebsart wird gefordert:

Pos.	Geräteart	S-Mode / T-Mode
1	MUC-Controller	Beide Varianten müssen unterstützt werden. Der konkret zu verwendende Modus ist per Parametrierung festzulegen.
2	Unidirektional sendende, „einfache“ Sensoren, beispielsweise HKV oder Wasserzähler	Alternativ kann das Gerät immer S-Mode oder immer T-Mode verwenden.
3	Komplexe, bidirektional arbeitende Endgeräte, beispielsweise Gas- oder Stromzähler	Das Gerät kann immer S-Mode oder immer T-Mode verwenden. Optional können hybride Geräte (diese können im Betrieb dynamisch zwischen S-Mode und T-Mode wechseln) realisiert werden. An hybride Geräte werden ergänzende Anforderungen gestellt.

Tab. 4: Betriebsart für Nahfunkstrecke (S-Mode/ T-Mode)

5.4.5.1 Ergänzende Anforderungen für hybride Geräte

Konkrete Anforderungen werden ergänzt, sobald das Systemverhalten näher bekannt sein wird.

5.4.6 Vermaschtes Funknetz

Die Definition wird ergänzt, sobald das dafür benötigte Protokoll verfügbar ist.

5.4.7 MUC, Leitungsgebundene Schnittstelle, lokale Erweiterungsschnittstelle

Werden Mess-Einrichtungen per Leitung an den MUC-Controller angeschlossen, sind die auf dieser Strecke zu verwendenden Transportprotokolle derart zu wählen, dass auf eine Transformation der über die Weitverkehrsstrecke geführten Datenstrukturen verzichtet werden kann.

Ein MUC-Controller muss in der Lage sein, über leitungsgebundene Schnittstellen geführte Informationen zu verschlüsseln.

Ein MUC-Controller enthält als leitungsgebundene Schnittstelle zur Ansteuerung von Mess-Einrichtungen oder anderen Sensoren / Aktoren, wie auch zur Verbindung von weiteren MUC-Controllern eine Ethernetschnittstelle. Diese ist gemäß SyM² auszuführen. Sie arbeitet als PoE-Senke, falls keine andere Speisung verwendet wird. Optional kann ein MUC-Controller auch die Variante PoE-Quelle anbieten. Wird die PoE-Funktion nicht verwendet, dürfen die für PoE genutzten Kontaktpunkte der Ethernet-Buchse nicht beschaltet werden.

5.4.7.1 MUC, Einsatz bei Mess-Einrichtungen mit M-Bus-Schnittstelle

Ein MUC-Controller kann optional eine leitungsgebundene M-Bus-Schnittstelle enthalten. Diese ist entweder integriert oder kann per Adapter an die lokale Erweiterungsschnittstelle angeschlossen werden.

5.4.7.2 MUC, Einsatz bei Mess-Einrichtungen mit Stromzählern und EDL-Zähler-Schnittstelle

Ein MUC-Controller kann optional eine leitungsgebundene EDL-Zähler-Schnittstelle enthalten. Diese ist entweder integriert oder kann per Adapter an die lokale Erweiterungsschnittstelle angeschlossen werden.

Die elektrische / mechanische Verbindung zwischen MUC-Controller und EDL-Zähler ist wie folgt auszuführen:

- Der MUC-Controller stellt einen Anschluss nach reduzierter RS232 mit folgenden elektrischen Parametern bereit:
 - Hilfsspannung Pegel nach V.28, minimal 25 mA,
In Diskussion:
Pegel nach V.28, minimal 5 mA @ 5 V und begrenzt auf maximal 20 mA @ 15 V
positive Spannung gegenüber Gnd
 - TX Pegel und Stromquelle nach V.28,
 - RX Pegel und Stromsenke nach V.28,
 - Gnd gemeinsame Masse.

Dieser Anschluss wird zur Anbindung der rückwärtigen, optischen Datenschnittstelle der EDL-Zähler verwendet. Über die vorstehend definierten elektrischen Eigenschaften wird die optische Schnittstelle versorgt.

- Die Steckvorrichtung ist als RJ10 auszuführen (siehe Bild 18, Blickrichtung in die RJ10-Buchse des MUC-Controller)

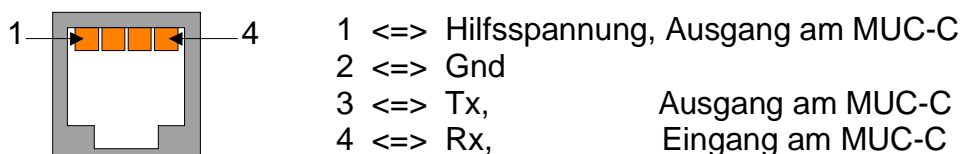


Bild 18: PIN-Belegung zum RJ10

Damit können eHZ direkt per SML angesprochen werden; eine Transformation der Datentelegamme im MUC-Controller ist nicht notwendig.

Im Einzelnen sind für eine konkrete Implementation folgende Parameter zu verwenden:

Pos.	Parameter	Festlegung
1	Protokoll	SML (siehe Lastenheft eHZ)
2	Maximale Latenzzeit zur Bearbeitung einer SML-Auftragsdatei (gemessen ab Ende der Übertragung der SML-Auftragsdatei bis zum Beginn der SML-Antwortdatei)	2 Sekunden
3	Baudrate	9600 Baud (siehe Lastenheft eHZ)
4	Bitformat	8 N 1

Tab. 5: Parameter zur Ausführung der Datenschnittstelle für EDL-Anbindung

Verwendung von PLC als Transportmedium

Es ist beabsichtigt, die Festlegungen des MUC-Lastenheftes bei Bedarf künftig auch auf den Einsatz der PLC-Technik zu erweitern.

Die Erweiterung wird vorgenommen werden, sobald eine akzeptable Standardisierung im Bereich PLC erkennbar ist.

Die PLC-Schnittstelle kann optional integriert oder per Adapter an die lokale Erweiterungsschnittstelle angebunden werden.

6 Anwendungsfälle

- Das MUC-Lastenheft legt Rahmenparameter fest, deren Ziel die Gewährleistung der Interoperabilität von Geräten verschiedener Hersteller ist. Konkrete Entwicklungen werden zwangsläufig auf diesem Lastenheft aufbauend mit der Ausarbeitung eines Pflichtenhefts beginnen.
- Damit Hersteller, die für ihre konkrete Entwicklung Pflichtenhefte auf Basis des MUC-Lastenhefts erstellen, möglichst gesichert vorgehen können, werden nachfolgend ausgewählte Anwendungsfälle beschrieben, die Hinweise zur Handhabung der Geräte liefern.

6.1 Inbetriebnahmeprozess / Installationsprozess

- Die Definition wird ergänzt, sobald die Anforderungen / beteiligten Partner näher genannt werden können.

7 Anhang

7.1 Statuswert

Ein MUC-Controller enthält ein Statuswort, dessen Bit-Bedeutung wie folgt gegeben ist:

Pos.	Merkmal		Festlegung
1	Bit 0	(LSB)	Immer ,0', kennzeichnet Statuswort-Bedeutung
2	Bit 1		Immer ,1', kennzeichnet Statuswort-Bedeutung
3	Bit 2		Immer ,0', kennzeichnet Statuswort-Bedeutung
4	...		Immer ,0', kennzeichnet Statuswort-Bedeutung
5	Bit 7		Immer ,0', kennzeichnet Statuswort-Bedeutung
6	Bit 8		Gesetzt auf ,1', wenn ein ,Fataler Fehler' erkannt wird; Rücksetzen nur per SML-Kommando durch Schreiben mit ,0'
7	Bit 9	Y	Gesetzt auf ,1', wenn ein Watchdog ausgelöst und danach die Betriebsbereitschaft erreicht worden ist; Rücksetzen nur per SML-Kommando durch Schreiben mit ,0'
8	Bit 10	Y	Gesetzt auf ,0', wenn Transportkanal verfügbar; Bei PSTN: Link mit Peer besteht; Bei GSM: Link mit Peer besteht; Bei GPRS: Erfolgreiche Anmeldung am GPRS-Dienst; Bei LAN/DSL: DHCP- und / oder PPPoE-Parameter erfasst; Rückgesetzt auf ,1', wenn Transportkanal nicht verfügbar ist; Keine Veränderung per SML-Schreib-Kommando
9	Bit 11	Y	Gesetzt auf ,0', wenn Weitverkehrsnetz verfügbar ist; (bei PSTN: Modem erkennt TAE-Spannung, bei GSM/GPRS: Modem erkennt Funknetz, bei LAN/DSL: Ethernet-Link vorhanden) Rückgesetzt auf ,1', wenn Weitverkehrsnetz nicht verfügbar ist; Keine Veränderung per SML-Schreib-Kommando
10	Bit 12	Y	Gesetzt auf ,0', wenn Anmeldung an Funknetz erfolgt ist; Bei PSTN: reserviert, immer ,0'; Bei GSM/GPRS: gesetzt auf ,0', wenn Anmeldung an Funknetz erfolgt ist; Bei LAN/DSL: reserviert, immer ,0'; Keine Veränderung per SML-Schreib-Kommando
11	Bit 13	Y	Gesetzt auf ,0', wenn IP-T-Anmeldung erfolgt ist; Bei PSTN: reserviert, immer ,0'; Bei GSM: reserviert, immer ,0'; Bei GPRS: gesetzt auf ,0', wenn IP-T-Anmeldung erfolgt ist; Bei LAN/DSL: gesetzt auf ,0', wenn IP-T-Anmeldung erfolgt ist; Keine Veränderung per SML-Schreib-Kommando
12	Bit 14	Y	Gesetzt auf ,1', falls bei der Ausführung ein ,Out of Memory' erkannt wird; Rücksetzen nur per SML-Kommando durch Schreiben mit ,0'

13	Bit 15		Reserviert, gesetzt auf ‚0‘
14	Bit 16	Z	Gesetzt, wenn die Endkunden-Ethernet-Schnittstelle vorhanden ist
15	Bit 17	Z	Gesetzt, wenn die Erweiterungs-Ethernet-Schnittstelle vorhanden ist
16	Bit 18	Z	Gesetzt, wenn die Wireless-M-Bus-Schnittstelle vorhanden ist
17	Bit 19	Z	Gesetzt, wenn die PLC-Schnittstelle vorhanden ist
18	Bit 20	Z	Reserviert, gesetzt auf ‚0‘
19	...	Z	Reserviert, gesetzt auf ‚0‘
20	Bit 31	Z	Reserviert, gesetzt auf ‚0‘
21	Bit 32		Gesetzt auf ‚1‘, wenn die Zeitbasis im MUC unsicher ist (siehe Kapitel 4.1.1.2); Rücksetzen nur durch Nachführen der Zeitbasis im MUC.
22	Bit 33		Reserviert, gesetzt auf ‚0‘
23	...		Reserviert, gesetzt auf ‚0‘
24	Bit 63 (MSB)		Reserviert, gesetzt auf ‚0‘

Tab. 6: Statuswort zum MUC-Controller

Änderungen der mit „Y“ gekennzeichneten Statusbits werden im Betriebslogbuch eingetragen; sie haben eine Entsprechung in der Liste der Ereignisse.

Die mit „Z“ markierten Statusbits sind als bit-kodierter Bereich aufzufassen, dessen Inhalt die in dem MUC-Controller vorhandenen Schnittstellen auflistet.

7.2 MUC, einheitlicher Adressierung

Zur einheitlichen Adressierung von MUC-Controllern wird folgendes Verfahren angewendet, dessen Zielsetzung ein Lösungsweg zur Vereinheitlichung der verschiedenen heute im Feld angewendeten Identifikationsverfahren ist:

Pos.	Byte 0 (MSB)	Adressierungstyp	Kodierung
1	01	8-Byte Wireless-M-Bus	Binärkodierung gemäß wireless M-Bus-Standard: Die 8 Bytes der Wireless-M-Bus-Adresse werden direkt im Anschluss an das Byte 0 (MSB) angeordnet.
2	02	8-Byte, Wired-M-Bus	Binärkodierung gemäß wired M-Bus-Standard: Die 8 Bytes der Wired-M-Bus-Adresse werden direkt im Anschluss an das Byte 0 (MSB) angeordnet.
3	03	9-Byte, Rhein-Energie	18-Stellen, BCD: Die 18 Stellen der Identifikationsnummer werden direkt im Anschluss an das Byte 0 (MSB) in BCD-Darstellung angeordnet.
4	04	7-Byte, E.ON	16-Stellen, E-ON Spezifikation: Die 16 Stellen der Identifikationsnummer werden in deren Binäräquivalent transformiert und direkt im Anschluss an das Byte 0 (MSB) als Kette von 7 Bytes angeordnet.

5	05	6-Byte, Ethernet-MAC-Adresse	<p>6-Byte Ethernet-MAC-Adresse:</p> <p>Bei vorhandener Erweiterungsschnittstelle ist die MAC-Adresse der Erweiterungsschnittstelle zu verwenden.</p> <p>Fehlt die Erweiterungsschnittstelle und ist eine Ethernet-Endkundenschnittstelle vorhanden, ist deren MAC-Adresse zu verwenden.</p> <p>Die 6 Bytes der MAC-Adresse werden direkt im Anschluss an das Byte 0 (MSB) angeordnet.</p>
6	06	DKE, E DIN 43863-5:2010-02	<p>16-Stellen, gemäß Normentwurf E DIN 43863-5 aus 02/2010:</p> <p>Der Normentwurf benutzt 3 alphanumerische Zeichen für die Identifikation des Herstellers. Diese 3 alphanumerischen Zeichen werden in die direkt auf das Byte 0 (MSB) folgenden 3 Bytes abgebildet.</p> <p>Im Anschluss werden die 13 dezimal Stellen, transformiert in deren Binäräquivalent, als Kette von 6 Bytes angeordnet.</p> <p>Die Kodierung erfolgt gemäß Bild 19</p>
7	07	7-Byte, IMEI	<p>15-Stellen, dezimal, gemäß IMEI:</p> <p>Die 15 Stellen der Identifikationsnummer werden in deren Binäräquivalent transformiert und direkt im Anschluss an das Byte 0 (MSB) als Kette von 7 Bytes angeordnet.</p>
8	08	7-Byte, RWE	<p>14 Stellen, RWE-Spezifikation: (xxxxxx-yyyyyy)</p> <p>Die Identifikationsnummer wird auf eine Byte-Kette mit 7 Bytes (Byte 1 bis Byte 7) abgebildet. Byte 1 schließt sich direkt an das Byte 0 (siehe Spalte MSB) an. Die Transformation auf die Bytes 1 – 7 (LSB) der Byte-Kette ist wie folgt durchzuführen:</p> <p>Die hinteren 7 Stellen (,yyyyyy') werden in ihre Binär-Darstellung umgeformt und in den Bytes 5-7 (also am Ende der Byte-Kette) abgelegt.</p> <p>Das Byte 4 wird mit dem ASCII-Code zu „-“ besetzt.</p> <p>Die ersten 6-Stellen (,xxxxxx') werden in ihre Binär-Darstellung umgeformt und in den Bytes 1-3 abgelegt.</p>
9	09	DKE, E DIN 43863-5:2010-07	<p>14-Stellen, gemäß Normentwurf E DIN 43863-5 aus 07/2010.</p> <p>Die Kodierung erfolgt gemäß Bild 20.</p>
10	0A	-	Reserviert für künftige Zwecke
11	...	-	Reserviert für künftige Zwecke
12	7E	-	Reserviert für künftige Zwecke

13	7F	Sub-Adressierung zur Kombination mit allen anderen Adressierungsarten dieser Tabelle	<p>Zur Identifikation von Unter-Instanzen (z. Bsp. für rein in Software realisierte MUC-Controller {„virtuelle MUC“} auf einem einzigen physikalischen Gerät) kann jeder anderen in dieser Tabelle eine aus 3 Bytes bestehende Sub-Adresse vorangestellt werden.</p> <p>Die Sub-Adresse wird mit der Kennung ‚0x7F‘ eingeleitet. Diesem Byte folgen zwei weitere Bytes in der Reihenfolge High-Byte, dann Low-Byte. Diese beiden Bytes legen die Sub-Adresse fest. Damit können maximal 65536 Sub-Adressen unterschieden werden.</p> <p>Der Zahlenwert ‚0‘ (also die Byte-Kette ‚0x7F 0x00 0x00‘) in der Sub-Adresse adressiert alle Unter-Instanzen. Er stellt damit die Broadcast-Adresse für die Unter-Instanzen dar.</p> <p>Beispiel für die Sub-Adressierung (Adresse: ‚1‘) in Verbindung mit der 6-Byte-MAC-Adresse (Adresse ‚0x10 0x11 0x12 0x13 0x14 0x15‘):</p> <p>0x7F 0x00 0x01 0x05 0x10 0x11 0x12 0x13 0x14 0x15</p>
14	80	-	Reserviert für künftige Zwecke
15	...	-	Reserviert für künftige Zwecke
16	FF	-	Reserviert für künftige Zwecke

Tab. 7: Kodierung zur ein-eindeutigen Adressierung der Sensoren / Aktoren

Damit können Server-Identifizierer maximal 10 (oder 13 bei Einsatz der Sub-Adressierung) Bytes enthalten.

Ein-eindeutige Geräte-Identifikation (DKE-Entwurf aus Feb. 2010),
Kodierung als Server-ID bei EDL21-Zählern sowie MUC-Controllern

Bezug: E DIN 43863-5:2010-02

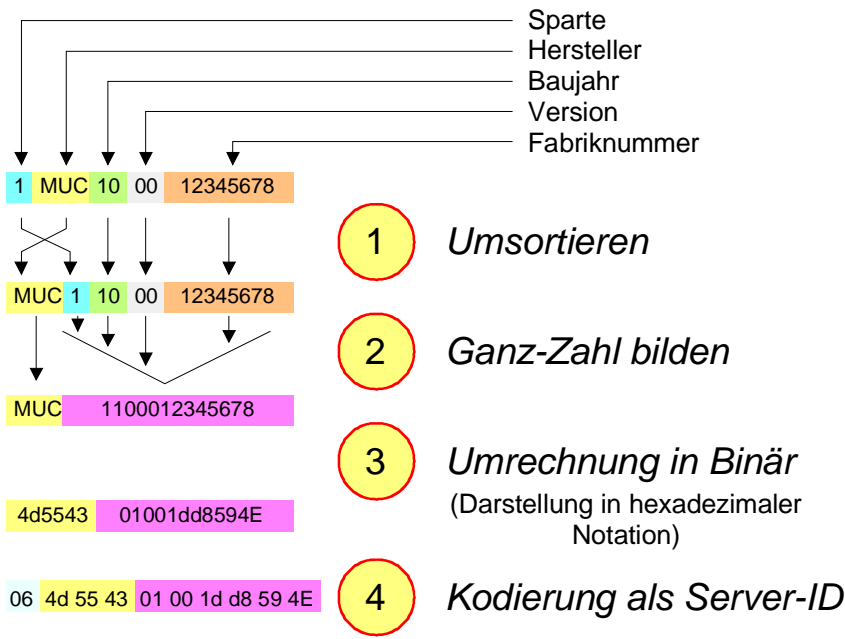


Bild 19: Kodierung für die Adressierungs-Variante E DIN 43863-5:2010-02

Ein-eindeutige Geräte-Identifikation (DKE-Entwurf aus Juli 2010),
Kodierung als Server-ID bei EDL21-Zählern sowie MUC-Controllern

Bezug: E DIN 43863-5:2010-07

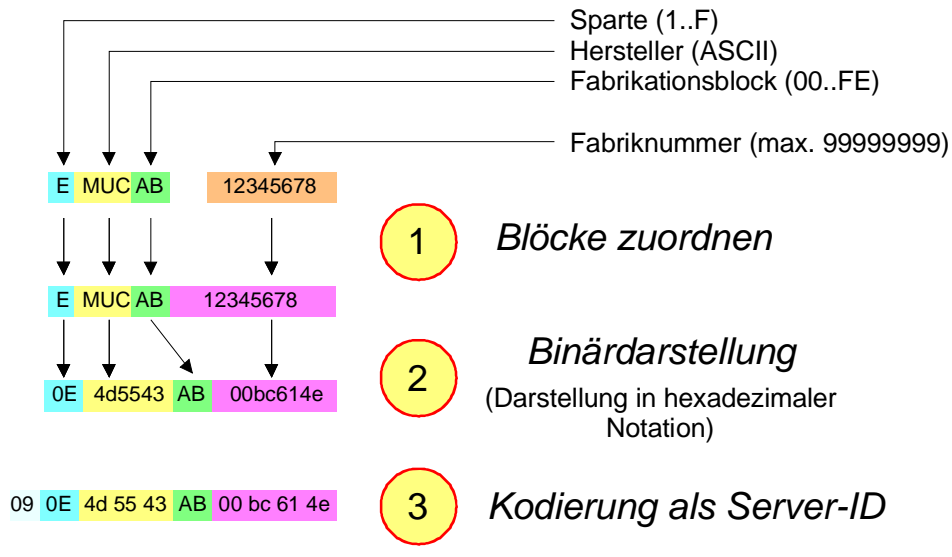


Bild 20: Kodierung für die Adressierungs-Variante E DIN 43863-5:2010-07

7.2.1 Data-Matrix-Code

Zu Darstellung der benutzten Adressierungsparameter – und möglicherweise ergänzender Informationen – auf dem Leistungsschild kann ein Data-Matrix-Code benutzt werden. Wird die Darstellung per Data-Matrix-Code benutzt, sind folgende Festlegungen zu beachten:

Aufbau einer Informationszeile im Data-Matrix-Code:

<Tag_Char_1>	<Tag_Char_2>	Payload (0..N ASCII-Zeichen)	<CR>	<LF>
--------------	--------------	------------------------------	------	------

Anforderungen:

- Der zulässige Zeichensatz wird mit ISO 8859-15 definiert.
- Die vier Rahmenzeichen (<Tag_Char_1>...<LF>) sind immer vorhanden.
- Zulässige Kombinationen für die beiden Tags <Tag_Char_1> und <Tag_Char_2> werden im zuständigen Gremium gepflegt und veröffentlicht.
- Bei den zulässigen Kombinationen sollte ein herstellerspezifisch nutzbarer Bereich eingefügt werden.
- Für die Payload sind ausschließlich Zeichen des Wertebereichs '0x20'..'0x7E' zugelassen.

Bild 21: Kodierung einer Datenzeile im Data-Matrix-Code

7.3 MUC-Datenstrukturen zur Parametrierung

7.3.1 Datenstrukturen zur Parametrierung der MUC-Controller-Funktionen

7.3.1.1 Direkte Parameter

Ein MUC-Controller erlaubt den direkten Zugriff auf folgende Eigenschaften, wobei je nach konkreter Variante einzelne Eigenschaften fehlen können:

Pos.	Kennzahl	Schreiben erlaubt	Opt.	Variante	Datentyp	Bedeutung
1	81 81 27 32 07 01	Ja	-	GSM / GPRS / PLC	Integer16	Periode zur Aufzeichnung der Einträge in das Betriebslogbuch. Wird hier ‚0‘ angegeben, wird die Aufzeichnung abgeschaltet. Durch Angabe von ‚-1‘ kann das Betriebslogbuch gelöscht werden.
2	00 00 60 08 00 FF	Nein	-	Alle	SML_Time	Betriebssekundenzähler
3	81 00 60 05 00 00	Ja	-	Alle	Unsigned64	Statuswort (per Schreiben ist das Rücksetzen ausgewählter Statusbits)
4	81 KK 00 00 01 00 Spalte KK ⇔ 03 Spalte KK ⇔ 04 Spalte KK ⇔ 01	Nein	-	Alle	Octet String	Schnittstellename; (Anzahl und Inhalt abhängig von Anzahl und Art der Schnittstellen im MUC-Controller).
5	01 00 00 09 0B 00	Ja	-	Alle	<ul style="list-style-type: none"> • Typ: SML_Time • Inhalt: UTC 	Aktuelle UTC Time
6	81 00 00 09 0B 01	Ja	-	Alle	<ul style="list-style-type: none"> • Typ: Integer16 • Inhalt: Offset in Minuten Wertebereich: -720 bis +720	Offset zur aktuellen Zeitzone
7	81 00 00 09 0B 20	Ja	-	Alle	<ul style="list-style-type: none"> • Typ: Unsigned16 • Inhalt: Stunden 	Periode, innerhalb derer die Zeitbasis nachgeführt werden muss, bevor die Zeit eigenständig in den Zustand ‚unsicher‘ wechselt

Tab. 8: Parameter für den direkten Zugriff

7.3.1.2 Datenstruktur zur Parametrierung der Sommerzeit- / Winterzeitwechsel

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Festlegung der Liste jahresbezogener Sommerzeit- / Winterzeitwechsel	81 00 00 09 0B 10	Nicht vorhanden	Ein oder mehrere Eintrag / Einträge der Position 2 dieser Tabelle
2	Kind-Element im Baum Jahresbezogener, an einen Stichtag gebundener, Zeitwechsel	81 00 00 09 0B NN	Nicht vorhanden	Ein oder mehrere Eintrag / Einträge der Position 3 bis 7 dieser Tabelle
3	Kind-Element im Baum Offset zur Berechnung der aktuellen Zeit, der nach dem Zeitwechsel zu benutzen ist	81 00 00 09 0B 03	<ul style="list-style-type: none"> Integer16 Der Wert wird in Minuten angegeben. 	
4	Kind-Element im Baum Monat, in der Zeitwechsel erfolgen soll	81 00 00 09 0B 04	<ul style="list-style-type: none"> Unsigned8 Wertebereich: 1 ⇔ Januar, 2 ⇔ Februar, ..., 12 ⇔ Dezember. 	
5	Kind-Element im Baum Stichtag im Monat, nach dem der Zeitwechsel erfolgen soll	81 00 00 09 0B 05	<ul style="list-style-type: none"> Unsigned8 Wertebereich: 1 .. 31 	
6	Kind-Element im Baum Wochentag, an dem der Zeitwechsel erfolgen soll.	81 00 00 09 0B 06	<ul style="list-style-type: none"> Unsigned8 Wertebereich: 0 ⇔ Der Stichtag ist direkt zu benutzen. 1 ⇔ Der erste Montag ist zu benutzen. ... 7 ⇔ Der erste Sonntag ist zu benutzen. 	
7	Kind-Element im Baum Minute (gerechnet ab 00:00 UTC des betroffenen Stichtags), an der der Zeitwechsel erfolgen soll.	81 00 00 09 0B 07	<ul style="list-style-type: none"> Integer16 Der Wert wird in Minuten angegeben. 	

