PD CLC/TS 52056-8-7:2015

BSI Standards Publication

Electricity metering data exchange — The DLMS/COSEM suite

Part 8-7: AMC-SS PLC communication profile for neighbourhood networks

bsi.

…making excellence a habit.™

National foreword

This Published Document is the UK implementation of CLC/TS 52056-8-7:2015.

The UK participation in its preparation was entrusted to Technical Committee PEL/13, Electricity Meters.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2015.
Published by BSI Standards Limited 2015

ISBN 978 0 580 85919 9
ICS 35.240.60; 91.140.50

Compliance with a British Standard cannot confer immunity from legal obligations.

This Published Document was published under the authority of the Standards Policy and Strategy Committee on 31 May 2015.

Amendments/corrigenda issued since publication

Date	Text affected

TECHNICAL SPECIFICATION

SPÉCIFICATION TECHNIQUE

TECHNISCHE SPEZIFIKATION

CLC/TS 52056-8-7

April 2015

ICS 35.240.60; 91.140.50

English Version

Electricity metering data exchange - The DLMS/COSEM suite - Part 8-7: AMC-SS PLC communication profile for neighbourhood networks

This Technical Specification was approved by CENELEC on 2014-11-11.

CENELEC members are required to announce the existence of this TS in the same way as for an EN and to make the TS available promptly at national level in an appropriate form. It is permissible to keep conflicting national standards in force.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

CENELEC

European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

© 2015 CENELEC All rights of exploitation in any form and by any means reserved worldwide for CENELEC Members.

Ref. No. CLC/TS 52056-8-7:2015 E

CONTENTS

List of Figures

List of Tables

Foreword

This document (CLC/TS 52056-8-7:2015) has been prepared by CLC/TC 13 "Electrical energy measurement and control".

The following date is fixed:

- latest date by which the existence of this document has to be announced at national level (doa) 2015-07-24

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CENELEC [and/or CEN] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association.

1 Scope

This Technical Specification specifies the DLMS/COSEM communication profile using a compatibly-extendable form (CX1) of Adaptive Multi-Carrier Spread-Spectrum (AMC-SS) PLC for neighbourhood networks. Its structure is in line with the DLMS/COSEM framework as described in EN 62056-1-0[GK1].

The transport layer, the application layer and the data model are as specified in the EN 62056 DLMS/COSEM suite.

2 Normative references

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

CLC/TS 50590:2015, *Electricity metering data exchange - Lower layer PLC profile using Adaptive Multi-Carrier Spread-Spectrum for CX1 networks*

EN 61334-4-32:1996, *Distribution automation using distribution line carrier systems – Part 4 : Data communication protocols – Section 32: Data link layer – Logical link control (LLC) (IEC 61334-4-32:1996)*

EN 61334-4-1:1996, *Distribution automation using distribution line carrier systems – Part 4 : Data communication protocols – Section 1: Reference model of the communication system (IEC 61334-4-1:1996)*

EN 62056-5-3, *Electricity metering data exchange – The DLMS/COSEM suite – Part 5-3: DLMS/COSEM application layer (IEC 62056-5-3)*

EN 62056-6-1, *Electricity metering data exchange – The DLMS/COSEM suite – Part 61: OBIS Object identification system (IEC 62056-6-1)*

EN 62056-6-2, *Electricity metering data exchange – The DLMS/COSEM suite – Part 62: Interface classes (IEC 62056-6-2)*

FprEN 62056-4-7:2014, *Electricity metering data exchange - The DLMS/COSEM suite – Part 4-7: DLMS/COSEM transport layer for IP networks (IEC 62056-4-7:201X, 13/1570/CDV)*

EN 62056-9-7:2013, *Electricity metering data exchange – The DLMS/COSEM Suite – Part 9-7: Communication profile for TCP-UDP/IP networks (IEC 62056-9-7:2013)*

RFC 2507 - IP Header Compression. Authors: M. Degermark, B. Nordgren, S. Pink. February 1999. Available from http://tools.ietf.org/html/rfc2507