Tab. 9: Parameter für die Festlegung der Sommerzeit- / Winterzeitwechsel
7.3.1.3 Datenstruktur zur Parametrierung der Rollen / Benutzerrechte

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Einstellung der Benutzerrechte / Rollen	81 81 81 60 FF FF	Nicht vorhanden	Ein oder mehrere Eintrag / Einträge der Position 2 dieser Tabelle
2	Kind-Element im Baum	81 81 81 60 NN FF	Nicht vorhanden	Ein oder mehrere Eintrag / Einträge

Rolle	NN legt die Rolle fest, Wertebereich: 0x01 - 0x08	der Position 3 dieser Tabelle
3 Kind-Element im Baum Zugriffsberechtigung	81 81 81 60 NN ZZ NN legt die Rolle fest, Wertebereich: 0x01 - 0x08 ZZ legt die Zugriffsberechtigung fest, Wertebereich: 0x01 – 0xFE	Nicht vorhanden Einträge der Positionen 4-7 dieser Tabelle, wobei Position 6 optional ist
4 Kind-Element im Baum Benutzername zum Zugriff	81 81 81 61 FF FF	• Octet String
5 Kind-Element im Baum Passwort zum Zugriff (Siehe Festlegung zum Benutzerkonzept: Das Lesen der Eigenschaft ist nur insofern zulässig, als der gelesene Wert verschlüsselt auszugeben ist. Zum Schreibzugriff ist der Klartext zu verwenden.)	81 81 81 62 FF FF	• Octet String
6 Kind-Element im Baum Public-Key für Authentifizierung beim Zugriff auf diese Instanz; diese Angabe darf fehlen, wenn der Zugriff ohne Signaturschutz erfolgen darf	81 81 81 63 FF FF	• Octet String
7 Kind-Element im Baum Liste Zugriffsberechtigungen je Server-Identifizier	81 81 81 64 01 SS Erste Server-ID-Gruppe: SS legt den Server-Identifizier fest, Wertebereich: 0x01 – 0xFE	• Octet String Server-Identifizier Ein oder mehrere Eintrag / Einträge nach Pos. 8 dieser Tabelle
8 Kind-Element im Baum OBIS-Kennzahl der Instanz, auf die die Zugriffsberechtigung erteilt werden soll	< OBIS-Kennzahl der Instanz > Wird hier das Wurzel-Element oder die OBIS-Kennzahl zu einem Ast angegeben, gelten die Rechte für alle Elemente des Astes, es sei denn, für ein darunter liegendes Element / ein Ast werden andere Angaben getroffen.	• Unsigned8 Bit 0 ⇔ Lesen Bit 1 ⇔ Schreiben Bit 2 ⇔ Ausführen Alle anderen Bits sind auf ‚0‘ zu setzen.

Tab. 10: Parameter zur Einstellung der Zugriffsrechte

7.3.1.4 Datenstruktur zum Lesen/Setzen der Parameter für die Endkundenschnittstelle

Das Lesen und Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig. Es erfolgt per SML_GetProcParameter oder SML_SetProcParameter.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element der Datenstruktur zum Lesen/Setzen der Eigenschaften der Schnittstelle zum Endkunden	81 02 00 07 00 FF	Nicht vorhanden	
2	Kind-Element im Baum. Bezug der ersten, eigenen IP-Adresse	81 02 00 07 01 FF	<ul style="list-style-type: none"> • Unsigned8 Inhalt: 0 ⇔ Manuelle Auswahl 1 ⇔ DHCP-Client Alle anderen Werte sind reserviert.	
3	Kind-Element im Baum. Erste, eigene, manuell gesetzte, IP-Adresse	81 02 17 07 00 01	Unsigned32 (IPv4) Octet String (IPv6)	
4	Kind-Element im Baum. Erste, eigene Netzwerkmaske	81 02 17 07 01 01	Unsigned32 (IPv4) Octet String (IPv6)	
5	Kind-Element im Baum. DHCP Server	81 02 00 07 02 FF	<ul style="list-style-type: none"> • Boolean Inhalt: TRUE ⇔ DHCP-Server aktiviert (in diesem Fall muss eine zugehörige, manuell gesetzte IP-Adresse vergeben werden)	
6	Kind-Element im Baum. DHCP-Server, lokale Netzwerkmaske	81 02 00 07 02 01	Unsigned32 (IPv4) Octet String (IPv6)	
7	Kind-Element im Baum. DHCP-Server, Default Gateway	81 02 00 07 02 02	Unsigned32 (IPv4) Octet String (IPv6)	
8	Kind-Element im Baum. DHCP-Server, DNS Server	81 02 00 07 02 03	Unsigned32 (IPv4) Octet String (IPv6)	
9	Kind-Element im Baum. DHCP-Server, Anfangsadresse des dynamischen IP-Adresspool	81 02 00 07 02 04	Unsigned32 (IPv4) Octet String (IPv6)	
10	Kind-Element im Baum. DHCP-Server, Endadresse des dynamischen IP-Adresspool	81 02 00 07 02 05	Unsigned32 (IPv4) Octet String (IPv6)	
11	Kind-Element im Baum. Zweite, eigene, manuell gesetzte, IP-Adresse	81 02 17 07 00 02	Unsigned32 (IPv4) Octet String (IPv6)	

12	Kind-Element im Baum. Zweite, eigene Netzwerkmaske	81 02 17 07 01 02	Unsigned32 (IPv4) Octet String (IPv6)
----	---	-------------------	--

Tab. 11: Parameter zur Einstellung der Endkundenschnittstelle**7.3.1.5 Datenstruktur für dynamischen Eigenschaften der Endkundenschnittstelle**

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig. Es erfolgt per SML_GetProcParameter.

Die Elemente können nicht beschrieben werden.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element der Datenstruktur zum Lesen des Status der Schnittstelle zum Endkunden	81 02 00 07 10 FF	Nicht vorhanden	Genau ein Eintrag der Pos. 2 und 3
2	Kind-Element im Baum. Aktuelle eigenen IP-Adresse	81 02 17 07 00 00	Unsigned32 (IPv4) Octet String (IPv6)	
3	Kind-Element im Baum. Aktuelle eigenen zweite IP-Adresse	81 02 17 07 00 02	Unsigned32 (IPv4) Octet String (IPv6)	

Tab. 12: Status der Endkundenschnittstelle**7.3.1.6 Datenstruktur zum Lesen/Setzen der Parameter für die Erweiterungsschnittstelle**

Derzeit werden keine Parameter (Ausnahme: Kapitel 7.3.1.8) zur Nutzung der Erweiterungsschnittstelle (siehe Kap. 5.4.7) benötigt. Falls vorhanden, ist die Erweiterungsschnittstelle gemäß SyM² (siehe Tab. 1, Pos. SyM²) auszuführen.

7.3.1.7 Datenstruktur für dynamischen Eigenschaften der Endkundenschnittstelle

Derzeit sind keine dynamischen Parameter zur Nutzung der Erweiterungsschnittstelle (siehe Kap. 5.4.7) verfügbar.

7.3.1.8 Datenstruktur allgemeiner Schnittstellen-Eigenschaften

Das Lesen oder Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig. Es erfolgt per SML_GetProcParameter / SML_SetProcParameter.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element der Liste aller Schnittstellen im MUC	81 81 C7 90 00 FF	Nicht vorhanden	Mindestens ein Listenelement der Position 2
2	Konten-Element im Baum. Identifikation der Schnittstelle	81 81 C7 90 00 NN 01 .. 1F ⇔ Index zur Auflistung der Schnittstellen	Octet String OBIS-Kennzahl	
3	Kind-Element im Baum. Schnittstellen-Name	81 81 C7 90 01 FF	Octet String	

4	Kind-Element im Baum. Schnittstellen-Status	81 81 C7 90 02 FF	Boolean TRUE ⇔ Schnittstelle aktiviert
---	--	-------------------	--

Tab. 13: Allgemeine Schnittstellen-Eigenschaften.

Die Liste hat alle Schnittstellen des MUC aufzulisten. Dies betrifft auch alle Schnittstellen, die nicht mit diesem Lastenheft definiert, in einem konkreten Gerät aber zusätzlich vorhanden sind.

Schnittstellen, die mit diesem Lastenheft definiert sind, sollen folgenden Schnittstellen-Namen benutzen:

- LAN/DSL: 81 48 17 07 00 FF LAN-DSL
- GSM: 81 04 02 07 00 FF GSM
- GPRS: 81 04 0D 07 00 FF GPRS
- Endkundenschnittstelle: 81 02 00 07 00 FF USER
- IPT-Schnittstelle: 81 49 0D 07 00 FF IP-T
- EDL-Zähler-Schnittstelle: 81 05 0D 07 00 FF MSB-EDL
- Wireless M-BUS: 81 06 19 07 00 FF wM-BUS
- PLC: 81 04 18 07 00 FF PLC (<Typ¹⁰>)
- Erweiterungsschnittstelle: 81 05 0D 07 00 FF SyM²

7.3.1.9 Datenstruktur zur Abfrage des WAN Status

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Abfrage des WAN-Status	81 04 00 06 00 FF	Nicht vorhanden	Einträge der Positionen 2-3
2	Kind-Element im Baum Aktuelle Schnittstellenname / Typ, kodiert als ASCII-String, zum verwendeten Weitverkehrsadapter.	81 04 00 00 01 00	• Octet String	
3	Kind-Element im Baum Firmwareversion, kodiert als ASCII-String, zum verwendeten Weitverkehrsadapter.	81 04 00 02 00 00	• Octet String	

Tab. 14: Parameter zum WAN-Status

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

Das Schreiben der Elemente ist nicht möglich.

¹⁰ Vom Hersteller je nach Technologie festzulegen.

7.3.1.10 Datenstruktur zum Lesen/Setzen der WAN Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Lesen/Setzen der WAN Parameter	81 04 00 07 00 FF	Nicht vorhanden	Einträge der Positionen 2-4
2	Kind-Element im Baum Automatischer Reboot: Periode, anzugeben in Sekunden, nach deren Ablauf das Modem / der Weitverkehrsadapter im KM-Modul neu initialisiert werden soll. Bei ‚0‘ ist der automatische Reboot inaktiv.	81 04 27 32 03 01	• Unsigned32	
3	Kind-Element im Baum Maximales Inter Message Timeout in Sekunden	81 42 64 3C 01 01	• Unsigned8	
4	Kind-Element im Baum Maximales Timeout zwischen SML_Close-Request und SML_Open-Response in Sekunden für die Datenrichtung Request von der Leitstelle zum MUC	81 42 64 3C 01 04	• Unsigned16 Untere Grenze vom Wertebereich legt der Hersteller fest.	
5	Kind-Element im Baum Maximales Timeout zwischen SML_Close-Request und SML_Open-Response in Sekunden für die Datenrichtung Request vom MUC an die Leitstelle	81 42 64 3C 01 05	• Unsigned16	

Tab. 15: Datenstruktur mit WAN-Parametern

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.11 Datenstruktur zum Lesen/Setzen der GSM Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Lesen/Setzen der GSM Parameter	81 04 02 07 00 FF	Nicht vorhanden	Einträge der Positionen 2-8
2	Kind-Element im Baum PIN zum Zugriff auf die SIM-Karte.	81 04 00 32 01 01	Octet String	
3	Kind-Element im Baum Festlegung der Betriebsart zur Provider-Auswahl. 0: Automatisch (SIM)	81 04 00 32 04 01	Unsigned8	

	1: Benutzerliste			
	2: Erst Benutzerliste, dann automatisch (SIM)			
4	Kind-Element im Baum GSM Bearer Service Type	81 04 00 32 08 01	Unsigned8	
5	Kind-Element im Baum GSM Quality of Service	81 04 00 32 09 01	Unsigned8	
6	Kind-Element im Baum Maximale Dauer einer GSM-Verbindung, anzugeben in Sekunden. Bei ,0' wird die Funktion auf inaktiv gesetzt.	81 04 27 32 01 01	Unsigned16	
7	Kind-Element im Baum Idle-Time einer GSM-Verbindung, anzugeben in Sekunden. Bei ,0' wird die Funktion auf inaktiv gesetzt.	81 04 27 32 02 01	Unsigned16	
8	Kind-Element im Baum Anzahl der Klingeltöne, bis Ruf angenommen wird. Wertebereich 0..10. Bei ,0' wird die Rufannahme abgeschaltet.	81 04 31 32 01 01	Unsigned8	

Tab. 16: Datenstruktur mit GSM-Parametern

Das Lesen sowie Schreiben der Elemente ist bis auf Position 2 von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

Das Schreiben der Position 2 ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.12 Datenstruktur zur Abfrage des IPT Status

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Abfrage des IPT Status	81 49 0D 06 00 FF	Nicht vorhanden	Einträge der Positionen 2-4
2	Kind-Element im Baum Ziel-IP-Adresse zum aktuellen IP-Telemetrie Master	81 49 17 07 00 00	Unsigned32 (IPv4) Octet String (IPv6) Aktuelle Ziel-IP	
3	Kind-Element im Baum Ziel-Port-Adresse zum aktuellen IP-Telemetrie Master	81 49 1A 07 00 00	Unsigned16 Aktueller Ziel-Port	
4	Kind-Element im Baum Quell-Port-Adresse zum aktuellen IP-Telemetrie Master.	81 49 19 07 00 00	Unsigned16 Aktueller Quell-Port	

Tab. 17: Datenstruktur zum IP-T-Status

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

Das Schreiben der Elemente ist nicht möglich.

7.3.1.13 Datenstruktur zur Lesen/Setzen der IPT Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Lesen/Setzen der IPT-Parameter	81 49 0D 07 00 FF	Nicht vorhanden	Einträge der Positionen 2 und 8
2	Kind-Element im Baum Wurzel primäre IPT-Parameter	81 49 0D 07 00 01	Nicht vorhanden	Einträge der Positionen 3-7
3	Kind-Element im Baum Ziel-Adresse zum primären IP-Telemetrie Master	81 49 17 07 00 01	Unsigned32 (IPv4) Octet String (IPv6 oder Hostname)	
4	Kind-Element im Baum Ziel-Port-Adresse zum primären IP-Telemetrie Master	81 49 1A 07 00 01	Unsigned16	
5	Kind-Element im Baum Quell-Port-Adresse zum primären IP-Telemetrie Master. Falls ‚0‘ eingetragen ist, kann ein Port frei gewählt werden.	81 49 19 07 00 01	Unsigned16	
6	Kind-Element im Baum Username zum primären IP-Telemetrie-Master	81 49 63 3C 01 01	Octet String	
7	Kind-Element im Baum Passwort zum primären IP-Telemetrie-Master	81 49 63 3C 02 01	Octet String	
8	Kind-Element im Baum Wurzel sekundäre IPT-Parameter	81 49 0D 07 00 02	Nicht vorhanden	Einträge der Positionen 9-15
9	Kind-Element im Baum Ziel-Adresse zum sekundären IP-Telemetrie Master	81 49 17 07 00 02	Unsigned32 (IPv4) Octet String (IPv6 oder Hostname)	
10	Kind-Element im Baum Ziel-Port-Adresse zum sekundären IP-Telemetrie Master	81 49 1A 07 00 02	Unsigned16	
11	Kind-Element im Baum Quell-Port-Adresse zum sekundären IP-Telemetrie Master. Falls ‚0‘ eingetragen ist, kann ein Port frei gewählt werden.	81 49 19 07 00 02	Unsigned16	
12	Kind-Element im Baum Username zum sekundären IP-Telemetrie-Master	81 49 63 3C 01 02	Octet String	
13	Kind-Element im Baum	81 49 63 3C 02 02	Octet String	

	Passwort zum sekundären IP-Telemetrie-Master		
14	Kind-Element im Baum Wartezeit, angegeben in Sekunden, nach der bei fehlerhafter TCP-Kommunikation ¹¹ die Wiederholung gestartet wird.	81 48 27 32 06 01	Unsigned8
15	Kind-Element im Baum Anzahl von Wiederholungen fehlerhafter TCP-Kommunikationsversuche. Bei ‚0‘ wird die Wiederholung fehlerhafter TCP-Kommunikationsversuche abgeschaltet.	81 48 31 32 02 01	Unsigned32

Tab. 18: Datenstruktur mit IP-T-Parametern

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.14 Datenstruktur zur Lesen/Setzen von AT-Hayes-Strings

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Lesen/Setzen der AT-Strings (falls vorhanden)	81 04 00 33 01 FF	Nicht vorhanden	Einträge der Positionen 2 und 4
2	Kind-Element im Baum AT-Hayes-String mit Template nach Modem-Power-ON vor PIN-Abfrage (falls vorhanden).	81 04 00 33 01 01	Octet String	
3	Kind-Element im Baum AT-Hayes-String mit Template nach PIN-Abfrage (falls vorhanden).	81 04 00 33 02 01	Octet String	
4	Kind-Element im Baum AT-Hayes-String mit Template zur GPRS-Initialisierung (falls vorhanden).	81 04 00 33 03 01	Octet String	

Tab. 19: Datenstruktur zu AT-Hayes Parametern

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

¹¹ Fehlerhafte TCP-Kommunikation bedeutet Verlust des TCP-Socket.

7.3.1.15 Datenstruktur zum Lesen / Setzen der Provider-abhängigen GPRS-Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Provider-abhängigen GPRS-Parameter	81 04 0D 07 00 FF	Nicht vorhanden	NN Einträge der Position 2 und deren Kind-Elemente.
2	Kind-Element im Baum Enthält den Provider Identifier NN zum APN Mit NN = 01...FE	81 04 0D 07 00 NN	Unsigned32 Aktuellen Provider Identifier	Einträge der Positionen 3 bis 9 dieser Tabelle, wenn erforderlich.
3	Kind-Element im Baum PPP-Username	81 04 61 3C 01 FF	Octet String Zugangsname	
4	Kind-Element im Baum PPP-Passwort	81 04 61 3C 02 FF	Octet String Zugangspass-wort	
5	Kind-Element im Baum Access Point Name	81 04 61 3C 03 FF	Octet String Access Point Name (APN)	
6	Kind-Element im Baum Primärer DNS-Server	81 48 17 07 04 FF	Unsigned32 (IPv4) Octet String (IPv6) Primärer DNS-Server	
7	Kind-Element im Baum Sekundärer DNS-Server	81 48 17 07 05 FF	Unsigned32 (IPv4) Octet String (IPv6) Sekundärer DNS-Server	
8	Kind-Element im Baum Tertiärer DNS-Server	81 48 17 07 06 FF	Unsigned32 (IPv4) Octet String (IPv6) Tertiärer DNS-Server	

Tab. 20: Datenstruktur mit GPRS-Provider-Parametern

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.16 Datenstruktur zum Lesen / Setzen der zulässigen GSM/GPRS-Betreiber

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Liste der zulässigen GSM/GPRS-Betreiber	81 04 0D 06 00 FF	Nicht vorhanden	Liste nach OBIS-T mit mindestens einem Eintrag entsprechend Pos. 2.
2	Kind-Element im Baum Zulässiger Provider NN Mit NN = 01 ... FE	81 04 0D 06 00 NN	Unsigned32 Zulässiger GSM/GPRS-Provider	

Tab. 21: Datenstruktur für GPRS-Betreiber-Parameter

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.17 Datenstruktur zur Abfrage dynamischer GPRS/GSM-Betriebsparameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der dynamisch gesetzten Provider-abhängigen GPRS-Parameter	81 04 0D 08 00 FF	Nicht vorhanden	Einträge der Positionen 2 bis 14, wenn vorhanden.
2	Kind-Element im Baum Enthält den aktuellen Provider Identifier zum APN	81 04 0D 07 00 00	Unsigned32 Aktueller Provider Identifier (APN)	
3	Kind-Element im Baum Enthält den aktuellen Provider Identifier des Netzes, bei dem das KM eingebucht ist	81 04 0D 06 00 00	Unsigned32 Aktueller Provider Identifier (Netz)	
4	Kind-Element im Baum Eigene IP-Adresse	81 48 17 07 00 00	Unsigned32 (IPv4) Octet String (IPv6) Eigene IP-Adresse	
5	Kind-Element im Baum Aktueller Primärer DNS-Server	81 48 17 07 04 00	Unsigned32 (IPv4) Octet String (IPv6) Aktueller Primärer DNS-Server	
6	Kind-Element im Baum Aktueller Sekundärer DNS-Server	81 48 17 07 05 00	Unsigned32 (IPv4) Octet String (IPv6) Aktueller Sekundärer DNS-Server	
7	Kind-Element im Baum Aktueller Tertiärer DNS-Server	81 48 17 07 06 00	Unsigned32 (IPv4) Octet String (IPv6) Aktueller Tertiärer DNS-Server	
8	Kind-Element im Baum Aktueller Zellenidentifizier	81 04 1A 07 00 00	Unsigned 16 Aktueller Zellenidentifizier	
9	Kind-Element im Baum Aktueller Location / Area Code	81 04 17 07 00 00	Unsigned 16 Aktueller Location / Area Code	
10	Kind-Element im Baum Aktuelle Feldstärke	81 04 2b 07 00 00	Integer 16 aktuelle Feldstärke anzugeben in dBm	
11	Kind-Element im Baum IMSI	81 04 00 00 04 01	Octet String IMSI	
12	Kind-Element im Baum IMEI	81 04 00 00 03 00	Octet String IMEI	
13	Kind-Element im Baum SIM Identifizier (ICC-ID)	81 04 00 00 05 01	Octet String SIM Identifizier (ICC-ID)	

14	Kind-Element im Baum Eigene Telefonnummer laut SIM-Karte	81 04 00 00 02 01	Octet String Eigene Telefonnummer laut SIM-Karte Falls die SIM-Karte diese Nummer nicht bereitstellt, ist ein leerer String zu liefern
----	---	-------------------	--

Tab. 22: Datenstruktur für dynamische GSM/GPRS-Betriebsparameter

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

Das Schreiben der Elemente ist nicht möglich.

7.3.1.18 Datenstruktur zum Lesen / Setzen der LAN/DSL-Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der LAN/DSL-Parameter	81 48 17 07 00 FF	Nicht vorhanden	Einträge der Positionen 2 bis 13 dieser Tabelle, wenn erforderlich.
2	Kind-Element im Baum „Rechnername“	81 48 00 00 00 00	Octet String Rechnername	Für den Rechnernamen müssen mindestens 32 Bytes angegeben werden können.
3	Kind-Element im Baum Primärer DNS-Server	81 48 17 07 04 01	Unsigned32 (IPv4) Octet String (IPv6) Primärer DNS-Server	
4	Kind-Element im Baum Sekundärer DNS-Server	81 48 17 07 05 01	Unsigned32 (IPv4) Octet String (IPv6) Sekundärer DNS-Server	
5	Kind-Element im Baum Tertiärer DNS-Server	81 48 17 07 06 01	Unsigned32 (IPv4) Octet String (IPv6) Tertiärer DNS-Server	
6	Kind-Element im Baum Eigene IP-Adresse	81 48 17 07 00 01	Unsigned32 (IPv4) Octet String (IPv6) Eigene IP-Adresse	
7	Kind-Element im Baum Eigene Subnetz-Maske	81 48 17 07 01 01	Unsigned32 (IPv4) Octet String (IPv6) Eigene Subnetz-Maske	
8	Kind-Element im Baum Default Gateway-IP-Adresse	81 48 17 07 02 01	Unsigned32 (IPv4) Octet String (IPv6) Gateway-IP	
9	Kind-Element im Baum	81 48 00 32 02 01	Boolean	

	DHCP Enabled		DHCP an/aus Bei ‚true‘ sind die IP-Parameter zum WAN per DHCP zu beziehen. Bei ‚false‘ sind die lokal gesetzten Werte zu verwenden
10	Kind-Element im Baum PPPoE Enabled	81 48 00 32 03 01	Boolean PPPoE an/aus Bei ‚true‘ ist für den WAN-Zugang PPPoE zu verwenden. Bei ‚false‘ ist der WAN-Zugang direkt per IP angebunden.
11	Kind-Element im Baum ICMP-Pakete beantworten	81 48 31 32 07 01	Boolean ICMP an/aus Bei ‚true‘ werden empfangene ICMP-Pakete beantwortet, per ‚false‘ werden diese Pakete verworfen
12	Kind-Element im Baum PPPoE-Username	81 04 62 3C 01 01	Octet String PPPoE-User Name
13	Kind-Element im Baum PPPoE-Passwort	81 04 62 3C 02 01	Octet String PPPoE-Passwort

Tab. 23: Datenstruktur mit LAN/DSL-Parametern

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.19 Datenstruktur zur Abfrage dynamischer LAN/DSL- Betriebsparameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der dynamisch gesetzten LAN/DSL-Parameter	81 48 0D 06 00 FF	Nicht vorhanden	Einträge der Positionen 2 bis 7, wenn vorhanden.
2	Kind-Element im Baum Primärer DNS-Server	81 48 17 07 04 00	Unsigned32 (IPv4) Octet String (IPv6) Primärer DNS-Server	
3	Kind-Element im Baum Sekundärer DNS-Server	81 48 17 07 05 00	Unsigned32 (IPv4) Octet String (IPv6) Sekundärer DNS-Server	
4	Kind-Element im Baum Tertiärer DNS-Server	81 48 17 07 06 00	Unsigned32 (IPv4) Octet String (IPv6) Tertiärer DNS-Server	

5	Kind-Element im Baum Eigene IP-Adresse	81 48 17 07 00 00	Unsigned32 (IPv4) Octet String (IPv6) Eigene IP-Adresse
6	Kind-Element im Baum Eigene Subnetz-Maske	81 48 17 07 01 00	Unsigned32 (IPv4) Octet String (IPv6) Eigene Subnetz- Maske
7	Kind-Element im Baum Gateway-IP-Adresse	81 48 17 07 02 00	Unsigned32 (IPv4) Octet String (IPv6) Gateway-IP

Tab. 24: Datenstruktur für dynamische LAN/DSL-Betriebsparameter

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig. Das Schreiben der Elemente ist nicht möglich.

7.3.1.20 Datenstruktur zum Lesen / Setzen der PLC-Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der PLC-Parameter	81 04 18 07 00 FF	Nicht vorhanden	Einträge der Pos. 2 bis 3 und 7 bis 20, wenn vorhanden
2	Kind-Element im Baum Utility ID	81 04 18 07 01 FF	Octet String Eindeutige Kennzeichnung des Netzbetreibers	
3	Kind-Element im Baum Netzwerk ID	81 04 18 07 02 01	Octet String Bezeichner zur Identifikation der ersten Netzwerk-ID des bevorzugten PLC-Netzwerks	
4	Kind-Element im Baum Encryption Key	81 04 18 07 11 FF	Octet String Wird kein Key angegeben, ist die Verschlüsselung abgeschaltet.	
5	Kind-Element im Baum PLC-Sendeleistung	81 04 18 07 12 FF	Unsigned8 Einstellung der Sendeleistung per Zuordnungsliste 0 ⇔ Default; 1 ⇔ Low; 9 ⇔ High	
6	Kind-Element im Baum Broadcast Duty Cycle	81 04 18 07 13 FF	Integer8 Prozentualer Wert Wertebereich: 1..100	

Tab. 25: Datenstruktur für PLC-Parameter

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.21 Datenstruktur zur Abfrage dynamischer PLC- Betriebsparameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der PLC-Parameter	81 04 0E 06 00 FF	Nicht vorhanden	Einträge der Positionen 2 bis 6 sowie 7 und 9, wenn vorhanden.
2	Kind-Element im Baum Device Serial Number (DSN)	81 04 0E 06 01 FF	Octet String Eindeutige Seriennummer des PLC-Netzknotens	
3	Kind-Element im Baum Device Node ID	81 04 0E 06 02 FF	Octet String Dem PLC-Netzknoten dynamisch zugewiesene Adresse	
4	Kind-Element im Baum Parent Node ID	81 04 0E 06 03 FF	Octet String Node Id des verbundenen Elternknoten oder Datenkonzentrators	
5	Kind-Element im Baum Current Network ID	81 04 0E 06 04 FF	Octet String Kennung des aktuellen PLC-Netzwerks	
6	Kind-Element im Baum Signalqualität	81 04 0E 06 05 FF	Integer16	
7	Kind-Element im Baum List of Child Node IDs	1. Liste: 81 04 0E 06 10 FF 2. Liste: 81 04 0E 06 11 FF ... usw. 32. Liste: 81 04 0E 06 2F FF	Octet String	Listenelemente gemäß Position 8 dieser Datenstruktur
8	Kind-Element im Baum Listenelemente Child Node ID	Listenelemente zu 1: 81 04 0E 06 10 NN Listenelemente zu 2: 81 04 0E 06 11 NN ... usw. Listenelemente zu 32: 81 04 0E 06 2F NN Wertebereich NN: 0x01 - 0xFE	Octet String Node ID eines direkt verbundenen PLC-Nodes	Listenelemente gemäß Position 9 bis 12 dieser Datenstruktur
9	Kind-Element im Baum Device Serial Number	81 04 0E 06 01 FF	Octet String Device Serial Number (DSN)	

10	Kind-Element im Baum Node Identifier	81 04 0E 06 02 FF	Octet String Node ID
11	Kind-Element im Baum Parent Node Identifier	81 04 0E 06 03 FF	Octet String Parent Node ID
12	Kind-Element im Baum MUC Server Identifier	81 81 C7 82 04 FF	Octet String Server ID

Tab. 26: Datenstruktur für dynamische PLC-Betriebsparameter

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.22 Datenstruktur zum Lesen / Setzen der W-MBUS-Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der W-MBUS-Parameter	81 06 19 07 00 FF	Nicht vorhanden	Einträge der Positionen 2 bis 7, wenn vorhanden.
2	Kind-Element im Baum Auswahl des an der Wireless M-Bus Schnittstelle zu verwendenden Funkprotokolls.	81 06 19 07 01 FF	Unsigned8 0 ⇔ T-Mode 1 ⇔ S-Mode 2 ⇔ S/T-Automatik (wechselnd) 3 ⇔ S/T-Automatik (Parallelbetrieb) Alle anderen Werte sind reserviert	
3	Kind-Element im Baum Zeitdauer, die fortlaufend im S-Mode empfangen wird (der Parameter wird nur benötigt, falls vorstehend die Variante „2 ⇔ S/T-Automatik“ gewählt ist).	81 06 19 07 02 FF	Unsigned8 Anzugeben in Sekunden 0 ist unzulässig	
4	Kind-Element im Baum Zeitdauer, die fortlaufend im T-Mode empfangen wird (der Parameter wird nur benötigt, falls vorstehend die Variante „2 ⇔ S/T-Automatik“ gewählt ist).	81 06 19 07 03 FF	Unsigned8 Anzugeben in Sekunden 0 ist unzulässig	
5	Kind-Element im Baum Automatischer Reboot: Periode, anzugeben in Sekunden, nach deren Ablauf das W-MBUS-Modem im MUC-C neu initialisiert werden soll. Bei ‚0‘ ist der automatische Reboot inaktiv.	81 06 27 32 03 01	Unsigned32	
6	Kind-Element im Baum Sendeleistung	81 06 19 07 04 FF	Unsigned8 Einstellung der Sendeleistung per Zuordnungsliste	

			0 ⇔ Default; 1 ⇔ Low; 2 ⇔ Medium; 3 ⇔ High
7	Kind-Element im Baum Maximales Inter Message Timeout in Sekunden	81 06 64 3C 01 01	Unsigned8
8	Kind-Element im Baum Maximales Timeout zwischen SML_Close-Request und SML_Open-Response in Sekunden für die Datenrichtung Request von der W-MBUS zum MUC	81 06 64 3C 01 04	• Unsigned16 Untere Grenze vom Wertebereich legt der Hersteller fest.
9	Kind-Element im Baum Maximales Timeout zwischen SML_Close-Request und SML_Open-Response in Sekunden für die Datenrichtung Request vom MUC an W-MBUS	81 06 64 3C 01 05	• Unsigned16

Tab. 27: Datenstruktur für Parameter zum Wireless M-Bus

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.23 Datenstruktur zum Lesen des W-MBUS-Status

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der W-MBUS- Parameter	81 06 0F 06 00 FF	Nicht vorhanden	Einträge der Positionen 2 bis 5, wenn vorhanden.
2	Kind-Element im Baum Hersteller-Typ-Bezeichnung, kodiert als ASCII-String, zum verwendeten W-MBUS-Adapter	81 06 00 00 01 00	• Octet String	
3	Kind-Element im Baum Eindeutige ID des W-MBUS- Adapters	81 06 00 00 03 00	• Octet String Kodierung gemäß EN 13757-3/4	
4	Kind-Element im Baum Firmwareversion des W-MBUS- Adapters	81 06 00 02 00 00	• Octet String	
5	Kind-Element im Baum Version der verwendeten W- MBUS-Hardware	81 06 00 02 03 FF	• Octet String	

Tab. 28: Datenstruktur zum Wireless M-Bus Status

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.24 Datenstruktur zum Lesen / Setzen der EDL-Zähler-Schnittstellen-Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der EDL-Zähler-Schnittstellen-Parameter	81 05 0D 07 00 FF	Nicht vorhanden	Einträge der Positionen 2 bis 5, wenn vorhanden.
2	Kind-Element im Baum Protokoll	81 05 0D 07 00 01	<ul style="list-style-type: none"> • Unsigned8 • Wertebereich: 1 ⇔ SML gemäß FNN-Lasten- heft EDL 	
3	Kind-Element im Baum Baudrate	81 05 0D 07 00 02	<ul style="list-style-type: none"> • Unsigned8 • Wertebereich: 0 ⇔ Auto-Baud 6 ⇔ 9600 baud 10 ⇔ 115200 baud 	

Tab. 29: Datenstruktur mit den EDL-Zähler-Schnittstellen-Parametern.

Das Lesen oder Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.25 Liste möglicher Push-Quellen

Pos.	Push-Quelle	Kodierung	Angaben zur Push-Quelle
1	Adressiertes Profil Diese Push-Quelle kann nur in Zusammenhang mit periodischen Push-Vorgängen verwendet werden.	81 81 C7 8A 42 FF	<p>Über das Feld ‚Listenelemente‘ wird die zu verwendende Push-Quelle angegeben (siehe Tab. 82).</p> <ul style="list-style-type: none"> • Als Bereich für das periodisch zu sendende Profil wird immer das Profil zum abgeschlossenen Push-Intervall ausgewählt.
2	Installationsparameter Diese Push-Quelle kann nur in Zusammenhang mit ereignisorientierten Push-Vorgängen verwendet werden.	81 81 C7 8A 43 FF	<p>Über das Feld ‚Listenelemente‘ werden die zu sendenden Installationsparameter¹² angegeben (siehe Tab. 83). Dabei muss mindestens ein Element dieser Tabelle angegeben sein.</p> <ul style="list-style-type: none"> • Das Ereignis wird eine Minute nach Erreichen der Betriebsbereitschaft und Verfügbarkeit der erforderlichen Kommunikationsverbindung ausgelöst.
3	Liste der sichtbaren Sensoren / Aktoren	81 81 C7 8A 44 FF	<p>Über das Feld ‚Listenelemente‘ werden die sichtbaren Sensoren / Aktoren übermittelt (siehe Tab. 49).</p> <ul style="list-style-type: none"> • Das Ereignis wird eine Minute nach Erreichen der Betriebsbereitschaft und Verfügbarkeit der erforderlichen Kommunikationsverbindung ausgelöst. • Das Ereignis wird zusätzlich periodisch ausgelöst. • Das Ereignis wird außerdem bei Änderung des Listeninhalts ausgelöst.

¹² Der Scanvorgang zur Erfassung der Daten kann nicht sicherstellen, dass immer alle am lokalen Bus angeschlossenen Module gemeldet werden.

4	Liste der aktiven Sensoren / Aktoren	81 81 C7 8A 45 FF	Über das Feld ‚Listenelemente‘ werden die aktiven Sensoren / Aktoren übermittelt (siehe Tab. 50).
			<ul style="list-style-type: none"> • Das Ereignis wird eine Minute nach Erreichen der Betriebsbereitschaft und Verfügbarkeit der erforderlichen Kommunikationsverbindung ausgelöst. • Das Ereignis wird zusätzlich periodisch ausgelöst. • Das Ereignis wird außerdem bei Änderung des Listeninhalts ausgelöst.

Tab. 30: Datenstruktur für mögliche Push-Quellen

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.26 Datenstruktur zum Transport der Eigenschaften von Push-Vorgängen.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Eigenschaften von Push-Vorgängen.	81 81 C7 8A 01 FF	Nicht vorhanden	Liste nach OBIS-T mit mindestens einem Eintrag entsprechend Pos. 2.
2	Kind-Element des Baums zum Transport der Eigenschaften von Push-Vorgängen. Liefert die Nummer des Push-Vorgangs. Enthält mindestens einen und optionale mehrere Verweise auf das / die zu transportierende(n) Attribute Mit NN = 01 ... 20	81 81 C7 8A 01 NN	Nicht vorhanden	Falls kein Listenelement angegeben ist, soll der Push Vorgang gelöscht werden/ ist er nicht vorhanden. Sobald mindestens ein Listenelement angegeben ist, ist der Push-Vorgang vorhanden bzw. wird ein neuer angelegt. Als Listenelemente sind die Einträge der Pos. 3 bis 7 zulässig
3	Kind-Element im Baum. Enthält die Information zum Push-Intervall. Die Angabe erfolgt in Sekunden. Wird / ist der Wert ‚0‘ angegeben, arbeitet der Push-Vorgang ereignisorientiert.	81 81 C7 8A 02 FF	Typ: Unsigned32 Inhalt: Push-Intervall	Nicht vorhanden
4	Kind-Element im Baum. Enthält die Information zur Push-Verzögerung. Die Angabe erfolgt in Sekunden.	81 81 C7 8A 03 FF	Typ: Unsigned32 Inhalt: Push-Verzögerung	Nicht vorhanden

5	Kind-Element im Baum. Enthält die Quelle zum Push-Vorgang.	81 81 C7 8A 04 FF	Typ: Octet String Inhalt: Je nach Push-Quelle, siehe Tab. 81	Je nach Quelle möglicherweise vorhanden. Die Quelle wird als Server-ID angegeben.
6	Kind-Element im Baum. Enthält das Ziel zum Push-Vorgang.	81 47 17 07 00 FF	Typ: Octet String Inhalt: Push-Target-Name nach IP-T	Je nach Ziel möglicherweise vorhanden.
7	Kind-Element im Baum. Enthält den Dienst zum Push-Vorgang.	81 49 00 00 10 FF	Typ: Octet String Inhalt: Siehe Tab. 32	Nicht vorhanden

Tab. 31: Datenstruktur für Transport-Parameter von Push-Vorgängen

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.1.27 Zeitverhalten periodischer Push-Vorgänge

Push-Vorgänge werden ausgelöst, wenn die Betriebsbereitschaft hergestellt worden und die zum Push erforderliche Kommunikationsverbindung verfügbar ist.

Die Push-Vorgänge zum adressierten Profil sind intervallgesteuert. Sobald der MUC-C mit der UTC-Zeit synchronisiert ist, werden die Push-Intervalle täglich auf 0:00 bezogen, wobei als Referenz UTC, UTC plus Zeitzone oder die gesetzliche / lokale Zeit herangezogen werden kann. Der Bezug wird mit dem Datenspiegel definiert.

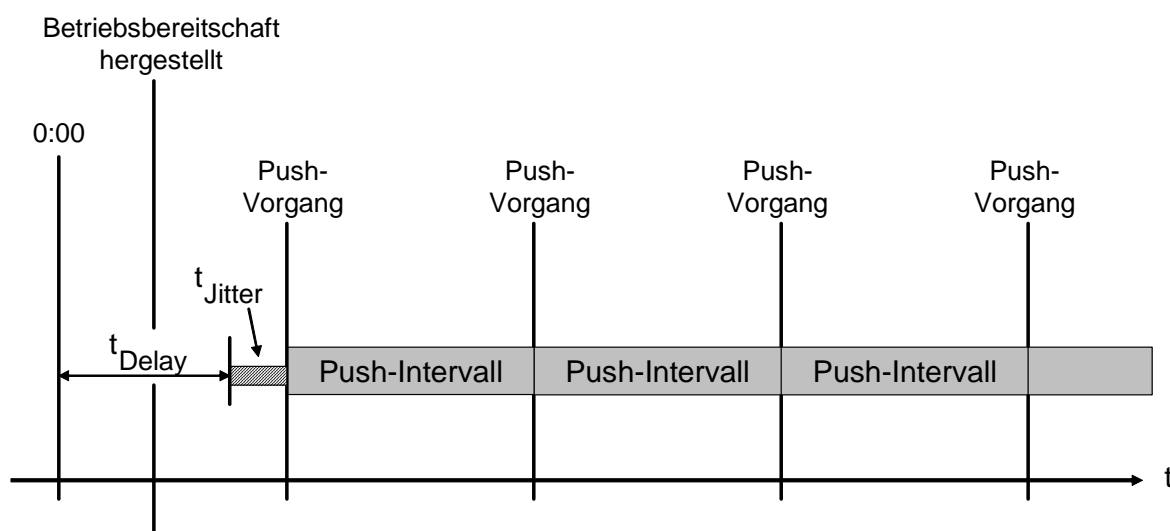


Bild 22: Zeitverhalten periodischer Push-Vorgänge

Der erste Push-Vorgang erfolgt nach Ablauf der Push-Verzögerung (t_{Delay}) und nach Ablauf einer Unschärfe (t_{Jitter}). Die Unschärfe t_{Jitter} wird automatisch nach folgender Regel eingestellt:

Formel:

$$t_{\text{Jitter}} = \text{ID} \times \frac{1,4}{8640} \times t_{\text{Interval}}$$

Beispiel:

$$t_{\text{Jitter}} = 255 \times \frac{1,4}{8640} \times 86400 \text{ s} = 3570 \text{ s} \Rightarrow \text{ca. 1h}$$

Server ID = 05 00 60 4C C7 1C (FF)

Bild 23: Berechnung zum Jitter bei periodischen Pushvorgängen

ID entspricht den letzten beiden Stellen der Server ID.

Ist ein Push-Intervall kleiner 1 Stunde, wird t_{Jitter} auf 0 gesetzt.

Mit der Push-Verzögerung wird beispielsweise eingestellt, dass der erste Push-Vorgang immer um 10:00 Uhr morgens zzgl. einer Unschärfe von t_{Jitter} erfolgen soll.

Wird das Push-Intervall auf 0 gesetzt, erfolgt der Pushvorgang sofort bei Eintreffen neuer Werte an der ausgewählten Push-Quelle.

7.3.2 Liste möglicher Push-Dienste

Pos.	Push-Dienst	Kodierung	Datentyp	Angaben zur Push-Dienst
1	SML als Response without Request über IP-Telemetrie	81 81 C7 8A 21 FF	Octet String	<ul style="list-style-type: none"> Als Push-Ziel wird das Push-Target, siehe E DIN 43863-4, verwendet. Zusätzliche Angaben werden nicht benötigt.
2	SML als Response without Request	81 81 C7 8A 22 FF	Octet String	<ul style="list-style-type: none"> Als Push-Ziel wird die SML-Client-Adresse verwendet. Zusätzliche Angaben werden nicht benötigt.
3	Herstellerspezifisches Protokoll über KNX (Der Push-Dienst steht nur an der Kundenschnittstelle zu Verfügung.)	81 81 C7 8A 23 FF	Octet String	<ul style="list-style-type: none"> Als Push-Ziel wird die KNX ID verwendet. Zusätzliche Angaben werden nicht benötigt.

Tab. 32: Datenstruktur für Push-Dienste

7.3.2.1 Datenstruktur zur Adressierung einer bestimmten Push-Quelle

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Kind-Element im Baum zum Transport der Adresse einer Push-Quelle. Enthält die Server-ID zur Push-Quelle.	81 81 C7 8A 81 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Server-ID der Push-Quelle 	Mindestens ein oder mehrere Einträge nach Position 2 dieser Tabelle.
2	Kind-Element im Baum zum Transport der Kennzahl (des „Kanals“) einer Push-Quelle. Enthält die Kennzahl der von der Push-Quelle zu liefernden (Mess-) Größe.	81 81 C7 8A 82 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Kennzahl der von der Push-Quelle zu liefernden (Mess-) Größe. Diese Kennzahl wird gemäß OBIS angegeben (Kodierung als Kette von 6 Bytes). 	Nicht vorhanden
3	Kind-Element im Baum zum Transport der Adresse einer Push-Quelle. Enthält die OBIS-Kennzahl zur Push-Quelle.	81 81 C7 8A 83 FF	<ul style="list-style-type: none"> • Octet String 	

Tab. 33: Datenstruktur zur Adressierung einer Push-Quelle

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.2 Datenstruktur zum Transport der Installationsparameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Kind-Element im Baum zum Transport des Installationsparameters: IP-Adresse zum WAN	81 81 C7 82 81 FF	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: IP-Adresse 	Nicht vorhanden
2	Kind-Element im Baum zum Transport des Installationsparameters: Geräteklasse	81 81 C7 82 02 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Geräteklasse 	Nicht vorhanden

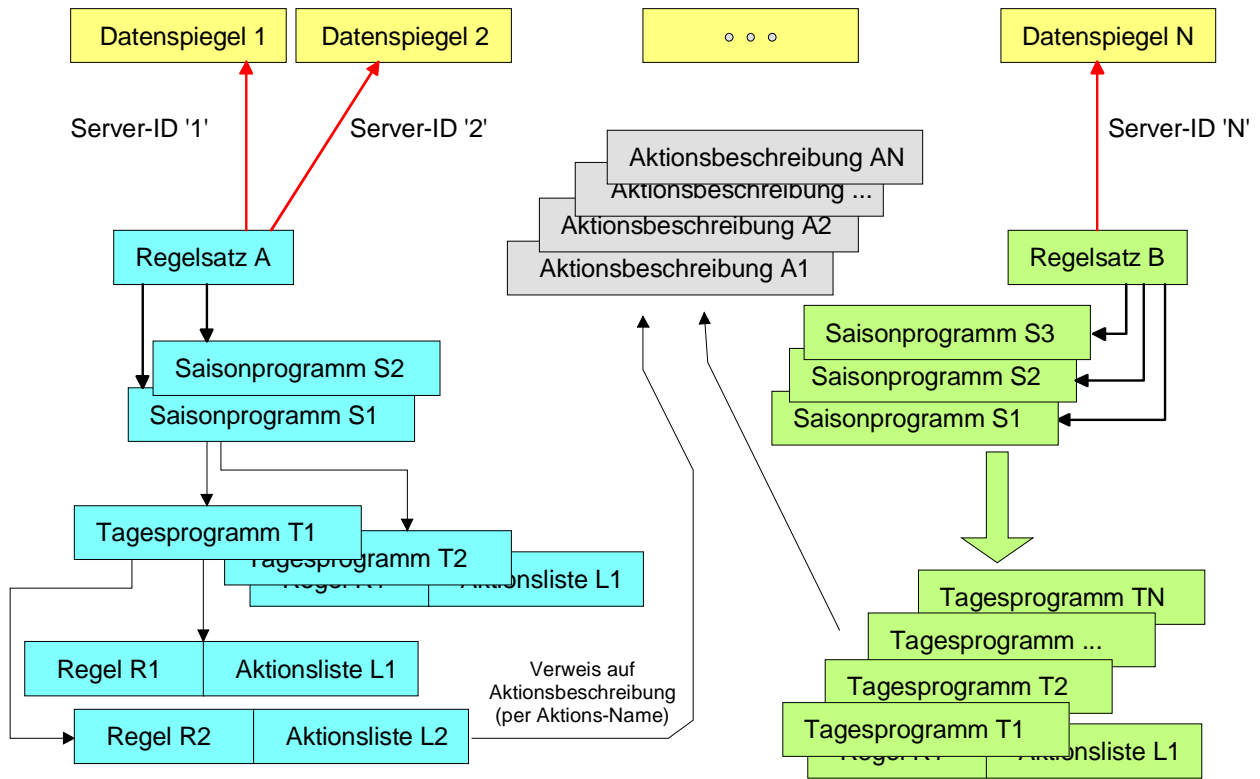
Tab. 34: Datenstruktur zum Transport der Installationsparameter

Je nach Betriebsumgebung können einzelne der oben aufgelisteten Parameter in der Liste der Detailinformationen eines Push-Vorgangs zu den Installationsparametern verwendet werden. Die gleichzeitige Verwendung aller Detailinformationen wird nicht auftreten.

7.3.2.3 Datenstruktur zum Transport der Eigenschaften von Regelsätzen

Regelsätze zur Beschreibung von Aktionsschaltprogrammen basieren auf folgender Struktur:

Je gezeichneter "gelben Kiste (<=> Zähler / Sensor / Aktor) gilt eine eigene Server-ID. Dieser kann jeweils eine eigene Benutzer/Password-Kombination zugeordnet sein. (Datenspiegel können auch reine Aktoren / Sensoren ohne Spiegel-Funktion sein)



Benutzerrechte: Können (und sollten) jeweils für einen farblich gleichen dargestellten Bereich auch identisch gewählt werden.

Konsistenzprüfung: Erfolgt auf Basis eines Regelsatzes. Änderungen an einem Regelsatz können per Delta-Übertragung erfolgen, wirken wegen der Konsistenzforderung aber immer nur auf einen Regelsatz. Die Konsistenzprüfung soll per Hash erfolgen.

Bild 24: Struktur von Aktionsschaltprogrammen

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Auflistung aller Regelsätze	81 81 C7 A0 01 FF	Nicht vorhanden	Liste mit keinem, einem oder mehreren Einträgen nach Pos. 2
2	Kind-Element mit Regelsatz-Identifikation	81 81 C7 A0 01 NN	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Name/ Freitext zum Regelsatz 	Nicht vorhanden

Tab. 35: Datenstruktur zur Auflistung aller Regelsätze

Diese Tabelle ist nur lesbar; sie wird mit Anlegen / Entfernen eines Regelsatz jeweils aktualisiert, Das Lesen ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Eigenschaften von Regelsätzen. Mit NN = 01 ... 0F	81 81 C7 A0 01 NN	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Name/ Freitext zum Regelsatz 	Liste mit genau einem Eintrag entsprechend Pos. 2 und Pos. 4
2	Kind-Element mit Liste der Server-ID, auf die der Regelsatz einwirkt.	81 81 C7 A0 02 FF	Nicht vorhanden	Mindestens ein Element nach Pos. 3
3	Kind-Element mit Server-ID Mit NN = 01 ... 20	81 81 C7 A0 02 NN	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Server-IDI 	Nicht vorhanden
4	Kind-Element mit Liste der Saisonprogramme.	81 81 C7 A0 03 FF	Nicht vorhanden	Mindestens ein Element nach Pos. 5
5	Kind-Element mit Saisonprogramm Mit NN = 01 ... 55 Jede Kennzahl korrespondiert mit einer Instanz gemäß Tab. 37	81 81 C7 A0 03 NN	Nicht vorhanden	Nicht vorhanden

Tab. 36: Datenstruktur für Parameter von Regelsätzen

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.4 Datenstruktur zum Transport der Eigenschaften von Saisonprogrammen

Instanzen dieser Tabelle sind nur in Kombination mit dem jeweils übergeordneten Regelsatz zu adressieren. Damit besteht der Tree-Path immer aus zwei Elementen:

- 81 81 C7 A0 01 NN

- 81 81 C7 A0 03 NN

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Eigenschaften von Saisonprogrammen. Mit NN = 01 ... 55	81 81 C7 A0 03 NN	Nicht vorhanden	Liste mit genau einem Eintrag entsprechend Pos. 2 bis Pos. 4 sowie Pos. 6
2	Kind-Element mit Startzeitpunkt	81 81 C7 A0 04 FF	Typ: SML_Time Inhalt: Startzeitpunkt	Nicht vorhanden
3	Kind-Element mit Merkmal, ist Einzelaktion'	81 81 C7 A0 05 FF	Typ: Boolean Inhalt: TRUE ⇔ Einzelaktion (mit erstem Abarbeiten einer	Nicht vorhanden

				Aktionsliste wird das Saisonpro- gramm deaktiviert) FALSE ⇔ Zyklische Nutzung, mit Start des Saisonprogramms wird ein anderes, aktivies zyklisch arbeitendes Saisonprogramm verdrängt	
4	Kind-Element mit Liste der Tagesprogramme	81 81 C7 A0 06 FF	Nicht vorhanden		Mindestens ein Element nach Pos. 5
5	Kind-Element mit Tagesprogramm Mit NN = 01 ... 55 Jede Kennzahl korrespondiert mit einer Instanz gemäß Tab. 38	81 81 C7 A0 06 NN	Nicht vorhanden		Nicht vorhanden
6	Kind-Element mit Liste der Tarife und Preise	81 81 C7 A0 07 FF	Nicht vorhanden		Mindestens ein Element nach Pos. 7
7	Kind-Element mit Tarif-Preis- Kombination Mit NN = 01 ... FE	81 81 C7 A0 07 NN	Nicht vorhanden		Jeweils genau ein Element nach Pos. 8 und Pos 9
8	Kind-Element mit Tarifnummer	81 81 C7 A0 08 FF	Typ: Octet String Inhalt: OBIS-Kennzahl des Registers. für das der Preis gilt		Nicht vorhanden
9	Kind-Element mit Preis	81 81 C7 A0 09 FF	Typ: SML_PeriodEntry Inhalt: Preis gemäß Vertrag pro Zähl- einheit (kWh, m ³ , ...)		Nicht vorhanden

Tab. 37: Datenstruktur für Parameter von Saisonprogrammen

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

Fallen mehrere Tagesprogrammen auf dasselbe Datum, so ist das Tagesprogramm mit der höchsten Priorität zu benutzen. Gibt es in diesem Fall keine höchste Priorität, ist das letzte Tagesprogramm zu benutzen.