Ipv4 TOS Byte and Ipv6 Traffic Class Octet http://www.iana.org/assignments/ipv4-tos-byte/ipv4-tos-byte.xml

3 Abbreviations

.cnf	.confirm (primitive)
.ind	.indication (primitive)
.req	.request (primitive)
.res	.response (primitive)
ACSE	Association Control Service Element
AL	Application Layer
AMC-SS	Adaptive Multi-Carrier Spread Spectrum
AP	Application Process
A-PDU	Application Protocol Data Unit
ASE	Application Service Element
CENELEC	European Committee for Electrotechnical Standardization
CIN	Channel Identification Number
CL	Convergence Sub-Layer
COSEM	Companion Specification for Energy Metering
CRC	Cyclic Redundancy Check
CX1	Compatibly Extendable form of AMC-SS PLC
D8PSK	Differential Eight-Phase Shift Keying
DBLMAX	Maximum Data Block Length
DBPSK	Differential Binary Phase Shift Keying
DID	Device Identifier
DL	Data Link
DLL	Data Link Layer (layer 2)
DLMS	Device Language Message Specification
DLS	Data Link Service
DP	Data Priority
DPSK	Differential Phase Shift Keying
DQPSK	Differential Quaternary Phase Shift Keying
DSAP	Destination Service Access Point
FEC	Forward Error Correction
HES	(Metering) Head End System
Hz	Hertz
IEC	International Electrotechnical Commission
IP	Internet Protocol
IPv4	Internet Protocol, version 4
IPv6	Internet Protocol, version 6
kHz	kilo Hertz
LA	Link Address
LCN	Link Channel Number
LLC	Logical Link Control (sub-layer)
L-SAP	Data Link Layer Service Access Point
LNAP	Local Network Access Point

MAC	Medium Access Control (sub-layer)
MPDU	MAC Protocol Data Unit
N_NIN	Network Identification Number of a network node
NIN	Network Identification Number
NN	Neighbourhood Network
NNAP	Neighbourhood Network Access Point
OBIS	OBject Identification System
OSI	Open System Interconnection
PDU	Protocol Data Unit
PHY	Physical
PLC	Power-Line Communications
PPDU	PHY Protocol Data Unit
SAP	Service Access Point
SDU	Service Data Unit
SSAP	Source Service Access Point
TCP	Transmission Control Protocol
xDLMS_ASE	extended DLMS Application Service Element

4 Targeted communication environments

The *DLMS/COSEM PLC AMC-SS communication profile* is intended for remote data exchange on Neighbourhood Networks (NN) between *Neighbourhood Network Access Points* (NNAP) and *Local Network Access Points* (LNAPs) or *End Devices* using AMC-SS technology over the low voltage electricity distribution network as a communication medium at the C interface. The functional reference architecture is shown in Figure 1.