7.3.2.5 Datenstruktur zum Transport der Eigenschaften von Tagesprogrammen

Instanzen dieser Tabelle sind nur in Kombination mit dem jeweils übergeordneten Regelsatz und zugehörigen Saisonprogramm zu adressieren. Damit besteht der Tree-Path immer aus drei Elementen:

- 81 81 C7 A0 01 NN
- 81 81 C7 A0 03 NN
- 81 81 C7 A0 06 NN

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Eigenschaften von Tagesprogrammen. Mit NN = 01 ... 55	81 81 C7 A0 06 NN	Nicht vorhanden	Liste mit genau einem Eintrag entsprechend Pos. 2 und Pos. 4
2	Kind-Element mit Liste der Regeln und Aktionslisten	81 81 C7 A0 0A FF	Nicht vorhanden	Mindestens ein Element nach Pos. 3
3	Kind-Element mit Regel und Aktionsliste	81 81 C7 A0 12 NN	Nicht vorhanden	Liste gemäß Tab. 39
4	Kind-Element mit Definition des betroffenen Tages	81 81 C7 A0 0B FF	Nicht vorhanden	Liste mit genau zwei Elementen nach Pos. 5 und einem Eintrag aus 6 bis 8
5	Kind-Element mit Priorität des betroffenen Tages	81 81 C7 A0 0C FF	Typ: Unsigned8 Inhalt: Priorität 0x00 ⇔ Niedrigste 0x01 ⇔ ... 0x02 ⇔ ... 0x07 ⇔ Höchste	Nicht vorhanden
6	Kind-Element mit Wochentag	81 81 C7 A0 0d FF	Typ: Unsigned8 Inhalt: Bitkodierter Wochentag, 0x01 ⇔ Montag,, 0x02 ⇔ Dienstag, 0x04 ⇔Mittwoch, ... 0x40 ⇔ Sonntag ... 0x7F ⇔ Alle Tage 0x8x ⇔ reserviert	Nicht vorhanden
7	Kind-Element mit festem Datum	81 81 C7 A0 0E FF	Typ: SML_Time Inhalt: Datumsangabe (Uhrzeitinformation wird ignoriert)	Nicht vorhanden
8	Kind-Element mit Liste für Offset und Referenzzeitpunkt	81 81 C7 A0 0F FF	Nicht vorhanden	Je ein Element gemäß Pos. 9 und Pos. 10
9	Kind-Element mit Offset	81 81 C7 A0 10 FF	Typ: Integer16 Inhalt: Offset in Tagen	Nicht vorhanden

				bezogen auf den Referenzzeitpunkt. Nur in der Variante 'Erster Werktag im Monat' ist der Offset als Anzahl von Werktagen zu werten. In allen anderen Fällen wird er als Offset von Tagen betrachtet.	
10	Kind-Element mit Definition des Referenzzeitpunkts	81 81 C7 A0 11 FF	Nicht vorhanden	Liste mit genau einem Eintrag nach Pos. 11 und optional einem weiteren nach Pos. 12	
11	Kind-Element mit Definition der Bedeutung zum Referenzzeitpunkt	81 81 C7 A0 11 01	Typ: Unsigned8 Inhalt: Bedeutung für Pos. 12 0x01 ⇔ Monatsanfang 0x02 ⇔ Monatsende 0x03 ⇔ Erster Sonntag im Monat 0x04 ⇔ Letzter Sonntag im Monat 0x05 ⇔ Ostersonntag 0x06 ⇔ Erster Januar 0x07 ⇔ Erster Werktag im Monat 0x08 ⇔ Letzter Werktag im Monat	Nicht vorhanden	
12	Kind-Element mit Referenzzeitpunkt	81 81 C7 A0 11 02	Typ: Unsigned16 Inhalt: Bitkodierter Monat 0x0001 ⇔ Januar, 0x0002 ⇔ Februar, 0x0004 ⇔ März, ... 0x0800 ⇔ Dez. 0x0FFF ⇔ jeder Monat alle anderen sind reserviert.	Nicht vorhanden	

Tab. 38: Datenstruktur für Parameter von Tagesprogrammen

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

Werktage sind Montag, Dienstag, Mittwoch, Donnerstag und Freitag,

7.3.2.6 Datenstruktur zum Transport der Eigenschaften von Regeln mit Aktionslisten

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Eigenschaften von Regeln und Aktionslisten Mit NN = 01 ... FE	81 81 C7 A0 12 NN	Nicht vorhanden	Liste mit genau einem Eintrag entsprechend Pos. 2 und Pos. 3
2	Kind-Element mit Regel	81 81 C7 A0 13 FF	Typ: Octet String Inhalt: OBIS-T Kennzahl der Regel	Nicht vorhanden
3	Kind-Element mit Liste der Aktionen	81 81 C7 A0 14 FF	Nicht vorhanden	Mindestens ein Element nach Pos. 4
4	Kind-Element mit Aktion Mit NN = 01 ... 10	81 81 C7 A0 14 NN	Typ: Octet String Inhalt: OBIS-T Kennzahl der Aktion	Nicht vorhanden

Tab. 39: Datenstruktur für Parameter mit Regeln und Aktionslisten

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.7 Datenstruktur zum Transport der Liste aller Aktionsbeschreibungen

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Eigenschaften Aktionen	81 81 C7 A0 15 FF	Nicht vorhanden	Liste mit genau einem Eintrag nach Pos. 2
2	Kind-Element mit den Eigenschaften einer Aktionen Mit NN = 01 ... FE	81 81 C7 A0 15 NN	Nicht vorhanden	Liste mit genau einem Eintrag der Pos. 3 bis Pos. 7
3	Kind-Element mit Aktion ,Register On Off'	81 81 C7 A0 16 FF	Nicht vorhanden	Liste gemäß Tab. 41
4	Kind-Element mit Aktion ,Register Reset'	81 81 C7 A0 17 FF	Typ: Octet String Inhalt: OBIS Kennzahl des Registers	Nicht vorhanden
5	Kind-Element mit Aktion ,Zustand ausgeben'	81 81 C7 A0 18 FF	Nicht vorhanden	Liste gemäß Tab. 45
6	Kind-Element mit Aktion ,Impuls ausgeben'	81 81 C7 A0 19 FF	Nicht vorhanden	Liste gemäß Tab. 46
7	Kind-Element mit Aktion ,SML-Datei senden'	81 81 C7 A0 1A FF	Nicht vorhanden	Liste gemäß Tab. 47

Tab. 40: Datenstruktur für Liste aller Aktionsbeschreibungen mit deren Namen (Kennzahlen)

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.8 Datenstruktur zum Transport der Eigenschaften von ‚Register On Off‘

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Eigenschaften zur Aktion ‚Register On Off‘	81 81 C7 A0 16 FF	Nicht vorhanden	Liste mit genau einem Eintrag der Pos. 2 bis Pos. 4
2	Kind-Element mit Aktionsparametern ‚Register intern‘	81 81 C7 A0 1b FF	Nicht vorhanden	Liste gemäß Tab. 42
3	Kind-Element mit Aktionsparametern ‚Register extern‘	81 81 C7 A0 1c FF	Nicht vorhanden	Liste gemäß Tab. 43
4	Kind-Element mit Aktionsparametern ‚Register EDL40‘	81 81 C7 A0 1d FF	Nicht vorhanden	Liste gemäß Tab. 44

Tab. 41: Datenstruktur für Parameter zur Aktion ‚Register on Off‘

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.9 Datenstruktur zum Transport der Eigenschaften von ‚Register intern‘

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Aktionsparameter ‚Register intern‘	81 81 C7 A0 1b FF	Nicht vorhanden	Liste mit genau einem Eintrag der Pos. 2 bis Pos. 3
2	Kind-Element mit Name des zugehörigen Registers	81 81 C7 A0 1e FF	Typ: Octet String Inhalt: OBIS Kennzahl des Registers	Nicht vorhanden
3	Kind-Element mit neuem Zustand	81 81 C7 A0 1f FF	Typ: Boolean Inhalt: TRUE ⇔ Aktiviert das Register	Nicht vorhanden

Tab. 42: Datenstruktur für Parameter zur Aktion ‚Register intern‘

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.10 Datenstruktur zum Transport der Eigenschaften von ‚Register extern‘

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der	81 81 C7 A0 1c FF	Nicht vorhanden	Liste mit genau einem Eintrag der

Aktionsparameter ‚Register extern‘				Pos. 2 bis Pos. 5
2	Kind-Element mit Name des zugehörigen Registers	81 81 C7 A0 20 FF	Typ: Octet String Inhalt: OBIS Kennzahl des Registers	Nicht vorhanden
3	Kind-Element mit Retriigger-Period	81 81 C7 A0 21 FF	Typ: Unsigned16 Inhalt: Periode in ‚ms‘ zur erneuten Ausgabe des Tarifsignals an einen EDL21-Zähler	Nicht vorhanden
4	Kind-Element mit Server-ID des externen Zählers	81 81 C7 A0 22 FF	Typ: Octet String Inhalt: Server-ID des EDL21-Zählers	Nicht vorhanden
5	Kind-Element mit Tariffinormation	81 81 C7 A0 23 FF	Typ: Unsigned8 Inhalt: Tariffinormation gemäß EDL21-Spezifikation	Nicht vorhanden

Tab. 43: Datenstruktur für Parameter zur Aktion ‚Register extern‘

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.11 Datenstruktur zur Aufzeichnung in einem Datenspiegel

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Aktionsparameter ‚Register EDL40‘	81 81 C7 A0 1d FF	Nicht vorhanden	Liste mit genau einem Eintrag nach Pos. 2
2	Kind-Element mit Name des zugehörigen Datenspiegels (siehe Tab. 75, Pos. 9)	81 81 C7 A0 24 FF	Typ: Octet String Inhalt: OBIS Kennzahl zum Datenspiegel	Nicht vorhanden

Tab. 44: Datenstruktur für Parameter zur Aufzeichnung in einem Datenspiegel

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

Das Ausführen dieser Aktion bewirkt die Aufzeichnung in den in der Datenstruktur angegebenen Datenspiegel (siehe Tab. 75).

Sollten mehrere Datenspiegel per Aktionsschaltprogramm angesteuert werden, ist je Datenspiegel eine Aktion zur Aufzeichnung anzulegen.

7.3.2.12 Datenstruktur zum Transport der Eigenschaften von ‚Zustand ausgeben‘

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Aktionsparameter ‚Zustand ausgeben‘	81 81 C7 A0 18 FF	Nicht vorhanden	Liste mit genau einem Eintrag der Pos. 2 bis Pos. 4
2	Kind-Element mit Name des zugehörigen Aktors	81 81 C7 A0 27 FF	Typ: Octet String Inhalt: OBIS Kennzahl des Aktors	Nicht vorhanden
3	Kind-Element mit Server-ID des zugehörigen Aktors	81 81 C7 A0 28 FF	Typ: Octet String Inhalt: Server-ID des Aktors	Nicht vorhanden
4	Kind-Element mit neuem Zustand	81 81 C7 A0 29 FF	Typ: Boolean Inhalt: Bedeutung legt Aktor fest	Nicht vorhanden

Tab. 45: Datenstruktur für Parameter zur Aktion ‚Zustand ausgeben‘

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.13 Datenstruktur zum Transport der Eigenschaften von ‚Impuls ausgeben‘

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Aktionsparameter ‚Impuls ausgeben‘	81 81 C7 A0 19 FF	Nicht vorhanden	Liste mit genau einem Eintrag der Pos. 2 bis Pos. 5
2	Kind-Element mit Name des zugehörigen Aktors	81 81 C7 A0 2A FF	Typ: Octet String Inhalt: OBIS Kennzahl des Aktors	Nicht vorhanden
3	Kind-Element mit Server-ID des zugehörigen Aktors	81 81 C7 A0 2b FF	Typ: Octet String Inhalt: Server-ID d. Aktors	Nicht vorhanden
4	Kind-Element mit neuem Zustand	81 81 C7 A0 2C FF	Typ: Boolean Inhalt: Bedeutung legt Aktor fest	Nicht vorhanden
5	Kind-Element mit Impulsdauer	81 81 C7 A0 2d FF	Typ: Unsigned16 Inhalt: Impulsdauer in ‚ms‘	Nicht vorhanden

Tab. 46: Datenstruktur für Parameter zur Aktion ‚Impuls ausgeben‘

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.14 Datenstruktur zum Transport der Eigenschaften von ‚SML-Datei ausgeben‘

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Aktionsparameter ‚SML-Datei ausgeben‘	81 81 C7 A0 1A FF	Nicht vorhanden	Liste mit genau einem Eintrag der Pos. 2 bis Pos. 5
2	Kind-Element mit Ziel-Server-ID	81 81 C7 A0 2E FF	Typ: Octet String Inhalt: Server-ID des Empfängers	Nicht vorhanden
3	Kind-Element mit Username beim Empfänger	81 81 C7 A0 2F FF	Typ: Octet String Inhalt: Username	Nicht vorhanden
4	Kind-Element mit Password beim Empfänger	81 81 C7 A0 30 FF	Typ: Octet String Inhalt: Password	Nicht vorhanden
5	Kind-Element mit Liste der SML-Messages	81 81 C7 A0 31 FF	Nicht vorhanden	Liste mit mindestens einem Element nach Pos. 6
6	Kind-Element mit einer SML-Message NN von 01 ... 03	81 81 C7 A0 31 NN	Nicht vorhanden	Liste gemäß Pos. 7 ff.
7	Kind-Element mit Message-ID der Nachricht	81 81 C7 A0 32 FF	Typ: Unsigned32 Inhalt: Message-ID	Nicht vorhanden
8	Kind-Element mit Obj-Name der Nachricht	81 81 C7 A0 33 FF	Typ: Octet String Inhalt: Ziel-Obj-Name	Nicht vorhanden
9	Kind-Element mit Payload der Nachricht (nur bei Message-ID ⇔ Set-Proc-Parameter benötigt)	81 81 C7 A0 34 FF	Typ: SML_ProcParValue Inhalt: -	Nicht vorhanden

Tab. 47: Datenstruktur für Parameter zur Aktion ‚SML-Datei ausgeben‘

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.15 Datenstruktur zum Transport der Liste aller ‚Regeln‘

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Liste aller Regeln	81 81 C7 A0 35 FF	Nicht vorhanden	Leere Liste oder Liste mit mindestens einem Eintrag nach Pos. 2
2	Kind-Element mit Eigenschaften der Regel	81 81 C7 A0 35 NN	Typ: Octet String Inhalt: OBIS-T Kennzahl der Regel	Liste mit einem Element nach folgenden Positionen: 3, 4 und 20, 5, 10, 14, 15 oder 16.
3	Kind-Element mit Totzeit zum Entprellen	81 81 C7 A0 36 FF	Typ: Unsigned16 Inhalt: Zeit in Sekunden, innerhalb derer nach dem Auslösen der Regel eine erneute Zustandsänderung der Regel verboten ist	Nicht vorhanden
4	Kind-Element mit Regel für Zeitbezug zum Auslösen oder Aktivieren des statischen Ergebnisses (Ausgangszustand) der Regel	81 81 C7 A0 37 FF	Typ: SML_Time Inhalt: Zeitpunkt zur Auslösung der Regel (Auswertung ohne Datum) Der Zeitpunkt gilt als Absolutwert bezogen auf 00:00, wenn keine der nachfolgenden Positionen angegeben ist. Ist eine der nachfolgenden Positionen angegeben und wird ein Zeitbezug definiert, ist dieser als Offset gerechnet ab dem Auslösen des Ereignisses zu werten. Die Regel löst dann um diesen Zeitoffset versetzt aus.	Nicht vorhanden
5	Kind-Element mit Regel für Schwellwert	81 81 C7 A0 38 FF	Nicht vorhanden	Liste nach Pos. 6 bis 10
6	Kind-Element mit Schwellwert-Eigenschaft ‚Server-ID‘	81 81 C7 A0 38 01	Typ: Octet String Inhalt: Server-ID der Datenquelle	Nicht vorhanden

7	Kind-Element mit Schwellwert-Eigenschaft ‚Obj-Name‘	81 81 C7 A0 38 02	Typ: Octet String Inhalt: OBIS Kennzahl der Datenquelle	Nicht vorhanden
8	Kind-Element mit Schwellwert-Eigenschaft ‚obere Schwelle‘	81 81 C7 A0 38 03	Typ: SML_PeriodEntry Inhalt: oberer Schwellwert	Nicht vorhanden
9	Kind-Element mit Schwellwert-Eigenschaft ‚untere Schwelle‘ (Falls keine Hysterese gewünscht ist, muss die ‚untere Schwelle‘ gleich der ‚oberen Schwelle‘ gesetzt werden.)	81 81 C7 A0 38 04	Typ: SML_PeriodEntry Inhalt: unterer Schwellwert	Nicht vorhanden
10	Kind-Element mit Schwellwert-Eigenschaft ‚Bedingung‘	81 81 C7 A0 38 05	Typ: Unsigned8 Inhalt: 0x00 ⇔ ‚== oS‘ 0x01 ⇔ ‚> oS‘ 0x02 ⇔ ‚< uS‘ 0x03 ⇔ ‚>= oS‘ 0x04 ⇔ ‚<= uS‘ oS ⇔ obere Schwelle uS ⇔ untere Schwelle‘	Nicht vorhanden
11	Kind-Element mit Regel für Signal-Eingang (Kann auch Ausgangszustand einer anderen Regel sein)	81 81 C7 A0 39 FF	Nicht vorhanden	Liste nach Pos. 12 bis ..15
12	Kind-Element mit Signal-Eigenschaft ‚Server-ID‘	81 81 C7 A0 39 01	Typ: Octet String Inhalt: Server-ID der Datenquelle	Nicht vorhanden
13	Kind-Element mit Signal-Eigenschaft ‚Obj-Name‘	81 81 C7 A0 39 02	Typ: Octet String Inhalt: OBIS Kennzahl der Datenquelle	Nicht vorhanden
14	Kind-Element mit Signal-Eigenschaft ‚Flanken- oder Zustandsart‘	81 81 C7 A0 39 03	Typ: Unsigned8 Inhalt: 1 ⇔ ‚steigende Flanke‘ 2 ⇔ ‚fallende Flanke‘ 3 ⇔ Eingangszustd. aktiv 4 ⇔ Eingangszustd. Inaktiv Alle anderen Werte	Nicht vorhanden

			sind unzulässig'	
15	Kind-Element mit Polling-Zyklus zur Abfrage des Signal-Eingangs	81 81 C7 A0 39 04	Typ: Unsigned16 Inhalt: Periode zur Abfrage des Signal-Zustands in Sekunde. Mit ,0' wird die maximal vom Gerät mögliche Zykluszeit gewählt.	Nicht vorhanden
16	Kind-Element mit Regel für Saisonstart oder Saisonende	81 81 C7 A0 3A FF	Typ: Boolean Inhalt: TRUE ⇔ Auslösung der Regel erfolgt bei Saisonstart FALSE ⇔ Auslösung der Regel erfolgt bei Saisonende	Nicht vorhanden
17	Kind-Element mit Regel für statisches Ergebnis ,TRUE' oder ,FALSE'	81 81 C7 A0 3b FF	Typ: Boolean Inhalt: Statisches Ergebnis der Regel	Nicht vorhanden
18	Kind-Element mit Regel für Operation	81 81 C7 A0 3C FF	Nicht vorhanden	Liste nach Pos. 19 bis ..21
19	Kind-Element mit linkem Operand	81 81 C7 A0 3C 01	Typ: Octet String Inhalt: Obj-Name der Regel	Nicht vorhanden
20	Kind-Element mit Operator	81 81 C7 A0 3C 02	Typ: Unsigned8 Inhalt: 0x00 ⇔ 'UND' 0x01 ⇔ 'ODER' 0x02 ⇔ 'UND NICHT' 0x03 ⇔ 'ODER NICHT'	Nicht vorhanden
21	Kind-Element mit rechtem Operand	81 81 C7 A0 3C 03	Typ: Octet String Inhalt: Obj-Name der Regel	Nicht vorhanden
22	Kind-Element mit Regel für Zeitbezug zum Auslösen oder De-Aktivieren des statischen Ergebnisses (Ausgangszustand) der Regel	81 81 C7 A0 3d FF	Typ: SML_Time Inhalt: Zeitpunkt zum Rücksetzen des statischen Zustands (Auswertung ohne Datum) Zur Bedeutung siehe oben.	Nicht vorhanden

23	Kind-Element zum Auslösen der Regel bei Betriebsbereitschaft	81 81 C7 A0 3E FF	Typ: Boolean Inhalt: TRUE ⇔ Regel wird mit Erreichen der Betriebsbereitschaft ausgelöst	Nicht vorhanden
24	Kind-Element zum Auslösen der Regel bei Zustandswechsel der Systemzeit	81 81 C7 A0 3F FF	Typ: Boolean Inhalt: TRUE ⇔ Regel wird mit Wechsel in den Zustand ‚Systemzeit gültig‘ ausgelöst FALSE ⇔ Regel wird mit Wechsel in den Zustand ‚Systemzeit ungültig‘ ausgelöst	Nicht vorhanden

Tab. 48: Datenstruktur für Liste aller ‚Regeln‘

Das Lesen sowie Schreiben der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.16 Datenstruktur mit Liste der sichtbaren Sensoren / Aktoren

Die Liste wird fortlaufend, dynamisch von dem MUC-Controller an Hand der eingehenden Telegramme (primär üblicherweise per Wireless M-Bus) aktualisiert. Sie enthält alle sichtbaren Sensoren/ Aktoren¹³.

Ein Sensor/Aktor wird aus der Liste entfernt, falls von ihm über einen Zeitraum von mehr als 24 Stunden kein vollständiges Link-Layer-Telegramm fehlerfrei eingegangen ist. (Entfernen eines Sensors/Aktors aus der Liste der sichtbaren Geräte führt nicht zu dem Entfernen aus der Liste der aktiven Sensoren/Aktoren)

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Liste der sichtbaren Sensoren / Aktoren	81 81 10 06 FF FF	Nicht vorhanden	Listenelemente gemäß Pos. 2; Leere Listen sind zulässig.
2	Kind-Element des Baums Listen zur Beschreibung sichtbarer Komponenten	1. Liste 81 81 10 06 01 FF 2. Liste 81 81 10 06 02 FF ... usw. 250. Liste 81 81 10 06 FA FF	Nicht vorhanden	Listenelemente gemäß Pos. 3 Leere Listen sind zulässig.
3	Listenelemente zur Beschreibung sichtbarer Komponenten	Listenelemente zu 1: 81 81 10 06 01 NN Listenelemente zu 2:	Nicht vorhanden	Jeweils genau ein Element nach Pos. 4 bis 7 dieser Tabelle

¹³ Generell wird verlangt, dass sich unidirektional angebundene Sensoren / Aktoren mindestens alle 4 Stunden einmal melden.

		81 81 10 06 02 NN ... usw. Listenelemente zu 32: 81 81 10 06 FA NN Wertebereich NN: 0x01 - 0xFE	
4	Kind-Element des Baums Server ID der sichtbaren Komponente	81 81 C7 82 04 FF	• Octet String
5	Kind-Element des Baums Geräteklasse	81 81 C7 82 02 FF	• Typ: Octet String • Inhalt: Geräteklasse
6	Kind-Element des Baums Status (wann zuletzt empfangen; falls möglich, erfolgt die Angabe in UTC. Fehlt UTC, ist der Betriebssekundenzähler zu benutzen)	Bei UTC: 01 00 00 09 0b 00 Bei Betriebssekundenzähler 00 00 60 08 00 FF	• SML_Time Zeitstempel des letzten Empfangs
7	Kind-Element des Baums Server-ID des übergeordneten Gerätes („Parent ID“)	81 81 C7 82 14 FF	• Octet String

Tab. 49: Datenstruktur für die Liste der sichtbaren Sensoren / Aktoren

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.17 Datenstruktur mit Liste der aktiven Sensoren / Aktoren

Die Liste muss per Parametrierung (erfolgt üblicherweise im Zuge der Installation) geschrieben werden. Sie enthält alle aktivierten Sensoren / Aktoren. Das Schreiben erfolgt über spezielle Datenstrukturen im Sinne des Hinzufügens / Entfernens von Listenelementen.

Die Liste der aktiven Sensoren/ Aktoren ist unabhängig von der Liste der sichtbaren Sensoren/ Aktoren. Damit können – evtl. fälschlicherweise – in der Liste der aktiven Sensoren/ Aktoren auch Sensoren/ Aktoren zu finden sein, die nicht (mehr) in der Liste der sichtbaren Sensoren/ Aktoren zu finden sind.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Liste der aktiven Sensoren / Aktoren	81 81 11 06 FF FF	Nicht vorhanden	Listenelemente gemäß Pos. 2; Leere Listen sind zulässig.
2	Kind-Element des Baums Listen zur Beschreibung aktiver Komponenten	1. Liste 81 81 11 06 01 FF 2. Liste 81 81 11 06 02 FF ... usw. 250. Liste 81 81 11 06 FA FF	Nicht vorhanden	Listenelemente gemäß Pos. 3 Leere Listen sind zulässig.
3	Listenelemente zur	Listenelemente zu 1:	Nicht vorhanden	Jeweils genau ein

	Beschreibung aktiver Komponenten	81 81 11 06 01 NN Listenelemente zu 2: 81 81 11 06 02 NN ... usw. Listenelemente zu 32: 81 81 11 06 FA NN Wertebereich NN: 0x01 - 0xFE		Element nach Pos. 4 bis 7 dieser Tabelle
4	Kind-Element des Baums Server ID der aktiven Komponente	81 81 C7 82 04 FF	• Octet String	
5	Kind-Element des Baums Geräteklasse	81 81 C7 82 02 FF	• Typ: Octet String • Inhalt: Geräteklasse	
6	Kind-Element des Baums Status (wann zuletzt empfangen; falls möglich, erfolgt die Angabe in UTC. Fehlt UTC, ist der Betriebssekundenzähler zu benutzen)	Bei UTC: 01 00 00 09 0b 00 Bei Betriebssekundenzähler 00 00 60 08 00 FF	• SML_Time Zeitstempel des letzten Empfangs	
7	Kind-Element des Baums Server-ID des übergeordneten Gerätes („Parent ID“)	81 81 C7 82 14 FF	• Octet String	

Tab. 50: Datenstruktur für die Liste der aktivierten Sensoren / Aktoren

Das Lesen der Elemente ist von der jeweils mit dem Zugriff eingenommenen Rolle abhängig.

7.3.2.18 Datenstruktur zum Transport neuer sichtbarer Sensoren / Aktoren

Die Datenstruktur wird per SML_GetProcParameter per Push übertragen und dient der Bekanntgabe neuer, sichtbarer Sensoren / Aktoren:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Liste der sichtbaren Sensoren / Aktoren	81 81 10 16 FF FF	Nicht vorhanden	Listenelemente gemäß Pos. 2; Leere Listen sind zulässig.
2	Kind-Element des Baums Listen zur Beschreibung sichtbarer Komponenten	1. Liste 81 81 10 16 01 FF 2. Liste 81 81 10 16 02 FF ... usw. 250. Liste 81 81 10 16 FA FF	Nicht vorhanden	Listenelemente gemäß Pos. 3 Leere Listen sind zulässig.
3	Listenelemente zur Beschreibung sichtbarer Komponenten	Listenelemente zu 1: 81 81 10 16 01 NN Listenelemente zu 2: 81 81 10 16 02 NN ... usw.	Nicht vorhanden	Jeweils genau ein Element nach Pos. 4 bis 6 dieser Tabelle

		Listenelemente zu 32: 81 81 10 16 FA NN		
		Wertebereich NN: 0x01 - 0xFE		
4	Kind-Element des Baums Server ID der sichtbaren Komponenten	81 81 C7 82 04 FF	• Octet String	Einträge gemäß Pos. 3 und 4
5	Kind-Element des Baums Geräteklasse	81 81 C7 82 02 FF	• Typ: Octet String • Inhalt: Geräteklasse	
6	Kind-Element des Baums Status (wann zuletzt empfangen)	Bei UTC: 01 00 00 09 0b 00 Bei Betriebssekun- denzähler 00 00 60 08 00 FF	• SML_Time Zeitstempel des letzten Empfangs. Die Angabe erfolgt als ‚local Timestamp‘. Fehlt diese, ist der Be- triebssekunden- zähler zu benutzen.	

Tab. 51: Datenstruktur für die Liste neu erkannter Sensoren / Aktoren

7.3.2.19 Datenstruktur zum Transport verschwundener, zuvor sichtbarer Sensoren / Aktoren

Die Datenstruktur wird per SML_GetProcParameter per Push übertragen und dient der Bekanntgabe verschwundener, zuvor sichtbarer Sensoren / Aktoren:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Liste der sichtbaren Sensoren / Aktoren	81 81 10 26 FF FF	Nicht vorhanden	Listenelemente gemäß Pos. 2; Leere Listen sind zulässig.
2	Kind-Element des Baums Listen zur Beschreibung sichtbarer Komponenten	1. Liste 81 81 10 26 01 FF 2. Liste 81 81 10 26 02 FF ... usw. 250. Liste 81 81 10 26 FA FF	Nicht vorhanden	Listenelemente gemäß Pos. 3 Leere Listen sind zulässig.
3	Listenelemente zur Beschreibung sichtbarer Komponenten	Listenelemente zu 1: 81 81 10 26 01 NN Listenelemente zu 2: 81 81 10 26 02 NN ... usw. Listenelemente zu 32: 81 81 10 26 FA NN Wertebereich NN: 0x01 - 0xFE	Nicht vorhanden	Jeweils genau ein Element nach Pos. 4 bis 6 dieser Tabelle

4	Kind-Element des Baums Server ID der sichtbaren Komponenten	81 81 C7 82 04 FF	<ul style="list-style-type: none"> • Octet String 	Einträge gemäß Pos. 3 und 4
5	Kind-Element des Baums Geräteklasse	81 81 C7 82 02 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Geräteklasse 	
6	Kind-Element des Baums Status (wann zuletzt empfangen)	Bei UTC: 01 00 00 09 0b 00 Bei Betriebssekun- denzähler 00 00 60 08 00 FF	<ul style="list-style-type: none"> • SML_Time <p>Zeitstempel des letzten Empfangs.</p> <p>Die Angabe erfolgt als ‚local Timestamp‘. Fehlt diese, ist der Be- triebssekunden- zähler zu benutzen.</p>	

Tab. 52: Datenstruktur für die Liste „verschwundener“ Sensoren / Aktoren

7.3.2.20 Datenstruktur zum Transport deaktivierter Sensoren / Aktoren

Die Datenstruktur wird per SML_GetProcParameter per Push übertragen und dient der Bekanntgabe deaktivierter Sensoren / Aktoren:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Liste der aktivierten Sensoren / Aktoren	81 81 11 26 FF FF	Nicht vorhanden	Listenelemente gemäß Pos. 2; Leere Listen sind zulässig.
2	Kind-Element des Baums Listen zur Beschreibung aktivierten Komponenten	1. Liste 81 81 11 26 01 FF 2. Liste 81 81 11 26 02 FF ... usw. 250. Liste 81 81 11 26 FA FF	Nicht vorhanden	Listenelemente gemäß Pos. 3 Leere Listen sind zulässig.
3	Listenelemente zur Beschreibung aktivierten Komponenten	Listenelemente zu 1: 81 81 11 26 01 NN Listenelemente zu 2: 81 81 11 26 02 NN ... usw. Listenelemente zu 32: 81 81 11 26 FA NN Wertebereich NN: 0x01 - 0xFE	Nicht vorhanden	Jeweils genau ein Element nach Pos. 4 bis 6 dieser Tabelle
4	Kind-Element des Baums Server ID der sichtbaren Komponenten	81 81 C7 82 04 FF	<ul style="list-style-type: none"> • Octet String 	Einträge gemäß Pos. 3 und 4

5	Kind-Element des Baums Geräteklasse	81 81 C7 82 02 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Geräteklasse
---	--	-------------------	---

Tab. 53: Datenstruktur für die Liste deaktivierter Sensoren / Aktoren**7.3.2.21 Datenstruktur zum Transport aktivierter Sensoren / Aktoren**

Die Datenstruktur wird per SML_GetProcParameter per Push übertragen und dient der Bekanntgabe aktivierter Sensoren / Aktoren:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Liste der aktivierten Sensoren / Aktoren	81 81 11 16 FF FF	Nicht vorhanden	Listenelemente gemäß Pos. 2; Leere Listen sind zulässig.
2	Kind-Element des Baums Listen zur Beschreibung aktivierten Komponenten	1. Liste 81 81 11 16 01 FF 2. Liste 81 81 11 16 02 FF ... usw. 250. Liste 81 81 11 16 FA FF	Nicht vorhanden	Listenelemente gemäß Pos. 3 Leere Listen sind zulässig.
3	Listenelemente zur Beschreibung aktivierten Komponenten	Listenelemente zu 1: 81 81 11 16 01 NN Listenelemente zu 2: 81 81 11 16 02 NN ... usw. Listenelemente zu 32: 81 81 11 16 FA NN Wertebereich NN: 0x01 - 0xFE	Nicht vorhanden	Jeweils genau ein Element nach Pos. 4 bis 6 dieser Tabelle
4	Kind-Element des Baums Server ID der sichtbaren Komponenten	81 81 C7 82 04 FF	• Octet String	Einträge gemäß Pos. 3 und 4
5	Kind-Element des Baums Geräteklasse	81 81 C7 82 02 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Geräteklasse 	
6	Kind-Element des Baums Status (wann zuletzt empfangen)	Bei UTC: 01 00 00 09 0b 00 Bei Betriebssekundenzähler 00 00 60 08 00 FF	<ul style="list-style-type: none"> • SML_Time Zeitstempel des letzten Empfangs. Die Angabe erfolgt als ‚local Timestamp‘. Fehlt diese, ist der Betriebssekundenzähler zu benutzen. 	

Tab. 54: Datenstruktur für die Liste aktivierter Sensoren / Aktoren

7.3.2.22 Datenstruktur zum Ergänzen eines aktiven Sensors / Aktors

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung des Kommandos: ‚Sensor / Aktor aufnehmen‘	81 81 11 06 FB FF	Nicht vorhanden	Mindestens ein Element nach Pos. 2
2	Kind-Element des Baums Server ID des zu aktivierenden Sensors	81 81 11 06 FB NN	Nicht vorhanden	Genau ein Element nach Pos. 3
3	Kind-Element des Baums Server ID des zu aktivierenden Sensors	81 81 C7 82 04 FF	• Octet String	Nicht vorhanden

Tab. 55: Datenstruktur zum Aktivieren eines Sensors / Aktors
7.3.2.23 Datenstruktur zum Deaktivieren eines Sensors / Aktors

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung des Kommandos: ‚Sensor / Aktor deaktivieren‘	81 81 11 06 FC FF	Nicht vorhanden	Mindestens ein Element nach Pos. 2
2	Kind-Element des Baums Server ID des zu deaktivierenden Sensors	81 81 11 06 FC NN	Nicht vorhanden	Genau ein Element nach Pos. 3
3	Kind-Element des Baums Server ID des zu deaktivierenden Sensors	81 81 C7 82 04 FF	• Octet String	Nicht vorhanden

Tab. 56: Datenstruktur zum Deaktivieren eines Sensors / Aktors
7.3.2.24 Datenstruktur zum Entfernen / Löschen eines Sensors / Aktors

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung des Kommandos: ‚Sensor / Aktor entfernen‘	81 81 11 06 FD FF	Nicht vorhanden	Mindestens ein Element nach Pos. 2
2	Kind-Element des Baums Server ID des zu entfernen Sensors	81 81 11 06 FD NN	Nicht vorhanden	Genau ein Element nach Pos. 3
3	Kind-Element des Baums Server ID des zu entfernen Sensors	81 81 C7 82 04 FF	• Octet String	Nicht vorhanden

Tab. 57: Datenstruktur zum Entfernen eines Sensors / Aktors

7.3.2.25 Datenstruktur zum Auftrag ‚Kommando Reset ausführen‘

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung des Kommandos: ‚Reset‘	81 81 C7 83 82 01	Nicht vorhanden	Nicht vorhanden

Tab. 58: Datenstruktur zum Auslösen von Reset
7.3.2.26 Datenstruktur zur Abfrage der Geräte-Identifikation

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Abfrage der Geräte-Identifikation.	81 81 C7 82 01 FF	Nicht vorhanden	Nicht vorhanden

Tab. 59: Datenstruktur zur Abfrage der Geräte-Identifikation
7.3.2.27 Datenstruktur zur Antwort der Geräte-Identifikation

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Antwort der Geräte-Identifikation.	81 81 C7 82 01 FF	Nicht vorhanden	Jeweils genau ein Element gemäß der Pos. 2 bis 5, 10 und optional 12 dieser Tabelle.
2	Kind-Element im Baum. Enthält die Information zur Geräteklasse.	81 81 C7 82 02 FF	Typ: Octet String Inhalt: Geräteklasse	Nicht vorhanden
3	Kind-Element im Baum. Enthält die Information zum Hersteller.	81 81 C7 82 03 FF	Typ: Octet String Inhalt: FLAG-Kennung	Nicht vorhanden
4	Kind-Element im Baum. Enthält die Server-Identifikation	81 81 C7 82 04 FF	Typ: Octet String Inhalt: Server-Id	Nicht vorhanden
5	Kind-Element im Baum. Enthält die Liste der Firmware-Abschnitte mit deren Versionsangaben.	81 81 C7 82 06 FF	Nicht vorhanden	Ein oder mehrere Elemente nach Position 6 dieser Tabelle
6	Kind-Element im Baum. Enthält die Information zu einem Firmware-Abschnitt.	Firmware: 81 81 C7 82 07 NN Datei: 81 81 C7 82 0C NN Applikation: 81 81 C7 82 0D NN	Nicht vorhanden	Ein Element nach Position 7 sowie danach ein Element der Position 8 dieser Tabelle
7	Kind-Element im Baum. Enthält den herstellereigenen Namen zu einem Firmware-Abschnitt.	81 81 C7 82 08 FF	Typ: Octet String Inhalt: Name	Nicht vorhanden

8	Kind-Element im Baum. Enthält die herstellerspezifische Versionsnummer zu einem Firmware-Abschnitt.	81 81 00 02 00 00	Typ: Octet String Inhalt: Version	Nicht vorhanden
9	Kind-Element im Baum. Enthält das Merkmal „aktiv“ zu einem Firmware-Abschnitt.	81 81 C7 82 0E NN	Typ: Boolean Inhalt: True ⇔ Aktiviert	Nicht vorhanden
10	Kind-Element im Baum. Enthält die Liste der herstellerspezifischen Beschreibungen zur Hardware-Ausstattung (Charge, Typ, ...)	81 81 C7 82 09 FF	Nicht vorhanden	Ein oder mehrere Elemente nach Position 10 dieser Tabelle
11	Kind-Element im Baum. Enthält einen herstellerspezifischen Freitext	81 81 C7 82 0A NN	Typ: Octet String Inhalt: Freitext	Nicht vorhanden
12	Kind-Element im Baum. Enthält den Public-Key zum MUC (benötigt bei künftigen Anwendungen)	81 81 C7 82 0B FF	Typ: Octet String Inhalt: Public-Key	Nicht vorhanden
13	Kind-Element im Baum. Enthält die Liste der maximal vom Gerät unterstützten Einträge bei Aktionsschaltprogrammen	81 81 C7 82 10 FF	Nicht vorhanden	Je ein Element nach Pos. 14 bis 18
14	Kind-Element im Baum. Enthält die maximal unterstützte Anzahl von Regelsätzen (siehe Tab. 36)	81 81 C7 82 10 01	Typ: Unsigned8 Inhalt: 0..255 '0' bedeutet, dass das Geräte nicht über die Option 'Aktionsschaltprogramm' verfügt.	Nicht vorhanden
15	Kind-Element im Baum. Enthält die maximal unterstützte Anzahl von Saisonprogrammen (siehe Tab. 37)	81 81 C7 82 10 02	Typ: Unsigned8 Inhalt: 0..55 '0' ist nur zulässig, wenn unter 'Regelsätzen' ebenfalls eine '0' zu finden ist.	Nicht vorhanden
16	Kind-Element im Baum. Enthält die maximal unterstützte Anzahl von Tagesprogrammen (siehe Tab. 38)	81 81 C7 82 10 03	Typ: Unsigned8 Inhalt: 0..55 '0' ist nur zulässig, wenn unter 'Regelsätzen' ebenfalls eine '0' zu finden ist.	Nicht vorhanden

17	Kind-Element im Baum. Enthält die maximal unterstützte Anzahl von Regeln (siehe Tab. 39)	81 81 C7 82 10 04	Typ: Unsigned8 Inhalt: 0..254 '0' ist nur zulässig, wenn unter 'Regelsätzen' ebenfalls eine '0' zu finden ist.	Nicht vorhanden
18	Kind-Element im Baum. Enthält die maximal unterstützte Anzahl von Aktionen (siehe Tab. 40)	81 81 C7 82 10 05	Typ: Unsigned16 Inhalt: 0..254x254 '0' ist nur zulässig, wenn unter 'Regelsätzen' ebenfalls eine '0' zu finden ist.	Nicht vorhanden

Tab. 60: Datenstruktur mit Antwort zur Geräteidentifikation

7.3.2.28 Datenstruktur zum remote Firmware-/Datei-Download (Übertragung)

Die Übertragung erfolgt dabei als Auftrag der Art „SetProcParameterRequest“ wobei folgende Datenstruktur einzusetzen ist:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport einer Firmware/Datei Liefert die Information zur Authorisierung sowie das zu ladende Binary. Enthält mindestens einen und optionale mehrere Verweise auf das / die zu ladenden Binaries.	Je nach Datei: Firmware: 81 81 C7 81 01 FF Firmware (indirekt adressiert): 81 81 C7 81 0E FF Datei: 81 81 C7 81 0C FF Datei (indirekt adressiert): 81 81 C7 81 0F FF Applikation: 81 81 C7 81 0D FF Application (indirekt adressiert): 81 81 C7 81 10 FF	Nicht vorhanden	Elemente Pos. 2 und Pos. 3 dieser Tabelle gefolgt von mindestens einem oder mehreren Elementen nach Pos. 4 dieser Tabelle.
2	Kind-Element im Baum. Enthält den Namen zum zu ladenden Binary.	81 81 00 02 00 02	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Firmware-/Datei-Name 	Nicht vorhanden
3	Kind-Element im Baum. Nummer der Nachricht.	81 81 00 02 00 05	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Nachrichtennr., beginnend mit 1 	Nicht vorhanden

4	Kind-Element im Baum. Enthält die Information zur Authorisierung zum Binary. Verweist auf die Kind-Elemente mit dem Blocknummer14 und dem zu ladenden Binary.	81 81 00 02 01 NN (NN ist beginnend mit 01 für das erste Listenelement zu bilden)	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Signatur zur Authorisierung 	Element 1: Siehe Pos. 5 Element 2: Siehe Pos. 6
5	Kind-Element im Baum. Enthält die Blocknummer, an den das zu ladende Binary gerichtet ist.	81 81 00 02 02 FF	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Blocknummer 	Nicht vorhanden
6	Kind-Element im Baum. Enthält das zu ladende Binary.	81 81 00 02 03 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Binary 	Nicht vorhanden

Tab. 61: Datenstruktur zum Übertragen einer Firmware / Datei

Indirekt adressierte Binaries sind zur Verteilung an nachgelagerte Geräte vorgesehen.

Der Name des zu ladenden Binary muss eindeutig in Bezug auf die sechs Namensräume Firmware, Firmware (indirekt adresiert), Datei, Datei (indirekt adresiert), Applikation und Applikation (indirekt adresiert) sein. Damit darf derselbe Name nicht mehrfach in demselben Namensraum benutzt werden. Wird ein Binary mit demselben Namen übertragen, ist als Fehlermeldung eine SML_Attention mit Code 81 81 C7 C7 E1 21 zu benutzen.

Es ist beabsichtigt, die Definitionen zur Benutzung von Datei- wie auch Applikations-Binaries bei Bedarf künftig zu erweitern.

7.3.2.29 Datenstruktur zum Auftrag ‚Firmware/Datei aktivieren‘

Das Kommando ‚Firmware/Datei aktivieren‘ wird durch Schreiben per ‚SetProcParameterRequest‘ unter Angabe des Befehlswords mit der Kennzahl ‚81 81 C7 83 82 07‘ aktiviert. Das betroffene Modul antwortet auf diesen Schreibbefehl mit ‚AttentionResponse‘.

Es ist Aufgabe der jeweils aktiven Firmware zu gewährleisten, dass bei Aktivieren / Deaktivieren / Entfernen das Gerät danach in einem betriebsfähigen Zustand und weiterhin über die Weitverkehrsschnittstelle erreichbar bleibt.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung des Kommandos: ‚Firmware / Datei aktivieren‘	Firmware: 81 81 C7 83 82 07 Firmware (indirekt adressiert): 81 81 C7 83 82 0E Datei: 81 81 C7 83 82 0C Datei (indirekt adressiert):	Optional: Octet String Name des betroffenen Binary	Genau ein Element nach Pos. 2 und 3 dieser Tabelle.

¹⁴ Die Blocknummer wird vom Hersteller festgelegt.

		81 81 C7 83 82 0F		
		Applikation: 81 81 C7 83 82 0D		
		Applikation (indirekt adressiert): 81 81 C7 83 82 10		
2	Kind-Element im Baum. Enthält den Index des von der Aktion betroffenen Binary	81 81 C7 83 83 01	<ul style="list-style-type: none"> • Typ: Unsigned8 • Inhalt: 0 ⇔ das jeweils zuletzt geladene Binary, 1 ⇔ das jeweils vorletzte geladene Binary 2 ⇔ das durch den Namen ausgewählte Binary 	Nicht vorhanden
3	Kind-Element im Baum. Enthält die durchzuführende Aktion	81 81 C7 83 83 03	<ul style="list-style-type: none"> • Typ: Unsigned8 • Inhalt: 0 ⇔ entfernen, 1 ⇔ aktivieren, 2 ⇔ deaktivieren, nur bei indirekter Adressierung: 3 ⇔ Verteilung starten, 4 ⇔ reserviert, 5 ⇔ Verteilung pausieren, 6 ⇔ pausierte Verteilung fortsetzen, 7 ⇔ Verteilung abbrechen, 8 ⇔ zur Verteilung anstehendes Binary entfernen, alle anderen sind reserviert 	Nicht vorhanden
4	Kind-Element im Baum. Enthält den Zeitpunkt, wann die Aktion durchgeführt werden soll Fehlt dieses Element, ist die Aktion sofort auszuführen	81 81 C7 83 83 04	<ul style="list-style-type: none"> • Typ: SML_Time • Inhalt: Zeitpunkt 	Nicht vorhanden

Tab. 62: Datenstruktur zum Aktivieren einer Firmware / Datei

Die mit ‚indirekt adressiert‘ bezeichneten Nachrichten weisen das direkt in der SML_SetProcParameter-Nachricht per Server-ID adressierte Gerät dazu an, in Interaktion mit nachgeordneten Geräten zu treten.

Aktion Entfernen:

Das Entfernen ist nur dann zu lässig, wenn das betroffene Binary nicht aktiviert ist. Ist das betroffene Binary aktiviert und wird die Aktion ‚Entfernen‘ verlangt, ist als Fehlermeldung die SML_Attention mit Nummer 81 81 C7 FE 0A („Auftrag nicht ausgeführt“) zu benutzen.

7.3.2.30 Auftrag/Antwort mit GET-Kontrollstruktur zur Statusabfrage des Firmware-/Datei-Downloads

Die Kontrollstruktur zur Statusabfrage des Firmware-/Datei-Downloads wird durch Lesen per „GetProcParameterRequest“ unter Angabe der Kennzahl „81 81 00 02 00 01“ hergestellt.

Die Systemkomponente, die diese Nachricht entgegen nimmt, antwortet darauf gemäß SML mit folgender Datenstruktur:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung der Antwort	81 81 00 02 00 01	Nicht vorhanden	Genau je ein Kind-Element nach Pos. 2 bis Pos. 4 dieser Tabelle
2	1. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Firmware-/Datei-Name	81 81 00 02 00 02	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Bezeichner 	Nicht vorhanden
3	2. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Anzahl aller Nachrichten zur Übertragung des Binary	81 81 00 02 00 03	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Nachrichten-Anzahl 	Nicht vorhanden
4	3. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Nummer der zuletzt erfolgreich übertragenen Nachricht des Binary	81 81 00 02 00 04	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Nachrichten-Nummer 	Nicht vorhanden

Tab. 63: Datenstruktur für Abfrage zum Stand der Firmware / Datei-Übertragung

7.3.2.31 Auftrag mit SET-Kontrollstruktur zum Start des Firmware-Datei-Downloads

Die SET-Kontrollstruktur zum Start des Firmware-Downloads wird durch Schreiben per „SetProcParameterRequest“ unter Angabe der Kennzahl „81 81 00 02 00 01“ hergestellt.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung des Kommandos: Auftrag zum Start eines Firmware-/Datei-Downloads	81 81 00 02 00 01	Nicht vorhanden	Genau je ein Kind Element nach Pos. 2 bis 3 dieser Tabelle
2	1. Kind-Element des Baums zur Übertragung eines Kommandos; Firmware-/Datei-Name	81 81 00 02 00 02	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Bezeichner 	Nicht vorhanden
3	2. Kind-Element des Baums zur	81 81 00 02 00 03	• Typ:	Nicht vorhanden

	Übertragung eines Kommandos; Anzahl aller Nachrichten zur Übertragung des Binary	81 81 00 02 00 04	Unsigned32 • Inhalt: Block-Anzahl	
4	3. Kind-Element des Baums zur Übertragung eines Kommandos; Nummer der zuletzt erfolgreich übertragenen Nachricht des Binary	81 81 00 02 00 04	• Typ: Unsigned32 • Inhalt: Bei SET ist immer ‚0‘ anzugeben	Nicht vorhanden

Tab. 64: Datenstruktur zum Start der Firmware / Datei-Übertragung

Ein Modul, das diese Nachricht entgegen nimmt, antwortet darauf gemäß SML mit einer Nachricht vom Typ „AttentionResponse“. Konnte das Kommando korrekt ausgeführt werden, wird als Antwortcode „OK“ (siehe SML) geliefert. Trat ein Fehler in der Bearbeitung auf, wird eine der mit SML definierten Fehlernummern erzeugt.

Falls bei der Anfrage ein unbekannter „Firmware-/Datei-Name“ angegeben ist, soll als Response eine Fehlermeldung (siehe SML) erzeugt werden.

Das Attribute „Firmware-/Datei-Name“ dient im Sinne einer Authentisierung für die laufende Firmware zur Erkennung, ob die zu ladende Datei akzeptiert werden kann.

7.3.2.32 Auftrag/Antwort zur Statusabfrage des Multicast-Firmware-Download

Die Kontrollstruktur zur Statusabfrage des Multicast-Firmwate-Download wird durch Lesen per „GetProcParameterRequest“ unter Angabe der Kennzahl „81 81 00 03 00 01“ hergestellt.