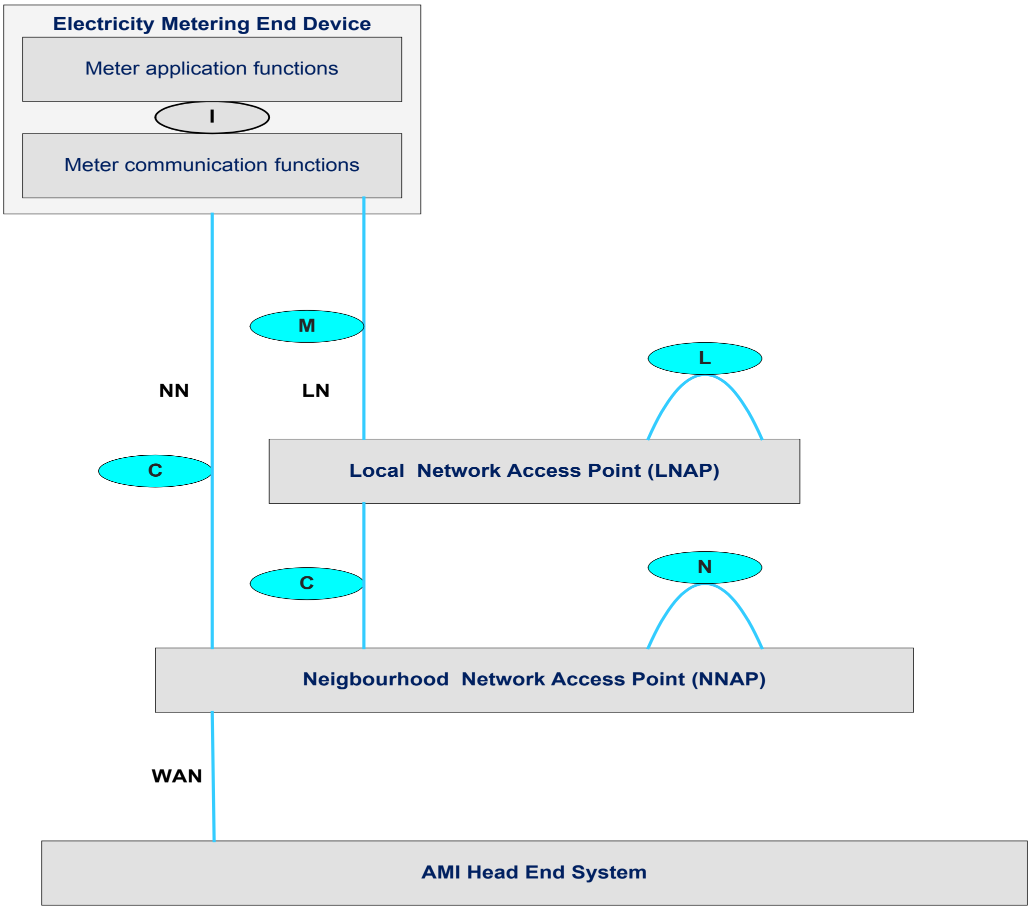

Figure 1 – Communication architecture

End devices – typically electricity meters – comprise application functions and communication functions. They may be connected directly to the NNAP via the C interface, or to an LNAP via an M interface, while the LNAP is connected to the NNAP via the C interface. The LNAP function may be co-located with the metering functions.

A NNAP comprises gateway functions and it may comprise concentrator functions. Upstream, it is connected to the Metering Head End System (HES) using suitable communication media and protocols.

End devices and LNAPs may communicate to different NNAPs, but to one NNAP only at a time. From the PLC communication point of view, the NNAP acts as the master node while end devices and LNAPs act as slave nodes.

NNAPs and similarly LNAPs may communicate to each other, but this is out of the scope of this specification, which covers the C interface only.

When the NNAP has concentrator functions, it acts as a DLMS/COSEM client. When the NNAP has gateway functionality only, then the HES plays the role of a DLMS/COSEM client. The end devices or the LNAPs play the role of DLMS/COSEM servers.

5 Reference model

The proposed protocol stacks use the following OSI layers as shown in Figure 2.