Die Systemkomponente, die diese Nachricht entgegen nimmt, antwortet darauf gemäß SML mit der folgenden Datenstruktur:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung der Antwort	81 81 00 03 00 01	Nicht vorhanden	Genau je ein Kind-Element nach Pos. 2 bis 4 dieser Tab.
2	1. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Multicast-Firmware-/Datei-Name	81 81 00 03 00 02	• Typ: Octet String • Inhalt: Bezeichner	Nicht vorhanden
3	2. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Anzahl aller Nachrichten zur Übertragung des Binary	81 81 00 03 00 03	• Typ: Unsigned32 • Inhalt: Nachrichten-Anzahl	Nicht vorhanden
4	3. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Nummer der zuletzt erfolgreich übertragenen Nachricht des Binary	81 81 00 03 00 04	• Typ: Unsigned32 • Inhalt: Nachrichten-Nummer	Nicht vorhanden

Tab. 65: Datenstruktur zur Statusabfrage für Multicast-Firmware-Download

7.3.2.33 Auftrag/Antwort zur Datenübertragung des Multicast-Firmware-Download

Der Start des Multicast-Firmware-Download erfolgt per „SetProcParameterRequest“ unter Angabe der Kennzahl 81 81 00 03 00 01.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung der Antwort	81 81 00 03 00 01	Nicht vorhanden	Genau je ein Kind-Element nach Pos. 2 bis Pos. 4 dieser Tabelle
2	1. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Multicast-Firmware-/Datei-Name	81 81 00 03 00 02	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Bezeichner 	Nicht vorhanden
3	2. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Anzahl aller Nachrichten zur Übertragung des Binary	81 81 00 03 00 03	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Block-Anzahl 	Nicht vorhanden
4	3. Kind-Element des Baums zur Übertragung der Kommando-Antwort: Nummer der zuletzt erfolgreich übertragenen Nachricht des Binary	81 81 00 03 00 04	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Bei SET ist immer ‚0‘ anzugeben 	Nicht vorhanden

Tab. 66: Datenstruktur zur Übertragung beim Multicast-Firmware-Download

Ein Modul, das diese Nachricht entgegen nimmt, antwortet darauf gemäß SML mit einer Nachricht vom Typ „AttentionResponse“. Konnte das Kommando korrekt ausgeführt werden, wird als Antwortcode „OK“ (siehe SML) geliefert. Trat ein Fehler in der Bearbeitung auf, wird eine der mit SML definierten Fehlernummern erzeugt.

Falls bei der Anfrage ein unbekannter „Multicast-Firmware-/Datei-Name“ angegeben ist, soll als Response eine Fehlermeldung (siehe SML) erzeugt werden.

Das Attribute „Multicast-Firmware-/Datei-Name“ dient im Sinne einer Authentisierung für die laufende Firmware zur Erkennung, ob die zu ladende Datei akzeptiert werden kann.

7.3.2.34 Datenstruktur zur Abfrage dynamischer Multicast-Firmware-Download-Betriebsparameter

Die Abfrage des dynamischen Zustands eines Multicast-Firmware-Download erfolgt per GetProcParameter:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Transport der Multicast-Firmware-Parameter	81 04 0E 07 00 FF	Nicht vorhanden	Listeneinträge gemäß Position 2 dieser Tabelle
2	Kind-Element im Baum Knoten Mit Multicast-Firmware Status der angegebenen Ebene	81 04 0E 07 00 NN NN = 01..08 (01 ⇔ Gesamtstatus)	Nicht vorhanden	Einträge der Positionen 3 bis 7 (bzw. 6)

3	Kind-Element im Baum Anzahl der aktiven Kinder	81 04 0E 07 01 FF	Unsigned32
4	Kind-Element im Baum Anzahl der abgeschlossenen Kinder	81 04 0E 07 02 FF	Unsigned32
5	Kind-Element im Baum Anzahl der fehlgeschlagenen Kinder	81 04 0E 07 03 FF	Unsigned32
6	Kind-Element im Baum Fortschritt angegeben in %	81 04 0E 07 04 FF	Unsigned32

Tab. 67: Datenstruktur zur Abfrage des Status beim Multicast-Firmware-Download
7.3.2.35 Datenstruktur zur Übertragung des Hersteller-spezifischen Ereignis-Logbuches

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Wert	Listenelement
1	Wurzel-Element eines optionalen Baums mit weiteren Kind-Elementen, deren Umfang, Inhalt und Kennzahlen der Hersteller festlegt.	81 81 C7 81 21 FF	Optional vorhanden, Aufbau legt Hersteller fest	Optional vorhanden, Aufbau legt Hersteller fest

Tab. 68: Datenstruktur zur Abfrage des Hersteller-spezifischen Logbuchs
7.3.2.36 Datenstruktur zur Abfrage der NTP-Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Abfrage der NTP-Parameter.	81 81 C7 88 01 FF	Nicht vorhanden	Nicht vorhanden

Tab. 69: Datenstruktur zur Abfrage der NTP-Parameter
7.3.2.37 Datenstruktur mit Antwort / zum Setzen der NTP-Parameter

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums mit den NTP-Parametern.	81 81 C7 88 01 FF	Nicht vorhanden	Jeweils mindestens ein Element gemäß der Positionen 2, 4 und 7 dieser Tabelle und optional die Elemente 5 und 6.
2	Kind-Element im Baum: Liste der Adressen der NTP-Server	81 81 C7 88 02 FF	Nicht vorhanden	Mindestens ein Element gemäß Position 3 dieser Tabelle
3	Kind-Element im Baum Enthält die IP-Adresse des NTP-Servers (NN = 01..04).	81 81 C7 88 02 NN	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: IP-Adresse oder Hostname 	Nicht vorhanden

4	Kind-Element im Baum. Enthält die Port-Nummer des NTP-Dienstes.	81 81 C7 88 03 FF	<ul style="list-style-type: none"> • Typ: Unsigned16 • Inhalt: Port-Nummer 	Nicht vorhanden
5	Kind-Element im Baum. Enthält die Periode zur Versendung des Synchron-Token. Diese Periode wird mit der Genauigkeit des Betriebssekundenzählers bestimmt.	81 81 C7 88 04 FF	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Periode, angegeben in Sekunden 	Nicht vorhanden
6	Kind-Element im Baum. Enthält den Versatz zur Versendung des Synchron-Signals zur Synchronisation angeschlossener Sensoren / Aktoren bezogen auf 00:00:00. Dieser Versatz wird einmalig mit Erreichen der Betriebsbereitschaft angewendet; alle weiteren Ausgaben zum Synchron-Signal erfolgen mit der vorgenannten Periode.	81 81 C7 88 05 FF	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Versatz / Offset, angegeben in Sekunden 	Nicht vorhanden
7	Kind-Element im Baum. Aktiviert / De-Aktiviert den Zugriff auf einen NTP-Server.	81 81 C7 88 06 FF	<ul style="list-style-type: none"> • Typ: Boolean • Inhalt: Bei ‚true‘ wird die periodische Erzeugung von Synchron-Token aktiviert 	Nicht vorhanden

Tab. 70: Datenstruktur für Zugriff auf die NTP-Parameter**7.3.2.38 Datenstruktur zur Abfrage der Zeitinformation**

Damit an einen MUC-Controller angeschlossene Sensoren / Aktoren Zugriff auf die im MUC-Controller verfügbare Zeitinformation haben, müssen MUC-Controller die nachstehend definierte Anfrage zur Zeitinformation beantworten können.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Abfrage der Zeitinformation.	81 81 C7 88 10 FF	Nicht vorhanden	Nicht vorhanden

Tab. 71: Datenstruktur für Abfrage der Zeitinformation**7.3.2.39 Datenstruktur mit Antwort der Zeitinformation**

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums mit der Zeitinformation.	81 81 C7 88 10 FF	Nicht vorhanden	Jeweils genau ein Element gemäß der Pos. 2 bis 5 dieser Tabelle.

2	Kind-Element im Baum. Aktuelle Zeit	01 00 00 09 0B 00	<ul style="list-style-type: none"> • Typ: SML_Time • Inhalt: Local Timestamp 	Nicht vorhanden
3	Kind-Element im Baum. Aktuellen Betriebssekundenzähler	00 00 60 08 00 FF	<ul style="list-style-type: none"> • Typ: SML_Time • Inhalt: Sekundenindexr 	Nicht vorhanden
4	Kind-Element im Baum. Offset zur aktuellen Zeitzone	81 00 00 09 0B 01	<ul style="list-style-type: none"> • Typ: Integer16 • Inhalt: Offset in Minuten • Wertebereich: -720 bis +720 	Nicht vorhanden
5	Kind-Element im Baum. Merkmal: UTC wurde innerhalb der Ganggenauigkeit synchronisiert	81 00 00 09 0B 02	<ul style="list-style-type: none"> • Typ: Boolean • Inhalt: TRUE ⇔ Synchronisiert 	Nicht vorhanden

Tab. 72: Datenstruktur für Zugriff auf die Zeitinformation

7.3.2.40 Datenstruktur zum Leeren eines Datensammlers

Das Leeren des Datensammlers wird durch Schreiben per „SetProcParameterRequest“ unter Angabe des Kommandos „81 81 C7 83 82 05“ aktiviert. Die adressierte Systemkomponente antwortet auf diesen Schreibbefehl mit AttentionResponse.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zur Übertragung des Kommandos: Leeren des Datensammlers	81 81 C7 83 82 05	Nicht vorhanden	Je ein Element nach Pos. 2 und 3 dieser Tabelle
2	Kind-Element im Baum. Server-ID zum Datenspiegel, dessen Datensammler geleert werden soll	81 81 C7 82 04 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Server-Identifizier 	Nicht vorhanden
3	Kind-Element im Baum. Kennzahl zum Datensammler, der geleert werden soll	81 81 C7 8A 83 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Kennzahl 	Nicht vorhanden

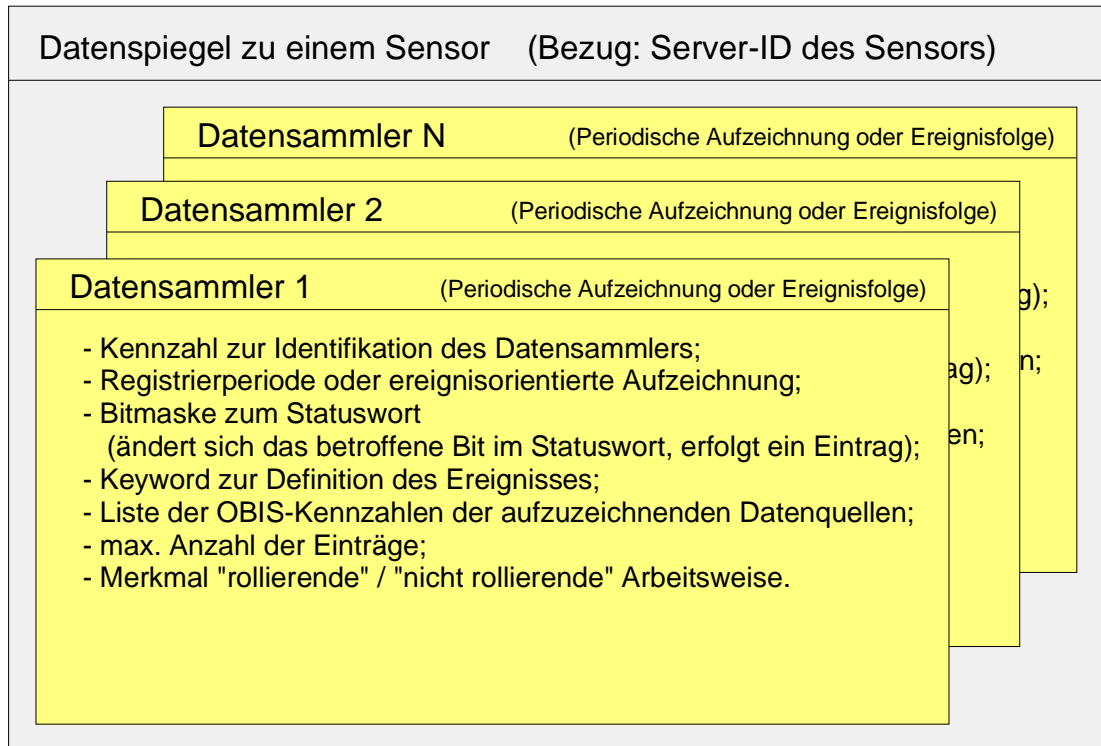
Tab. 73: Datenstruktur zum Leeren eines Datensammlers

7.3.2.41 Datenstruktur der Eigenschaften eines Datenspiegels

Ein Datenspiegel hält als Container die Verwaltungsinformationen zu einem Sensor / Aktor, so dass dessen Informationen (Zählerstände, ...) in einem Datensammler im MUC-Controller aufgezeichnet werden können.

Sobald ein Sensor / Aktor in die Liste der aktivierten Sensoren / Aktoren aufgenommen wird, verfügt er auch über den Datenspiegel. Wird er aus der Liste der aktivierten Sensoren / Aktoren entfernt, werden auch sein Datenspiegel sowie alle an diesem Datenspiegel hängenden Datensammler entfernt. Wird er erneut in die Liste der aktivierten Sensoren / Aktoren aufgenommen, erhält er einen neuen Datenspiegel.

Datenspiegel fassen damit alle Informationen zu einem Sensor / Aktor zusammen:



Parametrierung zu einem der Datensammler 1..N auf Basis von:

- Server-ID zum MUC,
- Kennzahl zur Identifikation der Datensammler-Beschreibung,
- Username (für MUC-Parametrier-Zugriff),
- Password (für MUC-Parametrier-Zugriff).

Bild 25: Datenspiegel / Datensammler (Parametrierung)

Datenbeschaffung zu einem der Datensammler 1..N auf Basis von:

- Server-ID zum Sensor (nicht zum MUC),
- Kennzahl zur Identifikation des Datensammlers,
- Username,
- Password.

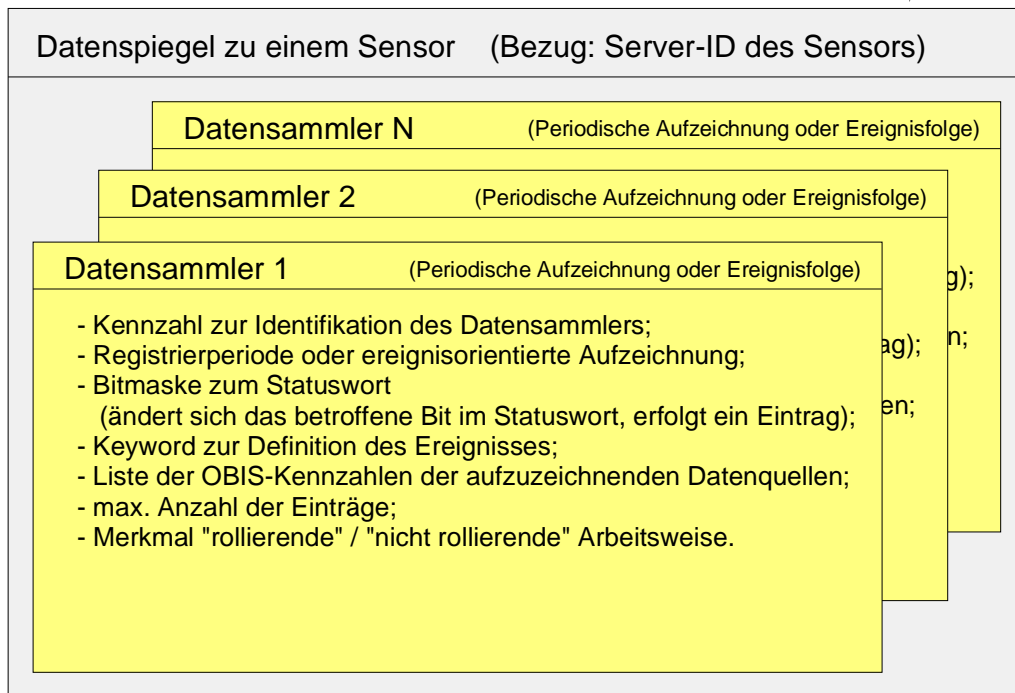


Bild 26: Datensammler (Zugriff zur Datenbeschaffung)

Die Eigenschaften zum Datenspiegel können via „GetProcParameter“ gelesen sowie per „SetProcParameter“ geschrieben werden. Der Zugriff ist nur dann möglich, wenn der zugehörige Sensor / Aktor aktiviert worden ist (siehe Bild 4 / Bild 5).

Das Kommando zu Lesen oder Schreiben muss an den zugehörigen Sensor / Aktor per Server-ID des Sensors / Aktors adressiert werden. Wird das Kommando an eine MUC-Controller adressiert, antwortet der MUC-Controller mit der Nachricht ‚AttentionResponse‘ und einem Fehler-Kode.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums mit den Eigenschaften eines Datenspiegels	81 81 C7 86 00 FF	Nicht vorhanden	Je ein Element der Positionen 2 bis 13
2	Kind-Element im Baum. Enthält die Server-Identifikation zur Datenquelle für den Datenspiegel	81 81 C7 82 04 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Server-Id 	Nicht vorhanden
3	Kind-Element im Baum. Enthält die Information zur Geräteklasse.	81 81 C7 82 02 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Geräteklasse 	Nicht vorhanden

4	Kind-Element im Baum. Enthält die Information zum Hersteller.	81 81 C7 82 03 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: FLAG-Kennung 	Nicht vorhanden
5	Kind-Element im Baum Enthält die zuletzt empfangene Statusinformation	81 00 60 05 00 00	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Statuswort 	Nicht vorhanden
6	Kind-Element im Baum Enthält die Bitmaske zur Definition von Bits des Statusworts, deren Änderung zu einem Eintrag im Betriebslogbuch zum Datenspiegel führt	81 81 C7 86 01 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Bitmaske 	Nicht vorhanden
7	Kind-Element im Baum Durchschnittliche Zeit zwischen zwei empfangenen Datensätzen	81 81 C7 86 02 FF	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: Zeit in ms 	Nicht vorhanden
8	Kind-Element im Baum Zeitstempel des MUC-Controllers, gebildet bei Eintreffen des letzten Datensatzes	Bei kalendarischer Zeitinformation: 01 00 00 09 0b 00 Bei Betriebs- sekundenzähler 00 00 60 08 00 FF	<ul style="list-style-type: none"> • Typ: SML_Time • Inhalt: Zeitstempel 	Nicht vorhanden
9	Kind-Element im Baum. Enthält den Public Key	81 81 C7 82 05 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Public Key 	Nicht vorhanden
10	Kind-Element im Baum. Enthält den AES-Schlüssel zum Wireless-M-Bus	81 81 C7 86 03 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: AES-Schlüssel 	Nicht vorhanden
11	Kind-Element im Baum. Enthält den Username zum Zugriff auf den Sensor / Aktor	81 81 61 3C 01 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Username 	Nicht vorhanden
12	Kind-Element im Baum. Enthält das Passwort zum Zugriff auf den Sensor / Aktor	81 81 61 3C 02 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Passwort 	Nicht vorhanden
13	Kind-Element im Baum. Zeitbezug für Datenspiegel und zugehörige Push-Vorgänge	81 81 C7 86 04 FF	<ul style="list-style-type: none"> • Typ: Unsigned8 • Inhalt: 0 ⇔ UTC, 1 ⇔ UTC plus Zeitzone 2 ⇔ lokale Zeit. 	Nicht vorhanden

Tab. 74: Datenstruktur für Parameter zum Datenspiegel

7.3.2.42 Liste vordefinierter Kennzahlen zur Datensammler-Identifikation

Zur Identifikation der Datensammler werden Kennzahlen nach OBIS verwendet. Im Sinne einer einheitlichen Benutzung wird folgende Zuordnung verlangt:

Pos.	OBIS-Kennzahl	Bedeutung
1	81 81 C7 86 10 FF	Standard-Lastgang mit minütlicher Registrierperiode und Zählerständen (Aufzeichnung zur vollen Minute)
2	81 81 C7 86 11 FF	Standard-Lastgang mit ¼ h Registrierperiode und Zählerständen (Aufzeichnung zur vollen Viertelstunde)
3	81 81 C7 86 12 FF	Standard-Lastgang mit 1 h Registrierperiode und Zählerständen (Aufzeichnung zur vollen Stunde)
4	81 81 C7 86 13 FF	Standard-Lastgang mit 24 h Registrierperiode und Zählerständen (Aufzeichnung zum Tageswechsel)
5	81 81 C7 86 14 FF	Ereignis-Speicher mit den Zählerständen / Ereignissen der letzten 2 Stunden
6	81 81 C7 86 15 FF	Ereignis-Speicher mit wöchentlich aufgezeichneten Zählerständen/ Ereignissen (Aufzeichnung zum Tageswechsel von Sonntag zum Montag)
7	81 81 C7 86 16 FF	Ereignis-Speicher mit monatlich aufgezeichneten Zählerständen/ Ereignissen
8	81 81 C7 86 17 FF	Ereignis-Speicher mit jährlich aufgezeichneten Zählerständen/ Ereignissen
9	81 81 C7 86 18 NN	Ereignis-Speicher mit frei laufender RP und aufgezeichneten Zählerständen/ Ereignissen Falls RP gleich Null gesetzt ist, erfolgt die Ansteuerung durch das Aktionsschaltprogramm (siehe Tab. 44) Wertebereich für NN: 0x01 bis 0x7F
10	81 81 C7 86 18 FF	Einbauzählerstand

Tab. 75: Liste vordefinierter Kennzahlen zur Datensammler-Identifikation

Je nach Anwendungsfall können weitere Kennzahlen definiert werden; diese sind dann vom Betreiber entsprechend zu verwalten.

7.3.2.43 Datenstruktur zum Setzen / Lesen der Eigenschaften eines Datensammlers

Die Eigenschaften eines Datensammlers können via „GetProcParameter“ gelesen sowie per „SetProcParameter“ geschrieben werden. Der Zugriff ist nur dann möglich, wenn der zugehörige Sensor / Aktor aktiviert worden ist (siehe Bild 4 / Bild 5).

Das Kommando zum Lesen oder Schreiben muss an den zugehörigen Sensor / Aktor per Server-ID des Sensors / Aktors adressiert werden. Wird das Kommando an eine MUC-Controller adressiert, antwortet der MUC-Controller mit der Nachricht ‚AttentionResponse‘ und einem Fehler-Kode.

Der Aufzeichnungs-Prozess eines Datensammlers arbeitet wie folgt:

- Mit dem Empfang eines Wertes wird dem Wert ein Zeitstempel (der „Empfangszeitstempel“) durch den MUC zugeordnet. Es werden also Tupel aus Wert und Empfangszeitstempel gebildet.
- Der Datensammler legt periodisch oder ereignisgesteuert das jeweils zuletzt gebildete Tupel in der Aufzeichnung ab.

- Ist zum Zeitpunkt der nächsten Aufzeichnung kein neuer Wert empfangen worden, wird das bereits zuvor einmal abgelegte Tupel erneut aufgezeichnet.
- Als Konsequenz können in einem Datensammler Einträge auftreten, die jeweils mit unterschiedlichen Zeitstempeln für den Aufzeichnungszeitpunkt jedoch mit demselben Inhalt (Tupel aus Wert und Empfangszeitpunkt) bestehen.
- Eine Leitstelle kann diese Wiederholung durch Interpretation der Tupel aus Wert und Empfangszeitstempel erkennen.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums mit den Eigenschaften des Datensammlers	81 81 C7 86 20 FF	Nicht vorhanden	Mindestens ein Element gemäß Pos. 2
2	Kind-Element im Baum Eigenschaften des Datensammlers	81 81 C7 86 20 NN NN ⇔ Datensammlernr. NN = 01..FE	Nicht vorhanden	Je ein Element der Positionen 3 bis 8
3	Kind-Element im Baum Merkmal ‚Aufzeichnung aktiv / abgeschaltet‘	81 81 C7 86 21 FF	<ul style="list-style-type: none"> • Typ: Boolean • Inhalt: TRUE ⇔ aktiv 	Nicht vorhanden
4	Kind-Element im Baum Anzahl der Registrierperioden (bei Überschreiten wird rollierend aufgezeichnet)	81 81 C7 86 22 FF	<ul style="list-style-type: none"> • Typ: Unsigned16 • Inhalt: Anzahl 	Nicht vorhanden
5	Kind-Element im Baum Registrierperiode, anzugeben in Sekunden	81 81 C7 87 81 FF	<ul style="list-style-type: none"> • Typ: Unsigned32 • Inhalt: RP in Sekunden (bei 0 erfolgt die Aufzeichnung per Ereignissteuerung)	Nicht vorhanden
6	Kind-Element im Baum OBIS-Kennzahl zur Adressierung des Datensammlers	81 81 C7 8A 83 FF	<ul style="list-style-type: none"> • Typ: Octet String • Inhalt: Kennzahl 	Nicht vorhanden
7	Kind-Element im Baum Liste mit den aufzuzeichnenden per OBIS gekennzeichneten Messwerten	81 81 C7 8A 23 FF	<ul style="list-style-type: none"> • Nicht vorhanden 	Ein oder mehrere Elemente nach Pos. 7 dieser Tabelle
8	Kind-Element im Baum Listenelemente mit den aufzuzeichnenden per OBIS gekennzeichneten Messwerten	81 81 C7 8A 23 NN	<ul style="list-style-type: none"> • Octet String 	

Tab. 76: Datenstruktur mit Parametern zum Datensammler

7.3.2.44 MUC und EDL21-Zähler im EDL40-System

Grundsätzlich ist im EDL40-System zu gewährleisten, dass aus einem Datensammler Zählerstände mit zugehörigen Zeitstempeln, Metadaten und Signaturen („signierte Tupel“) geliefert werden, bei denen die Zeitstempel gemäß der EDL40-Systemarchitektur immer auf volle Viertelstunden gerastert sind. (Hier handelt es sich um die Zeitstempel, die unter der OBIS-Kennzahl 01 00 01 11 00 FF, siehe FNN-Lastenheft EDL, die Führungsgröße für signierte Zählerstände zu ‚+A‘ bilden).

Die vorstehend benannten signierten Tupel werden in einem Datensammler aufgezeichnet, wobei das zeitlich sortierende Kriterium im Datensammler der Betriebssekundenzähler¹⁵ des MUC ist. Der Zeitpunkt der Aufzeichnung wird durch die Führungsgröße (siehe oben, die Führungsgröße ist der mit dem signierten Tupel angelieferte Zeitstempel) bestimmt. Mit der Parametrierung des Datensammlers wird festgelegt, welches signierte Tupel aufzuzeichnen, welche Registrierperiode und welche Führungsgröße zu verwenden ist.

Auf die Datensammler kann nur per SML_ProfileList zugegriffen werden. Anfragen an den Datensammler mit ‚beginTime‘ / ‚endTime‘ beziehen sich auf den Betriebssekundenzähler des MUC.

Anfragen zur Auslesung des Datensammlers per SML_ProfileList-Request können wahlweise den Betriebssekundenzähler oder die gesetzliche Zeit in den Elementen ‚beginTime‘ und ‚endTime‘ benutzen. Als Bezug für die Bewertung von ‚beginTime‘ und ‚endTime‘ wird damit immer die Zeitbasis des MUC herangezogen. Wird ein Datensammler für signierte Tupel per Zeitbasis des MUC abgefragt, ist zu beachten, dass die Antwort auch den per MUC-Zeitbasis definierte Zeitraum enthalten und damit abweichend zu den EDL40-Zeitstempeln („Signatur-Delay“) sein kann.

Über das optionale Attribute ‚object_List‘ kann der Anfragende bei Bedarf die in der Antwort gewünschten Spalten explizit angeben und damit die Ausgabe der Spalte mit der gesetzlichen Zeit des MUC unterdrücken.

Der Zugriff auf die Datensammler erfolgt unter Verwendung des Server-ID des zugeordneten Sensors / Aktors (siehe Bild 6 / Bild 7).

Wegen der möglicherweise unterschiedlichen Zeitstempel bei verschiedenen signierten Meßgrößen (signierte Tupel) derselben Datenquelle wird empfohlen, je signierte Meßgröße genau einen Datensammler zu benutzen.

Als ‚valTime‘ (siehe ‚SML_GetProfileList.Res‘) ist der Betriebssekundenzähler (des MUC) zum Zeitpunkt der Aufzeichnung in den Lastgangsammler zu verwenden. Zusätzlich ist im Lastgang eine Spalte unter der OBIS-Kennzahl 01 00 00 09 0B 00 zu führen, in der die gesetzliche Zeit des MUC (Typ: SML_Time in der Variante local Timestamp) ergänzt wird.

¹⁵ Der Betriebssekundenzähler des MUC wurde gewählt, da dieser als einzige zeitliche Größe im System aus MUC und angeschlossenen Zählern eine gesichert streng monoton wachsende Eigenschaft bietet.

- Eine sinnvolle Zuordnung der einzelnen Inhalte kann damit wie folgt dargestellt werden, wobei andere Varianten (beispielsweise die Aufzeichnung aller Informationen in einem einzigen Datensammler¹⁶) ebenfalls per Parametrierung gewählt werden können:

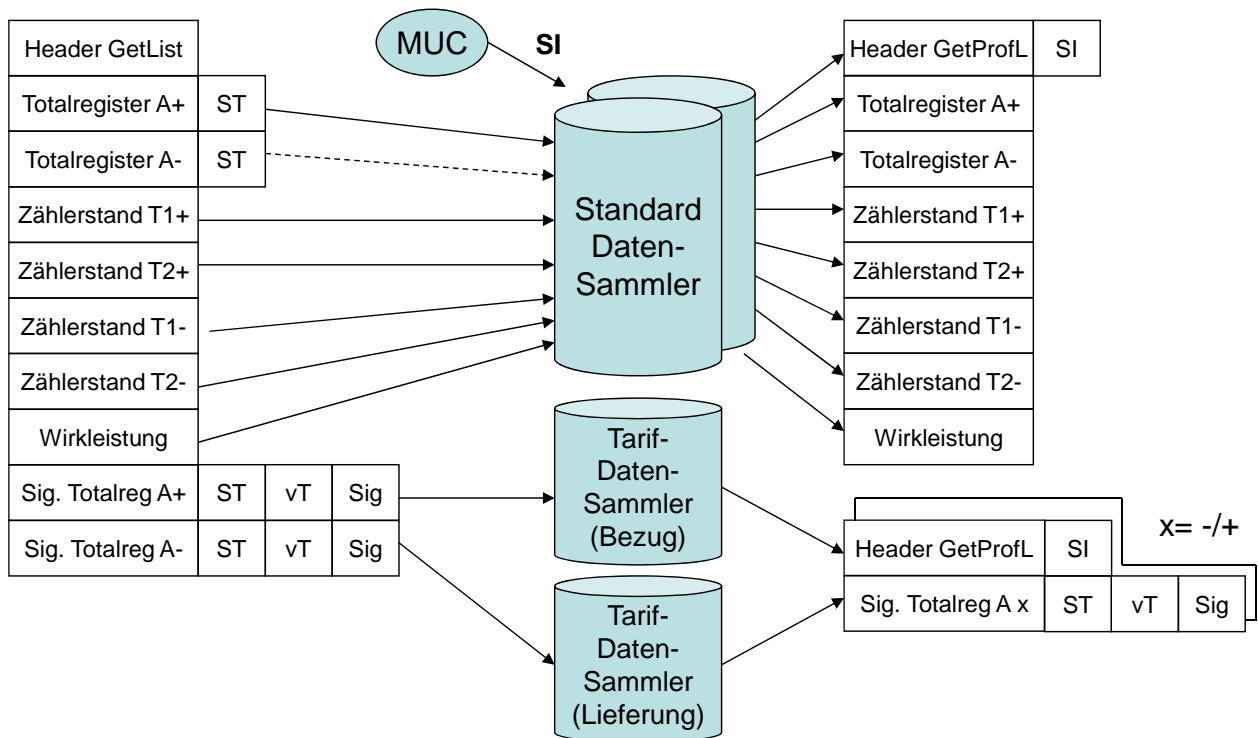


Bild 27: Datensammler (Beispiel-Zuordnung von EDL40-Informationen)

Zur Abbildung dieser Struktur ist SML wie folgt zu verwenden (Strecke EDL21-Zähler an MUC):

Ausgehend vom EDL21-Zähler im EDL40-Modus wird ein signiertes Tupel als ‚SML_ListEntry‘ angeliefert. Der Betriebssekundenindex des EDL21-Zählers wird, auch im EDL40-Modus, im Element ‚actSensorTime‘ als Teil der ‚SML_GetList.Res‘ übertragen. Das Element ‚SML_ListEntry‘ ist dann Teil der ‚SML_GetList.Res‘ als eines von mehreren Unterelementen im ‚SML_List‘.

SML_GetList.Res	::= SEQUENCE		
{			
clientId	Octet String	OPTIONAL,	
serverId	Octet String,		
listName	Octet String	OPTIONAL,	
actSensorTime	SML_Time	OPTIONAL,	← Betriebssekundenindex vom Zähler zum Zeitpunkt der SML-Nachrichten-Ausgabe,
valList	SML_List,		← Werteliste, bei der als ein Element das signierte Tupel mitgesendet wird

¹⁶ Es wird für diese Art der Parametrierung darauf hingewiesen, dass dann die Zählerstände zu den Totalregistern abweichend von den Zählerständen in den signierten Tupeln sein können. Die Ursache liegt in der für die Signaturberechnung in der Datenquelle benötigten Latenzzeit.


```

listSignature      SML_Signature      OPTIONAL,
actGatewayTime    SML_Time          OPTIONAL
}

```

Jedes signierte Tupel, aufgebaut als ‚SML_ListEntry‘, liefert folgende Detail-Informationen:

```

SML_ListEntry      ::= SEQUENCE
{
  objName           Octet String,           ← OBIS zum signierten Tupel,
  status            SML_Status              OPTIONAL,   ← Status vom Zähler,
  valTime           SML_Time                OPTIONAL,   ← Zeitstempel vom Zähler
                                                           zum Zeitpunkt der
                                                           Messwert-Bildung,
  unit              SML_Unit                OPTIONAL,   ← Zählerstand (Einheit),
  scaler            Integer8                OPTIONAL,   ← Zählerstand (Faktor),
  value             SML_SimpleValue,        ← Zählerstand (Wert),
  valueSignature    SML_Signature           OPTIONAL   ← Signatur zum Zählerstand
}

```

Zur Abbildung dieser Struktur ist SML wie folgt zu verwenden (Strecke MUC an Zentrale):

Jedes zu einem Zeitpunkt aufgezeichnete signierte Tupel wird per ‚SML_PeriodEntry‘ als Teil einer ‚SML_GetProfileList.Res‘ vom MUC an eine Zentrale gesendet.

```

SML_GetProfileList.Res ::= SEQUENCE
{
  serverId          Octet String,
  actTime           SML_Time,               ← Betriebssekundenindex vom
                                                           MUC zum Zeitpunkt der
                                                           SML-Nachrichten-Ausgabe,
  regPeriod         Unsigned32,             ← Registrierperiode laut
                                                           MUC-Parametrierung,
  parameterTreePath SML_TreePath,          ← OBIS-Kennzahl laut MUC zum
                                                           Datensammler,
  valTime           SML_Time,               ← Betriebssekundenzähler des
                                                           MUC zum Zeitpunkt der
                                                           Aufzeichnung des Eintrags,
  status            Unsigned64,             ← Status des MUC zum Zeitpunkt
                                                           der Aufzeichnung des Eintrags,
  period_List       List_of_SML_PeriodEntry, ← Liste jener Werte, die zum
                                                           Aufzeichnungszeitpunkt durch
                                                           den MUC gespeichert wurden,
  rawdata           Octet String            OPTIONAL,
  periodSignature    SML_Signature           OPTIONAL
}

```

```

SML_PeriodEntry ::= SEQUENCE
{
  objName      Octet String,
  unit         SML_Unit,
  scaler       Integer8,
  value        SML_Value,

  valueSignature SML_Signature OPTIONAL
}

```

← OBIS-Kennzahl zum Meßwert,
 ← Zählerstand (Einheit),
 ← Zählerstand (Faktor),
 ← Zählerstand (Wert), Status
 und Zeitstempel des Zählers
 zum Zeitpunkt der Messwert-
 Bildung,
 ← Signatur zum Zählerstand

Der Wert zum Zählerstand wird in Verbindung mit dem zugehörigen Zeitstempel als ‚SML_TimestampedValue‘ wie folgt kodiert:

```

SML_Value ::= IMPLICIT CHOICE
{
  boolean-Value      boolean,
  byte-List          Octet String,
  8-Bit-Integer      Integer8,
  16-Bit-Integer     Integer16,
  32-Bit-Integer     Integer32,
  64-Bit-Integer     Integer64,
  8-Bit-Unsigned     Unsigned8,
  16-Bit-Unsigned   Unsigned16,
  32-Bit-Unsigned   Unsigned32,
  64-Bit-Unsigned   Unsigned64
  smlList            SML_ListType
}

```

← Zählerstand (Wert), Status
 und Zeitstempel des Zählers
 zum Zeitpunkt der Messwert-
 Bildung,

```

SML_ListType ::= CHOICE
{
  smlTime          [0x01] SML_Time
  smlTimestampedValue [0x02] SML_TimestampedValue
}

```

← Zählerstand (Wert), Status
 und Zeitstempel des Zählers
 zum Zeitpunkt der Messwert-
 Bildung,

```

SML_TimestampedValue ::= SEQUENCE
{
  smlTime          SML_Time
}

```

← Zeitstempel des Zählers,

```

status          SML_Status          ← Status des Zählers,
simpleValue     SML_SimpleValue     ← Zählerstand (Wert)
}
    
```

7.3.2.45 Datenstruktur zum Setzen / Lesen der Eigenschaften von Schwellwerten

Die Datenstruktur wird definiert, sobald die Rahmenbedingungen zur Handhabung entsprechender Tarifmodelle / Abschaltanlagen / Aufgaben näher bekannt sein werden.

7.3.2.46 Datenstruktur zu Alarm-Meldungen bei Überschreitung von Schwellwerten eines internen Datensammlers

Die Datenstruktur wird definiert, sobald die Rahmenbedingungen zur Handhabung entsprechender Tarifmodelle / Abschaltanlagen / Aufgaben näher bekannt sein werden.

7.3.2.47 Datenstruktur zum Schalten von Relaisausgängen (Relais-Attribute)

Zur Ansteuerung von Relaisausgängen wird ein Konzept aus Relaisausgang und diesen ansteuernde Quelle verwendet.

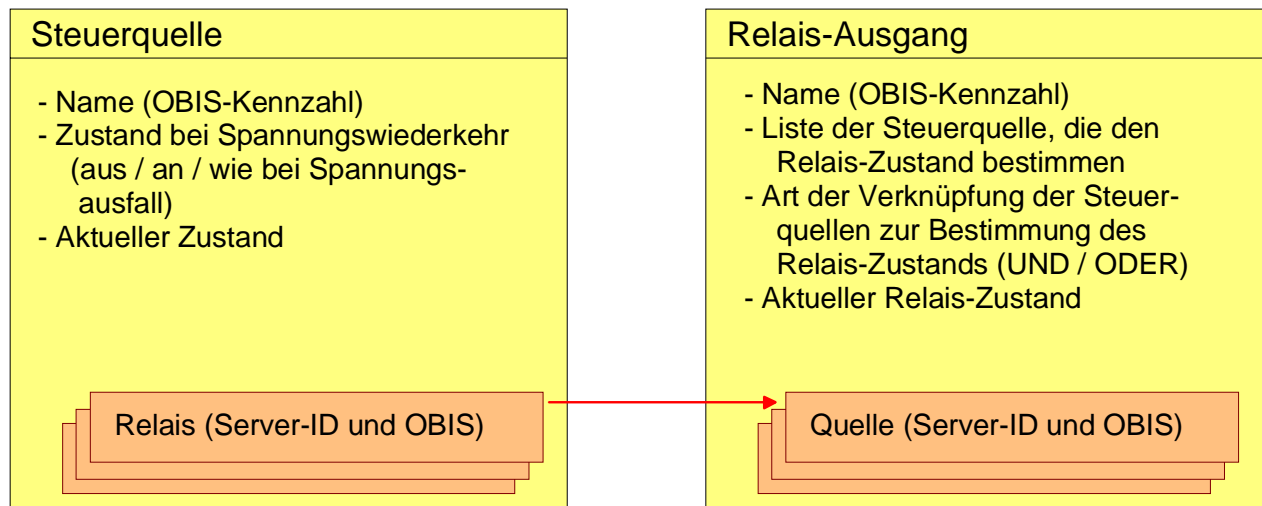


Bild 28: Konzept von Relaisausgang und Steuerquelle

- Zur Parametrierung des Relaisausgangs ist folgende Datenstruktur zu verwenden:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element der Datenstruktur zum Schalten eines Relaisausgangs	81 81 C7 8B 00 FF	Nicht vorhanden	Mindestens ein oder mehrere Element(e) nach Positionen 2 dieser Tabelle
2	Kind-Element im Baum. Identifikation zum Ausgang	81 81 C7 8B 00 NN Wertebereich: 0x01 – 0x7F	Nicht vorhanden	Je ein Element nach Pos. 3 - 5
3	Kind-Element im Baum. Verknüpfungsart	81 81 C7 8B 01 FF	Unsigned 8 0 ⇔ UND 1 ⇔ ODER alle anderen Werte sind reserviert	Nicht vorhanden

4	Kind-Element im Baum. aktueller Relaiszustand (Diese Eigenschaft kann nur gelesen werden!)	81 81 C7 8B 02 FF	Boolean	Bei ‚true‘ ist der Ausgang aktiviert
5	Kind-Element im Baum. Liste der Steuerquellen, die auf das Relais einwirken	81 81 C7 8B 03 FF	Nicht vorhanden	Ein oder mehrere Elemente nach Pos. 6
6	Kind-Element im Baum. Listenelemente zu den Steuerquellen, die auf das Relais einwirken	81 81 C7 8B 03 NN Wertebereich: 0x01 – 0x7F	Nicht vorhanden	Genau ein Element nach Pos. 7 und ein weiteres nach Pos. 8 dieser Tabelle
7	Kind-Element im Baum. Server-ID der Steuerquelle	81 81 C7 82 04 FF	Octet String	Nicht vorhanden
8	Kind-Element im Baum. Identifikation der Steuerquelle	81 81 C7 8B 04 FF	Octet String (siehe Pos. 2 der Tab. 78)	Nicht vorhanden

Tab. 77: Datenstruktur mit Parametern zu Relais-Ausgängen

7.3.2.48 Datenstruktur zum Schalten von Relaisausgängen (Steuerungs-Attribute)

Zur Parametrierung der Ansteuerung ist folgende Datenstruktur zu verwenden:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element der Datenstruktur zum Schalten eines Relaisausgangs	81 81 C7 8B 10 FF	Nicht vorhanden	Mindestens ein oder mehrere Element(e) nach Positionen 2 dieser Tabelle
2	Kind-Element im Baum. Identifikation zur Steuerquelle	81 81 C7 8B 10 NN	Nicht vorhanden	Genau je ein Element nach Pos. 3 bis 7
3	Kind-Element im Baum. Zustand bei Wiederkehr der Betriebsbereitschaft	81 81 C7 8B 11 FF	Unsigned 8 0 ⇔ AUS 1 ⇔ AN 2 ⇔ Wie bei Verlust der Betriebs- bereitschaft alle anderen Werte sind reserviert	Nicht vorhanden
4	Kind-Element im Baum. aktueller Schaltzustand	81 81 C7 8B 12 FF	Boolean Bei ‚true‘ ist der Ausgang aktiviert	

Tab. 78: Datenstruktur mit Parametern zu Relais-Steuerquellen

7.3.2.49 Datenstruktur zur Beschreibung von Parametrier-Schablonen

Zur vereinfachten Parametrierung von MUC-Controllern können Parametrier-Schablonen („Templates“) benutzt werden. Dabei beschreibt eine Parametrier-Schablone jene Eigenschaften, die beim Anlegen

eines neuen Sensors für die Informationen Datenspiegel, Datensammler und Push-Parameter anzuwenden sind.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element der Datenstruktur mit allen Parametrier-Schablonen	81 81 C7 86 24 FF	Nicht vorhanden	Mindestens ein oder mehrere Element(e) nach Position dieser Tabelle und genau ein Element nach Pos. 7
2	Kind-Element im Baum. Listeneintrag für eine Parametrier-Schablone	81 81 C7 86 24 NN 01 <= NN <= FE	Nicht vorhanden	Jeweils genau ein Element nach Pos. 3 bis 6 dieser Tabelle
3	Kind-Element im Baum. Knoten-Element im Sinne des Wurzel-Elements einer Parametrier-Schablone zum Datenspiegel (siehe Tab. 74)	81 81 C7 86 00 FF	Nicht vorhanden	Siehe Tab. 74
4	Kind-Element im Baum. Knoten-Element im Sinne des Wurzel-Elements einer Parametrier-Schablone zum Datensammler (siehe Tab. 76)	81 81 C7 86 20 FF	Nicht vorhanden	Siehe Tab. 76
5	Kind-Element im Baum. Knoten-Element im Sinne des Wurzel-Elements einer Parametrier-Schablone zu den Push-Parametern (siehe Tab. 31)	81 81 C7 8A 01 FF	Nicht vorhanden	Siehe Tab. 31
6	Kind-Element im Baum. Verweis auf Aktions-schaltprogramm, dem der Datenspiegel bei Benutzung der Parametrier-Schablone zugeordnet werden soll	81 81 C7 86 25 FF	Octet String Leer bedeutet: Keine Zuweisung	Nicht vorhanden
7	Kind-Element im Baum. Liste der Sensoren / Aktoren, auf die die Parametrier-Schablone angewendet werden soll	81 81 C7 86 26 FF	Nicht vorhanden	Mindestens ein Element nach Pos. 8 dieser Tabelle
8	Kind-Element im Baum. Listenelement zur Beschreibung der Beziehung Parametrier-Schablone ↔ Sensor / Aktor	81 81 C7 86 27 NN Wertebereich: 01..FE	Nicht vorhanden	Je ein Element nach Pos. 9 bis ... dieser Tabelle
9	Kind-Element im Baum. Zu benutzende Parametrier-Schablone	81 81 C7 86 28 FF	Octet String Kennzahl gemäß Pos. 2 dieser Tabelle	Nicht vorhanden

10	Kind-Element im Baum. Geräteidentifikation, an Hand derer zugehörige Sensoren / Aktoren erkannt werden	81 81 C7 86 29 FF	Octet String Sensor / Aktor spezifisches Identifikationsmerkmal (bspw. die Geräte-Identifikation / Server-ID) Ein leerer Octet String ist als generelle Wildcard anzusehen. Bei Geräten nach OMS: - Ident-Nummer, anzugeben in 4 Bytes mit LSB zu Beginn ¹⁷ Bei EDL-Zählern: Server-ID ¹⁸ , Kennzahl 0x010000009FF	Nicht vorhanden
11	Kind-Element im Baum. Filter zur Geräteidentifikation	81 81 C7 86 2A FF	Octet String Ein gesetztes Bit im Filter-Octet-String bedeutet: Das korrespondierende Bit im Vergleichsstring ¹⁹ ist auszuwerten.	Nicht vorhanden
12	Kind-Element im Baum. Medium, an Hand derer zugehörige Sensoren / Aktoren erkannt werden	81 81 C7 86 2b FF	Octet String Sensor / Aktor spezifisches Medium Ein leerer Octet String ist als generelle Wildcard anzusehen. Bei Geräten nach OMS: - Medium 1 Byte gemäß Tabelle „Device Type Identification“ der EN 13757-3 ¹⁷ Bei EDL-Zählern: - 1 Byte gemäß OBIS Value Group A	Nicht vorhanden

¹⁷ Kodierung gemäß EN 13757-3 wobei Wildcards nicht zulässig sind.

¹⁸ Wildcards sind wie folgt zu Kodieren:

¹⁹ Als Vergleichsstring ist die vom zu prüfenden Sensor / Aktor kommende Byte-Kette zu benutzen. Ist der Vergleichsstring länger als die mit dem Filter definierte Byte-Kette, sind die überhängenden Bytes im Vergleichsstring zu ignorieren. Ist der Vergleichsstring kürzer als die mit dem Filter definierte Byte-Kette, gilt als Ergebnis immer ein „Ablehnen“.

13	Kind-Element im Baum. Filter zum Medium	81 81 C7 86 2C FF	Octet String Zu Bedeutung und Auswertung siehe Pos. 11 dieser Tabelle	Nicht vorhanden
14	Kind-Element im Baum. Herstellerkennung, an Hand derer zugehörige Sensoren / Aktoren erkannt werden	81 81 C7 86 2d FF	Octet String Hersteller-Kennung gemäß FLAG Ein leerer Octet String ist als generelle Wildcard anzusehen. Bei Geräten nach OMS: - Herstellerkennung ⇔ EN 13757-3, 2 Bytes ¹⁷ Bei EDL-Zählern: - 3 Byte in ASCII gemäß FLAG Association	Nicht vorhanden
15	Kind-Element im Baum. Filter zur Herstellerkennung	81 81 C7 86 2E FF	Octet String Zu Bedeutung und Auswertung siehe Pos. 11 dieser Tabelle	Nicht vorhanden
16	Kind-Element im Baum. Versionskennung, an Hand derer zugehörige Sensoren / Aktoren erkannt werden	81 81 C7 86 2F FF	Octet String Versionskennung Bei Geräten nach OMS: - 1 Byte ¹⁷ Bei EDL-Zählern: - Versionsstring gemäß FNN- Lastenheft, Kennzahl 0x0100000200FF	Nicht vorhanden
17	Kind-Element im Baum. Filter zur Versionskennung	81 81 C7 86 30 FF	Octet String Zu Bedeutung und Auswertung siehe Pos. 11 dieser Tabelle	Nicht vorhanden
18	Kind-Element im Baum. Geräteklasse, an Hand derer zugehörige Sensoren / Aktoren erkannt werden	81 81 C7 86 31 FF	Octet String Geräteklasse Bei Geräten nach OMS: - nicht benutzt Bei EDL-Zählern: - Siehe FNN- Lastenheft, Kennzahl 0x8181C78202FF	Nicht vorhanden
19	Kind-Element im Baum.	81 81 C7 86 32 FF	Octet String	Nicht vorhanden

Filter zur Geräteklasse

Zu Bedeutung und
Auswertung siehe
Pos. 11 dieser
Tabelle**Tab. 79: Datenstruktur mit Liste aller Parametrier-Schablonen**

Die Datenstruktur kann, entsprechende Zugriffsrechte vorausgesetzt, gelesen und geschrieben werden.

Die erste Übereinstimmung, beginnend vom Beginn der Liste, zwischen einem neu erkannten Sensor / Aktor und dessen Identifikation in der Datenstruktur führt zum Anlegen eines Datenspiegels unter Benutzung der zugehörigen Parametrier-Schablone.

7.3.2.50 Datenstruktur zur Beschreibung von Parametrier-Schablonen

Zur vereinfachten Parametrierung von MUC-Controllern können Parametrier-Schablonen („Templates“) benutzt werden. Dabei beschreibt eine Parametrier-Schablone jene Eigenschaften, die beim Anlegen eines neuen Sensors für die Informationen Datenspiegel, Datensammler und Push-Parameter anzuwenden sind.

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element der Datenstruktur zur manuellen Anwendung Parametrier-Schablonen	81 81 C7 86 33 FF	Nicht vorhanden	Je ein Element nach Position 2 bis 3 dieser Tabelle
2	Kind-Element im Baum. Zu benutzende Parametrier-Schablone	81 81 C7 86 34 FF	Octet String Kennzahl gemäß Tab. 79	Nicht vorhanden
3	Kind-Element im Baum. Server-ID, auf den die Parametrier-Schablone anzuwenden ist	81 81 C7 86 35 FF	Octet String Server-ID	Nicht vorhanden

Tab. 80: Kommando zur manuellen Anwendung von Parametrier-Schablonen

Wird das Kommando angewendet, sind folgende Aktionen durchzuführen:

- Der zuvor diesem Sensor / Aktor zugeordnete Datenspiegel wird durch einen neuen Datenspiegel mit den durch die Parametrier-Schablone definierten Eigenschaften ersetzt; die Datensammler bleiben erhalten.