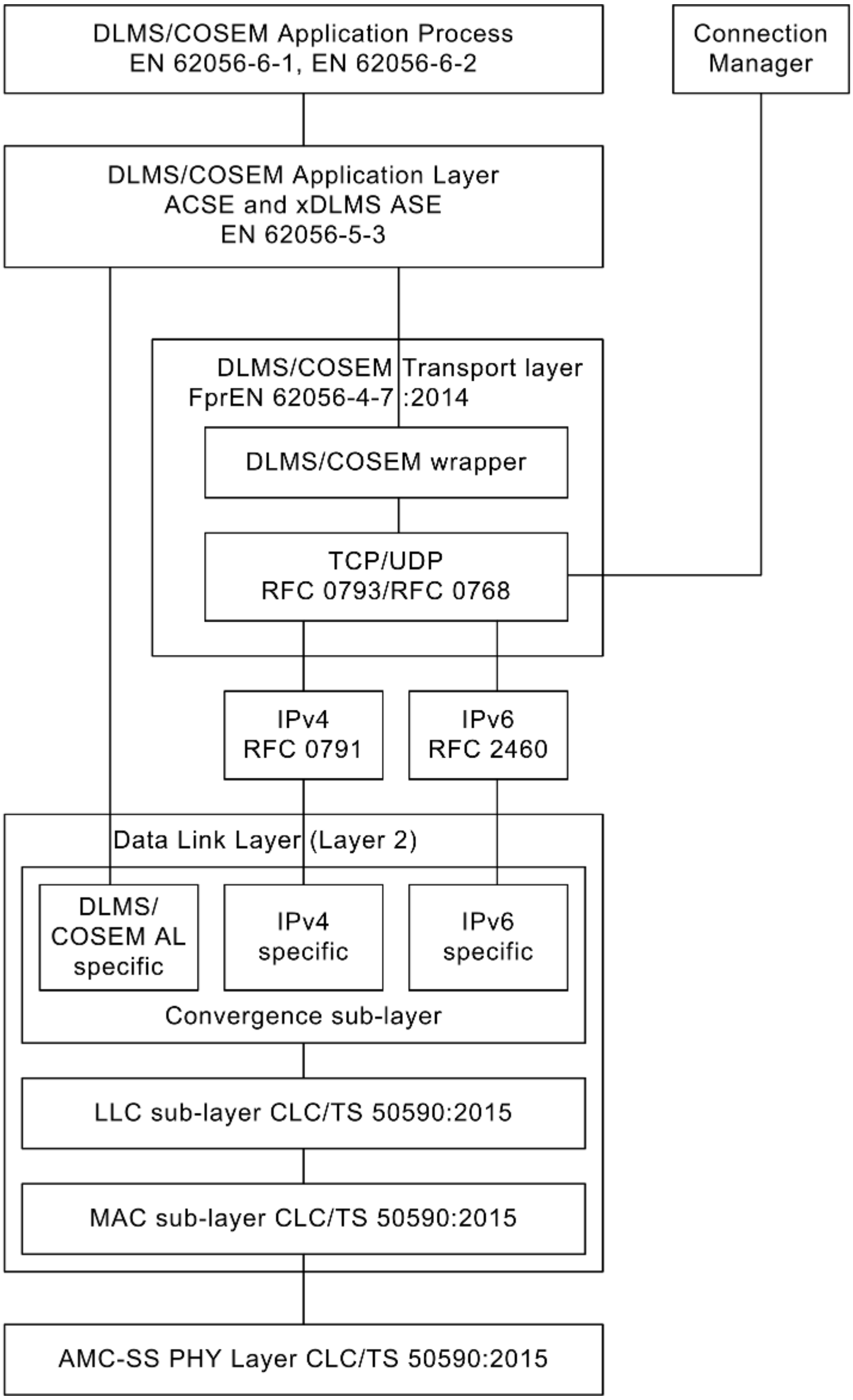

Figure 2 – DLMS/COSEM AMC-SS PLC architecture

The protocol layer are:

- the DLMS/COSEM data model as specified in EN 62056-6-1 and EN 62056-6-2;
- the DLMS/COSEM Application layer as specified in EN 62056-5-3 covering the Application, Presentation and Session functionalities;

- the DLMS/COSEM transport layer as specified in FprEN 62056-4-7:2014, used with the DLMS/COSEM TCP-UDP/IPv4 or TCP-UDP/IPv6 profile over AMC-SS, if IP transport is used;
- the IPv4 or the IPv6 network layer, if IP transport is used;
- the AMC-SS Data link layer, which consists of the Convergence, LLC and MAC sub-layers;
- the AMC-SS Physical layer.

NOTE The Physical layer and data link layer service primitives are specified in CLC/TS 50590[GK2].