7.3.2.51 Datenstruktur zum Tunneln von Fremdprotokollen (3rd-Party-Protocol-Request)

Zum Senden von Datenpaketen fremder Protokolle wird folgende Datenstruktur definiert, wobei dieser Auftrag per ‚SML_SetProcParameter‘ übertragen werden muss:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Tunneln von Fremdprotokollen (Senden)	81 80 81 01 01 FF	Unsigned8 Wertebereich: Protokoll-Typ, siehe Tab. 84	Jeweils 1 Kind-Element nach Pos 2 – 5

2	Kind-Element im Baum Sitzungs-Identifikation	81 82 81 51 01 FF	Octet String Wertebereich: Applikations- spezifisch
3	Kind-Element im Baum Das Datenpaket im Tunnel	81 80 81 51 01 FF	Octet String Wertebereich: Ein Datenpaket mit maximal 4096 Bytes
4	Kind-Element im Baum Die Zieladresse, wo das Datenpaket hin zu senden ist	81 80 81 51 03 FF	Octet String Wertebereich: Eine für das zu tunnelnde Protokoll spezifische Zieladresse mit maximal 512 Bytes
5	Kind-Element im Baum Ein vom Auftraggeber frei zu wählender Identifier, über den die Antwort auf das getunnelte Paket abgeholt werden kann. Fehlt diese Angabe, wird das zu tunnelnde Paket im Sinne von „Push“, also ohne die Erwartung einer Antwort, weitergeleitet.	81 80 81 51 02 FF	Unsigned64

Tab. 81: Kommando zum Tunneln von Fremdprotokollen in Senderichtung

Zulässige Antwortcodes sind:

- OK (Datenpaket wurde von Empfänger weiter gesendet);
- FAIL (Datenpaket konnte nicht weiter gesendet werden);
(unter ‚FAIL‘ werden alle mit SML definierten Fehlercodes zusammengefasst)

7.3.2.52 Datenstruktur zum Tunneln von Fremdprotokollen (3rd-Party-Protocol-Response)

Zum Empfang von Datenpaketen fremder Protokolle wird folgende Datenstruktur definiert, wobei dieser Auftrag per ‚SML_SetProcParameter‘ übertragen werden muss und der Empfang über die Detail-Informationen einer ‚SML_Attention‘ geantwortet wird:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Tunneln von Fremdprotokollen (Empfang)	81 80 81 01 02 FF	Nicht verwendet	Jeweils 1 Kind- Element nach Pos 2 – 5
2	Kind-Element im Baum Sitzungs-Identifikation	81 82 81 51 01 FF	Octet String Wertebereich: Applikations- spezifisch	
3	Kind-Element im Baum Der vom Auftraggeber beim Senden frei gewählte Identifier,	81 80 81 51 02 FF	Unsigned64	

	über den die Antwort auf ein zuvor per Tunnel gesendetes Paket abgeholt werden kann. Fehlt diese Angabe, wird das zu nächste verfügbare Paket angefordert.		
4	Kind-Element im Baum Merkmal ‚diese Anfrage ist die letzte Anfrage zu dem Tunnelvorgang‘	81 80 81 51 05 FF	Boolean Per ‚true‘ teilt der Auftraggeber mit, dass er zu diesem unter Pos. 3 identifizierten Tunnelvorgang keine weiteren Anfragen für die Empfangsrichtung stellen wird
5	Kind-Element im Baum Anzahl der in der Antwort maximal zulässig zu übertragenden Bytes	81 80 81 51 06 FF	Unsigned64

Tab. 82: Kommando zum Tunneln von Fremdprotokollen in Empfangsrichtung (Polling-Request)

Zulässige Antwortkodes sind:

- OK (Ein Datenpaket wurde empfangen)
(das Datenpaket wird per ‚attentionDetails‘ bereitgestellt).
- FAIL
(unter ‚FAIL‘ werden alle mit SML definierten Fehlerkodes zusammengefasst)
- FAIL_NoData
(es stehen keine empfangene Daten zur Verfügung)
- FAIL_SessionIdentifier
Die angegebene Sitzungs-Identifikation ist unpassend oder es besteht keine Sitzung.

Per ‚attentionDetails‘ zu liefernden Informationen:

Pos.	Merkmal / Eigenschaft	Kennzahl	Typ / Inhalt	Listenelement
1	Wurzel-Element des Baums zum Tunneln von Fremdprotokollen (Empfangen)	81 80 81 01 02 FF	Unsigned8 Wertebereich: Protokoll-Typ, siehe Tab. 84	Jeweils 1 Kind-Element nach Pos 2 – 4
2	Kind-Element im Baum Das Datenpaket im Tunnel	81 80 81 51 01 FF	Octet String Wertebereich: Ein Datenpaket mit maximal 4096 Bytes	
3	Kind-Element im Baum Die Quelladresse, von der das	81 80 81 51 04 FF	Octet String Wertebereich:	

	Datenpaket empfangen wurde.		Eine für das zu tunnelnde Protokoll spezifische Quelladresse mit maximal 512 Bytes
4	Kind-Element im Baum Der vom Auftraggeber beim Senden frei gewählte Identifier, über den die Antwort auf ein zuvor per Tunnel gesendetes Paket zugeordnet werden kann. Fehlt diese Angabe, wurde das nächste verfügbare Paket ausgeliefert, dem keine Anfrage zugeordnet werden kann.	81 80 81 51 02 FF	Unsigned64
5	Kind-Element im Baum Merkmal ‚zu diesem Tunnelvorgang sind weitere Datenpakete verfügbar‘.	81 80 81 51 07 FF	Boolean Per ‚true‘ wird mitgeteilt, dass zu diesem Tunnelvorgang weitere Datenpakete verfügbar sind.

Tab. 83: Kommando zum Tunneln von Fremdprotokollen in Empfangsrichtung (Polling-Response)

7.4 Binär-Tunnel für andere Protokolle

Um die Möglichkeit zu geben, unabhängig von SML Protokolle Dritter von einem MUC an ein Backend richten zu können (oder von einem Backend über einen MUC an ein dahinter liegendes ‚Legacy-Device‘), werden zwei Datenstrukturen (siehe Kapitel 7.3.2.51 ff.) definiert.

Mit der ersten der beiden Datenstrukturen können Datenpakete fremder Protokolle versendet werden.

Die zweite der beiden Datenstrukturen erlaubt das Empfangen von Datenpaketen fremder Protokolle, die entweder im Sinne einer Antwort oder spontan eingetroffen sind. Der Empfang dieser Datenpakete erfolgt damit aus Sicht des Auftraggebers (MUC oder Backend) immer im Sinne von ‚Polling‘.

Durch die Aufteilung der Vorgänge ‚Auftrags-Datenpaket zum Fremdprotokoll senden‘ und ‚Antwort-Datenpaket zum Fremdprotokoll abholen per Polling‘ hat das Timingverhalten von Fremdprotokollen keine Auswirkung auf die anderen Request-Response-Prozesse in SML und / oder IP-Telemetrie.

Folgende Fremdprotokolle sind zulässig:

Pos.	Bedeutung	Kodierung im ersten Byte	Art der Fremd-Protokoll-Adressen
1	Reserviert für künftige Erweiterungen	0x00	
2	http	0x01	Adressen werden als URL angegeben
3	M-Bus Applikationsprotokoll	0x02	Adressen werden nach Tab. 7 kodiert
4	Reserviert für künftige Erweiterungen	0x03	
5	Reserviert für künftige Erweiterungen	...	

6	Reserviert für künftige Erweiterungen	0x7F	
7	Applikations spezifisch	0x80	Applikations spezifisch
8	Applikations spezifisch	...	Applikations spezifisch
9	Applikations spezifisch	0xFF	Applikations spezifisch

Tab. 84: Liste vordefinierter Identifier zu Fremdprotokollen

7.5 Aktionsschaltprogramm

7.5.1 Grundsätzliche Festlegungen zum Funktionsverhalten

Zum grundsätzlichen Verhalten von Aktionsschaltprogrammen wird definiert:

- Das Aktionsschaltprogramm wird mit Erreichen der Betriebsbereitschaft gestartet.
- Zeitbezogene Aktionen werden nur ausgeführt, wenn die Zeitbasis sicher / gültig ist.
- Mit Erreichen der Betriebsbereitschaft und dem Status „Zeitbasis sicher“ ist der Zustand einzustellen, der als Wirkung aller zeitbezogenen Ereignisse des aktiven Regelsatzes zu dem Zeitpunkt des Erreichens der Betriebsbereitschaft gefordert ist.
- Mit Erreichen der Betriebsbereitschaft und dem Status „Zeitbasis unsicher“ ist das Ereignis „Zeitbasis unsicher“ auszulösen.
- Mit dem Ereignis „Erreichen der Betriebsbereitschaft“ können Zustände (beispielsweise der Zustand eines Eingangskontakts) abgefragt werden, um die aus diesem Zustand abgeleiteten Aktionen bei Erreichen der Betriebsbereitschaft wieder herzustellen.
- Beschreibt eine Regel ein Ereignis, wird die zugehörige Aktionsliste genau dann ausgeführt, wenn das Ereignis stattfindet. Findet ein Ereignis mehrfach während des Gültigkeitszeitraums eines Regelsatzes statt, wird die zugehörige Aktionsliste auch mehrfach ausgeführt.
- Ereignisse, die in einem Regelsatz beschrieben sind und die bereits vor dem Gültigkeitszeitraum eines Regelsatzes ausgelöst worden sind (z.B. das Erreichen der Betriebsbereitschaft) führen mit Beginn der Gültigkeit des Regelsatzes nicht zur Auslösung der zugehörigen Aktionsliste.
- Zeitbezogene wie alle anderen Ereignisse, die während des Ausfalls der Betriebsbereitschaft stattgefunden haben werden mit Erreichen der Betriebsbereitschaft nicht nachgeholt (Ausnahme: Ereignis „Zeitbasis unsicher“, siehe oben).

Sobald mehrer Saisonprogramme in einem Aktionsschaltprogramm definiert sind, sind folgende Abhängigkeiten zu beachten:

- Mit Abschalten eines Saisonprogrammes werden alle in dem Saisonprogramm enthaltenen Tagesprogramme beendet. Ein Tagesprogramm ist nur dann aktiv, wenn es Teil eines aktiven Saisonprogrammes ist.
- Mit Abschalten eines Saisonprogrammes werden alle von dem Saisonprogramm gestarteten periodischen Vorgänge beendet.

- Singuläre Aktionen, die mit dem Ereignis Saisonende aktiviert werden, sind im Sinne der letzten Aktion einer Saison auszuführen. Sie werden nicht vorzeitig abgebrochen (ein beispielsweise über 5 Minuten auszugebender Impuls wird vollständig ausgegeben).
- Zeitbasierte Regelergebnisse, die über das Tages- oder Saisonende hinaus andauern, werden mit Abschalten des Saisonprogrammes abgebrochen.

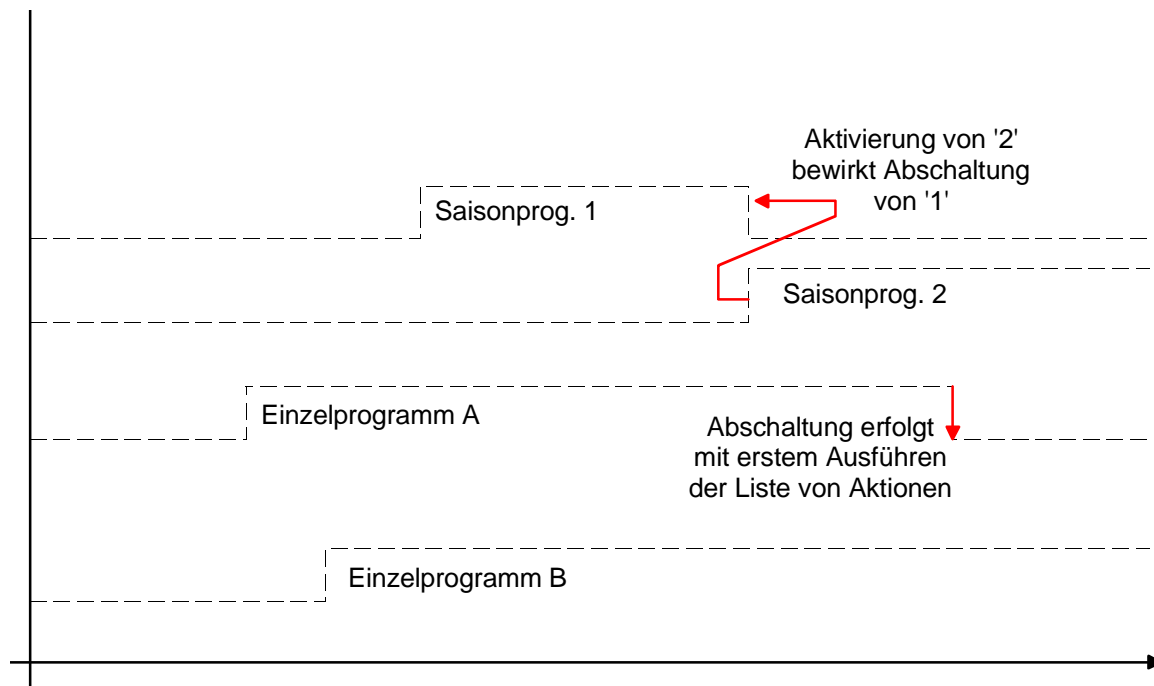


Bild 29: Verhalten bei Definition mehrerer Saisonprogramme.

7.6 Betriebslogbuch

Der MUC-Controller enthält ein Logbuch zur Erfassung von Betriebsereignissen. Dieses arbeitet nach folgendem Prinzip:

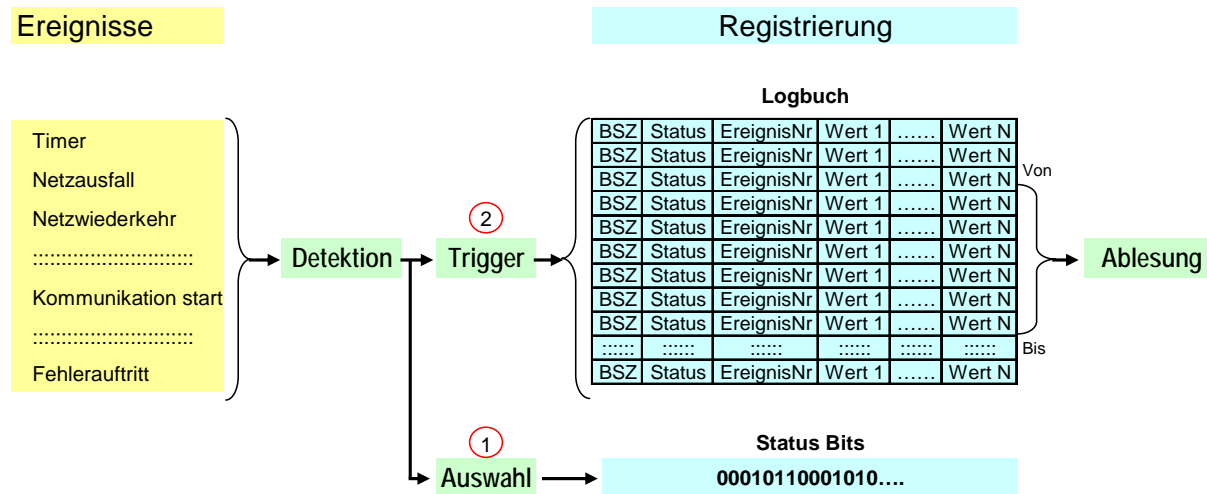


Bild 30: MUC-Controller, Arbeitsweise zum Logbuch für Betriebsereignisse

Dieses Logbuch wird per SML-Nachricht „GetProfileListResponse“ transportiert. Zur Anfrage wird eine Kennzahlen-Liste verwendet, deren Inhalt die konkret zu liefernden Logbucheinträge bezeichnet. Als Kennzahlen für diese Liste sind nur Kennzahlen gemäß folgender Tabelle zulässig.

Jeder Logbuch-Eintrag erzeugt einen Eintrag vom Typ „GetProfileListResponse“. Das Element ‚valTime‘ erhält dabei den Zeitpunkt des Logbuch-Eintrags. In ‚status‘ wird das zum Zeitpunkt des Logbucheintrags geltende Statuswort eingetragen.

Jedes Element eines Logbuch-Eintrags wird per „SML_PeriodEntry“ übertragen.

Liste der im Betriebslogbuch (MUC-C) zu registrierenden Eigenschaften

Pos.	Eigenschaft	Kennzahl	Datentyp	Einheit	MUC-C-Variante
1	Adressiert das Betriebslogbuch	81 81 C7 89 E1 FF	SML_TreePath	-	Alle
2	Betriebssekundenzähler	Nicht benötigt	Unsigned32	-	Alle
3	Statuswort	Nicht benötigt	Unsigned64	-	Alle
4	Ereignis	81 81 C7 89 E2 FF	Unsigned32	255	Alle
5	Peer Adresse ²⁰	81 81 00 00 00 FF	Octet String	255	Alle
6	aktuelle Feldstärke	81 04 2b 07 00 00	Integer16	254	GSM / GPRS
7	aktuelle Zelleninformation	81 04 1A 07 00 00	Unsigned16	255	GSM / GPRS
8	aktueller Location- oder Areacode	81 04 17 07 00 00	Unsigned16	255	GSM / GPRS
9	aktueller Provider-Identifizier	81 04 0D 06 00 00	Unsigned32	255	GSM / GPRS
10	Zeitsynchronisation per NTP-Zugriff erfolgt (Zeitangabe in UTC, die per NTP geliefert worden ist)	01 00 00 09 0B 00	SML_Time	UTC	Alle
11	1. Ereignis spezifische Zusatzinformation	Siehe Tab. 86	Siehe Tab. 86	...	Siehe Tab. 86
12
13	N. Ereignis spezifische Zusatzinformation	Siehe Tab. 86	Siehe Tab. 86	...	Siehe Tab. 86

Tab. 85: Betriebslogbuch, Liste der je Ereignis immer vorhandenen Zusatzinformationen.

Pos.	Spalte	Eigenschaft	Kennzahl	Datentyp	Einheit	MUC-C-Variante
1	Nächste freie Ereignis spez. Info	PLC-Signalqualität	81 04 0E 06 05 FF	Integer16		PLC
2	Nächste freie Ereignis spez. Info	Feldstärke Störpegel	81 06 00 02 04 FF	Integer16	dbm	Wireless M-Bus
3	Nächste freie Ereignis spez. Info	Push Vorgang	81 81 C7 8A 01 FF	Unsigned8 Nummer des Vorgangs		Alle

²⁰ Als ‚Peer Adresse‘ wird jeweils die OBIS-T-Kennzahl der Ereignisquelle abgelegt.

4	Nächste freie Ereignis spez. Info	Push Target	81 47 17 07 00 FF	Octet String	Alle
5	Nächste freie Ereignis spez. Info	Push Vorgang	TODO	Octet String Server-ID zum Push-Vorgang	Alle
6	Nächste freie Ereignis spez. Info	PLC-Network-ID	81 04 0E 06 04 FF	Octet String	PLC
7	Nächste freie Ereignis spez. Info	Utility-ID	81 04 18 07 01 FF	Octet String	PLC
8	Nächste freie Ereignis spez. Info	Datei- Übertragungs- Fortschritt in %	81 04 0E 07 04 FF	Unsigned32	PLC
9	TODO	Attribute zum Datei-Transfer			
10	Nächste freie Ereignis spez. Info	Adresse zum IP- T-Master	81 49 17 07 00 00	Bei IPv4: Unsigned32 Bei IPv6: Octet String	Alle
11	Nächste freie Ereignis spez. Info	Port zum IP-T- Master	81 49 1A 07 00 00	Unsigned16	Alle
12	Nächste freie Ereignis spez. Info	Adresse zum NTP-Server	81 81 C7 88 02 00	Bei IPv4: Unsigned32 Bei IPv6: Octet String	Alle

Tab. 86: Betriebslogbuch, Liste der Ereignis spezifischen Zusatzinformationen.

Die Periode zur Aufzeichnung der Betriebsereignisse kann gemäß Tab. 8 eingestellt werden.

Jedes Ereignis führt zu einem Eintrag im Betriebslogbuch.

Ereignisse werden wie folgt kodiert:

31-24	23-20	19 - 8	7 – 0	Bit-Nummer, Bit 31 ⇔ MSB	
Quelle	Ebene	reserviert	Ereignisnr.	Ereignis	Bedeutung
0x00	0x8	alle 0	0x00	Timer	Das eingestellte Zeitintervall ist seit dem letzten Ereignis abgelaufen (auch nach letztem Timeout)
0x00	0x1	alle 0	0x01	Netzwiederkehr	Die Speisung vom Modul ist wieder verfügbar
0x00	0x1	alle 0	0x02	Netzausfall	Die Speisung vom Modul ist nicht mehr vorhanden (der Eintrag erfolgt evtl. nach nächster Netzwiederkehr)
0x00	0x1	alle 0	0x03	Firmware-Aktivierung	Das Ereignis wird als erster Eintrag in das Betriebslogbuch nach erfolgreicher Aktivierung einer geladenen Firmware eingetragen
0x00	0x8	alle 0	0x04	Periodischer Reset	Periodischer Modem-Reset erfolgt
0x00	0x8	alle 0	0x05	Watchdog	Watchdog aufgetreten

0x00	0x8	alle 0	0x06	Synchron-Token erzeugt	Synchron-Token vom MUC-C generiert und gesendet
0x00	0x8	alle 0	0x07	Synchron-Token durchgeleitet	Synchron-Token von der WAN-Schnittstelle empfangen und über den lokalen Bus weitergeleitet
0x00	0x8	alle 0	0x08	Push per IP-T erfolgreich	Push-Nachricht wurde via IP-T erfolgreich versendet Ergänzend anzugeben: - Push Target, siehe Tab. 86, - Push Vorgang, siehe Tab. 86, - Push-Server-ID, siehe Tab. 86
0x00	0x8	alle 0	0x09	Push per IP-T nicht erfolgreich	Push-Nachricht konnte via IP-T nicht versendet werden Ergänzend anzugeben: - Push Target, siehe Tab. 86, - Push Vorgang, siehe Tab. 86, - Push-Server-ID, siehe Tab. 86.
0xXX	0x1	alle 0	0x08	WAN verfügbar	Modem erkennt Funknetz,
0xXX	0x1	alle 0	0x08	WAN verfügbar	Ethernet-Link vorhanden
0xXX	0x1	alle 0	0x08	WAN verfügbar	PLC-Netz erkannt
0xXX	0x1	alle 0	0x08	lokaler Bus verfügbar	Gerät erkennt Ethernet-Link am lokalen Bus
0xXX	0x1	alle 0	0x09	WAN nicht verfügbar	Funknetz nicht erkannt
0xXX	0x1	alle 0	0x09	WAN nicht verfügbar	Ethernet-Link nicht mehr vorhanden
0xXX	0x1	alle 0	0x09	WAN nicht verfügbar	PLC-Netz nicht mehr vorhanden
0xXX	0x1	alle 0	0x09	lokaler Bus nicht verfügbar	Ethernet-Link am lokalen Bus nicht mehr vorhanden
0xXX	0x1	alle 0	0x0A	Funknetz-Einwahl	GSM-Modem konnte sich erfolgreich ins GSM-Funknetz einwählen
0xXX	0x1	alle 0	0x0B	Funknetz-Abmeldung	GSM-Modem hat sich selbst vom GSM-Funknetz getrennt
0xXX	0x1	alle 0	0x0B	PLC-Netzabmeldung	PLC-Modem hat sich selbst vom PLC-Netz getrennt
0xXX	0x1	alle 0	0x0C	Funknetz-Abwurf	GSM-Modem wurde vom GSM-Funknetz getrennt
0xXX	0x4	alle 0	0x0A	IP-Zugang erfolgt	Punkt zu Punkt Verbindung zum Provider ist erfolgt
0xXX	0x4	alle 0	0x0A	PLC-Zugang erfolgt	Punkt zu Punkt Verbindung zum PLC-Konzentrator ist erfolgt
0xXX	0x4	alle 0	0x0D	IP-Zugang abgelehnt	Punkt zu Punkt Verbindung zum Provider wurde abgelehnt
0xXX	0x4	alle 0	0x0D	IP-Zugang abgelehnt	Punkt zu Punkt Verbindung zum PLC-Konzentrator wurde abgelehnt
0xXX	0x4	alle 0	0x0C	IP-Zugang getrennt	Punkt zu Punkt Verbindung wurde vom Provider beendet
0xXX	0x4	alle 0	0x0C	IP-Zugang getrennt	Punkt zu Punkt Verbindung wurde vom PLC-Konzentrator beendet

0xXX	0x4	alle 0	0x0B	IP-Zugang beendet	Punkt zu Punkt Verbindung wurde vom MUC-C beendet
0xXX	0x4	alle 0	0x0E	IP-Zugang verloren	IP Zugang unerwartet abgebrochen
0xXX	0x4	alle 0	0x0E	PLC-Zugang verloren	PLC Zugang unerwartet abgebrochen
0xXX	0x7	alle 0	0x0A	AL-Zugang erfolgt	Verbindung zu Dienst ²¹ ist aufgebaut Bei IP-T ergänzend anzugeben: - IP-T-Master und Port, siehe Tab. 86. Bei NTP ergänzend anzugeben: - NTP-Server, siehe Tab. 86.
0xXX	0x7	alle 0	0x0D	AL-Zugang abgelehnt	Verbindung zu Dienst abgelehnt
0xXX	0x7	alle 0	0x0C	AL-Zugang getrennt	Verbindung zu Dienst von Provider beendet
0xXX	0x7	alle 0	0x0B	AL-Zugang beendet	Verbindung zu Dienst von MUC-C beendet
0xXX	0x7	alle 0	0x0D	AL-Zugang verloren	Verbindung zu Dienst unerwartet abgebrochen
0xXX	0x7	alle 0	0x1A	AL-Push-Kanal aufgebaut	
0xXX	0x7	alle 0	0x1b	AL-Push-Kanal beendet	
0xXX	0x7	alle 0	0x1C	AL-Push-Kanal getrennt	
0xXX	0x7	alle 0	0x1d	AL-Push-Kanal Aufbau abgelehnt	
0xXX	0x7	alle 0	0x1E	AL-Push-Kanal verloren	
0xXX	0x8	alle 0	0x0A	Verbindung erfolgt	Reserviert für SyM ² , bei MUC nicht benutzt
0xXX	0x8	alle 0	0x0B	Verbindung beendet	Reserviert für SyM ² , bei MUC nicht benutzt
0xXX	0x8	alle 0	0x0C	Datensammler-Zugriff ²²	Lesender Zugriff auf einen Datensammler über die WAN-Schnittstelle (Pull wie Push).
0x00	0x8	alle 0	0x0D	FW-Unicast	Siehe Tab. 61, Eintrag erfolgt immer bei: - Start der Übertragung, - Ende der Übertragung, - Statusänderung der Übertragung, siehe TODO, - Zyklisch zusammen mit jedem zyklischen Logbucheintrag, wenn zeitgleich eine Übertragung aktiv (nicht pausiert) ist,
0x00	0x8	alle 0	0x0E	FW-Multicast	Wie vorstehend
0x00	0x8	alle 0	0x0F	Datei-Transfer	Wie vorstehend

²¹ NTP: 0xXX ⇔ 0x4A, IP-T: 0xXX ⇔ 0x49

²² Der Eintrag erfolgt im Logbuch am Datenspiegel.

Pos.	Hinweisnummer	Bedeutung
1	81 81 C7 C7 E1 00	Siehe SyM ² -Pflichtenheft
2	bis	Siehe SyM ² -Pflichtenheft
3	81 81 C7 C7 E1 20	Siehe SyM ² -Pflichtenheft
4	81 81 C7 C7 E1 21	Das zu übertragende Binary wird nicht akzeptiert, da der Name des zu übertragenden Binar bereits verwendet wird.

Tab. 88: MUC-spezifische Fehlernummern