6 Physical Layer (PHY)

6.1 Overview – main features and functions

This layer provides the interface between the equipment and the physical transmission medium that is the low-voltage distribution network. It transmits and receives MPDUs between neighbour nodes. The AMC-SS PHY uses a fast frequency-hopping spread spectrum technique combined with Differential Phase Shift Keying (DPSK) and forward error-correcting coding. Three differential modulation schemes are used: DBPSK, DQPSK and D8PSK. The system operates in the CENELEC A-band. This band covers the frequency range from 3 kHz up to 95 kHz. Frequencies in this band shall only be used for applications for monitoring or controlling the low-voltage distribution network, including energy usage of connected equipment and premises. A typical example of an application in this band would be metering communications.

This technique provides the following advantages:

- Robustness against time-frequency-selective fading;
- Robustness against pulse and narrowband interference, pulsating non-gaussian noise and combinations of them;
- Robustness against unwanted intermodulation effects;
- Low linearity requirements for the analogue front end;
- High power efficiency as a result of low peak to average ratio of the transmitted signal;
- Good electromagnetic compatibility between neighbouring systems.

The physical layer of AMC-SS is defined in Clause 5 of CLC/TS 50590:2015. The parameters of the physical layer are preconfigured.

6.2 PHY layer services

PHY services are generated by the MAC layer entity whenever data is to be transmitted to a peer MAC entity or entities, and passed to the PHY entity to request the sending of a PPDU to one or more remote PHY using the PHY transmission procedures. The primitives which are used between the MAC layer and the PHY layer are shown in Figure 3.

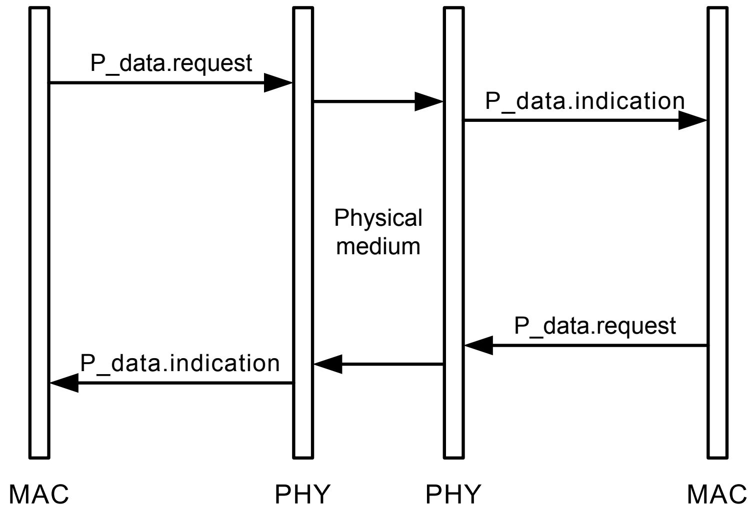

Figure 3 – Primitives between layer 2 and layer 1

The data, received from the MAC layer, is FEC-encoded. The encoded bit-sequence is segmented, interleaved, differentially encoded and mapped to the symbols and carrier frequencies.

7 Data Link Layer

7.1 Functions and structure

The Data Link Layer of AMC-SS provides a point to multi-point and point to point communication on the low voltage distribution network between a master node and one or more slave nodes. It contains three sub-layers:

- the Medium Access Control sub-layer;
- the Logical Link Control sub-layer;
- the Convergence sub-layer.

The structure of the Data Link Layer is shown in Figure 4.

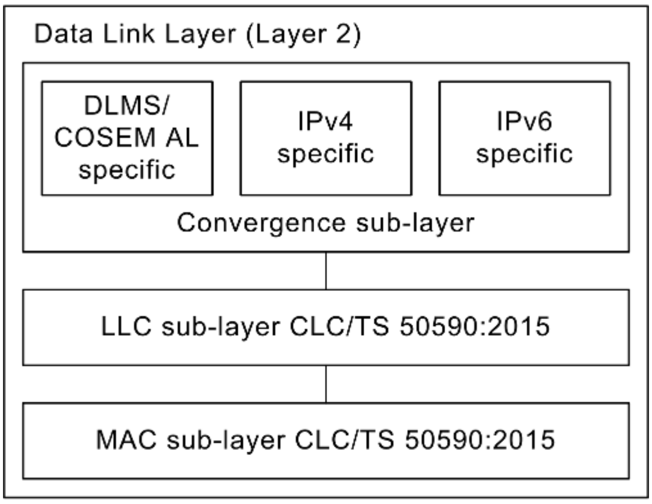

Figure 4 – Structure of Data Link Layer

Layer 2 services are used by the higher layer entity whenever data is to be transmitted to a peer entity or entities, and passed to the layer 2 entity to request the sending of a MPDU to one or more remote entities. The transmission of data is initiated with a CL_data.request primitive or CL_IPv4_data.request primitive (in the case of IPv4) or CL_IPv6_data.request primitive (in the case of IPv6). The success or failure of the transmission is locally indicated with a CL_data_request.confirm primitive or CL_IPv4_data_request.confirm primitive (in the case of IPv4) or CL_IPv6_data_request.confirm primitive (in the case of IPv6).

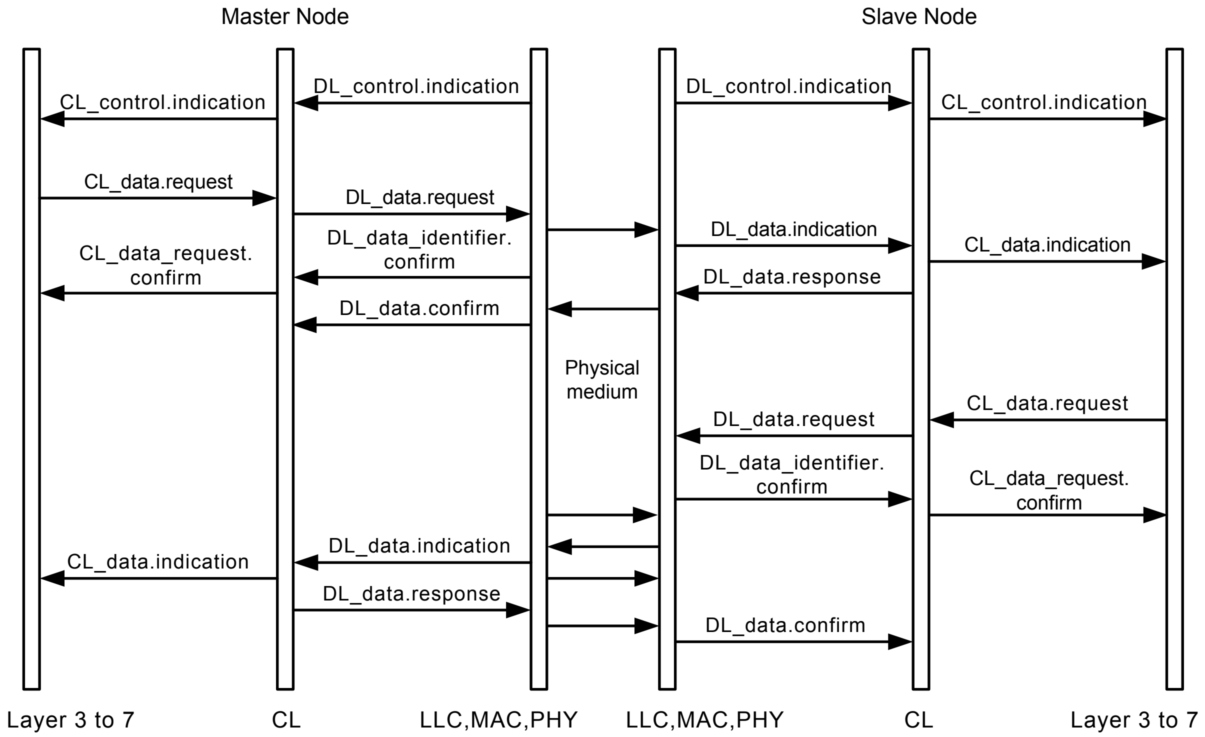

Figure 5 – DLMS/COSEM AL specific primitives between higher layer and layer 2 in a bidirectional data transmission

Received data is passed to the higher layer with a CL_data.indication primitive or CL_IPv4_data.indication primitive (in the case of IPv4) or CL_IPv6_data.indication primitive (in the case of IPv6).

Broadcast messages from a master node to the slave nodes, that are registered with it, are supported. In this case ‘request’ and ‘indication’ primitives are used for passing the information between layers. The success or failure of the transmission is locally indicated with a CL_data_request.confirm primitive or CL_ipv4_data_request.confirm primitive (in the case of IPv4) or CL_ipv6_data_request.confirm primitive (in the case of IPv6).

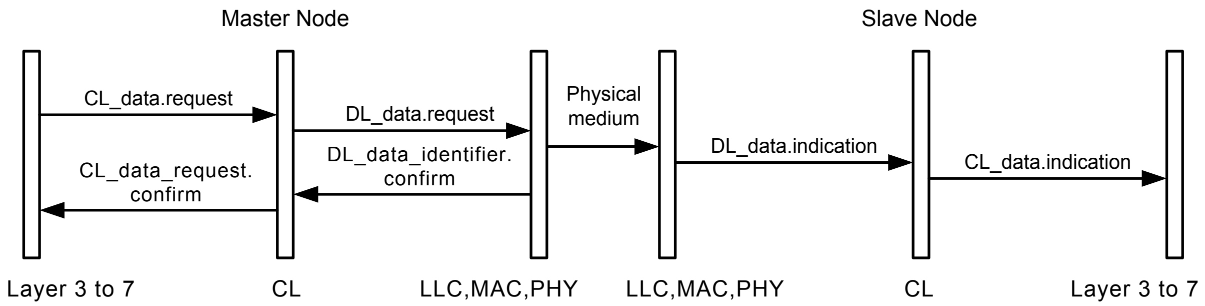

Figure 6 – DLMS/COSEM AL specific primitives between higher layer and layer 2 in the case of broadcast

An unsolicited message is transmitted by the slave node when it is polled by the master node or during the quick-check procedure (see Clause 6 of CLC/TS 50590:2015). In both cases ‘request’ and ‘indication’ primitives are used for passing the information between layers. The success or failure of the transmission is locally indicated with a CL_data_request.confirm primitive or CL_ipv4_data_request.confirm primitive (in the case of IPv4) or CL_ipv6_data_request.confirm primitive (in the case of IPv6).

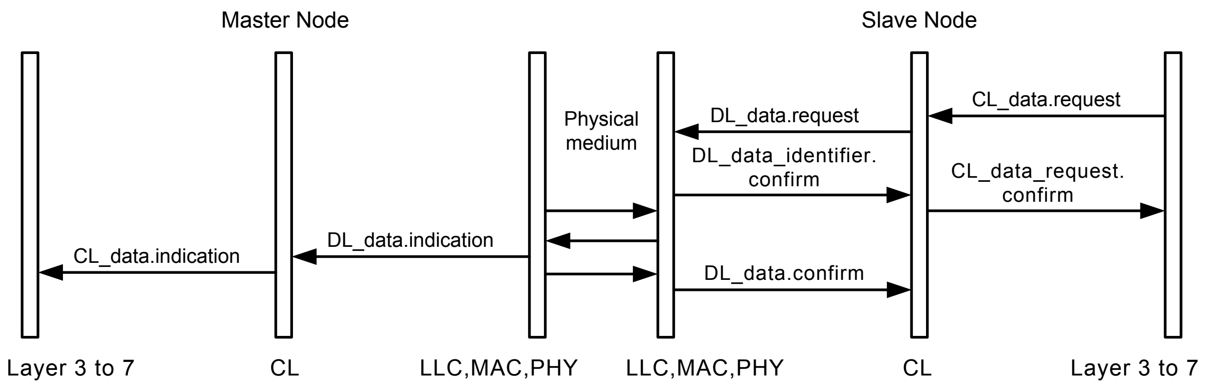

Figure 7 – DLMS/COSEM AL specific primitives between higher layers and layer 2 in the case of an unsolicited data transmission

The Figure 5, Figure 6 and Figure 7 above show the DLMS/COSEM AL specific examples of CL primitive usage. In case of IPv4 or IPv6, instead of the primitives with the prefix ‘CL’ the corresponding primitives with the prefixes ‘CL_IPv4’ or ‘CL_IPv6’ respectively are used.

The functions and the coding of the message are defined in Annex B of CLC/TS 50590:2015.

7.2 The Medium Access Control sub-layer

The Medium Access Control sub-layer provides a connection-less transport-mechanism. Point-to-point, point-to-multipoint and multicast/broadcast services are supported. The MAC sub-layer provides the simultaneous forwarding procedure, where all involved nodes retransmit the received messages simultaneously (synchronous). This procedure extends the range and improves the transmission quality. The transmitted information is protected by CRC sequences. The MAC sub-layer of AMC-SS is defined in 6.1 to 6.3 of CLC/TS 50590:2015. The parameters of the MAC sub-layer are preconfigured and listed in Annex D of CLC/TS 50590:2015.

7.3 The Logical Link Control sub-layer

7.3.1 Overview

The Logical Link Control sub-layer is specified in 6.4 to 6.9 of CLC/TS 50590:2015. The parameters of the LLC sub-layer are preconfigured and listed in Annex D of CLC/TS 50590:2015. The LLC sub-layer provides two types of services:

- layer-2-network management functions;
- point to point and point to multi-point communication functions.

7.3.2 Layer-2-network management functions

The transport of network management information is indicated by setting the Data Link Service (DLS) bit in the LLC control field to DLS=0. The network management functions are:

- Search for new slave nodes;
- Registration and link address management;
- Polling of registered nodes;
- Telegrams for the coordination between nodes;
- Cell change by slave node;
- Clock synchronization for lower layer management;
- Status enquiry;
- PHY link test;
- PHY quality data enquiry.

The network management procedures are described in Clauses 6 and 7 of CLC/TS 50590:2015. The coding of the messages is defined in Annex B of CLC/TS 50590:2015.

7.3.3 Communication functions

The transport of higher layer information is indicated by setting the Data Link Service (DLS) bit in the LLC control field to DLS=1. If DLS=1, then the first byte of the data block contains the LLC extended control field, which consists of the 4-bit Data Priority (DP) and the 4-bit Link Channel Number (LCN). These are used for the prioritization and multiplexing of different protocols and applications. The usage of the Data Priority is left to the implementation. The coding of the LLC extended control field byte is shown in the Table 1 below.

Table 1 – Coding of the DLC extended control field

Field	Code value	Meaning
Data Priority, DP	0 to 15	0: lowest priority 1-14: priority 1 to 14 15: highest priority
Link Channel Number, LCN	0 to 4, 8 to 15 5 6 7	reserved IPv4-PDU IPv6-PDU DLMS-PDU

7.3.4 Segmentation and reassembly

The Logical Link Control sub-layer will perform segmentation and reassembly, if the packet is too long. The coding details are defined in Annex B of CLC/TS 50590:2015. An example diagram can be found in 6.5.12 of CLC/TS 50590:2015.

7.4 The Convergence sub-layer

7.4.1 Overview

The Data Link layer contains at the top a convergence sub-layer that provides the service access points for the higher layers. The following types of this sub-layer are specified:

- DLMS/COSEM AL specific convergence sub-layer;
- IPv4 specific convergence sub-layer;
- IPv6 specific convergence sub-layer.

7.4.2 DLMS/COSEM AL specific convergence sub-layer

7.4.2.1 General

The DLMS/COSEM AL specific convergence sub-layer provides transport and distribution of the DLMS/COSEM Application layer data. This convergence sub-layer provides the service access points (SAPs) for the application processes in the master and slave nodes.

7.4.2.2 Primitives

7.4.2.2.1 CL_data.request

The CL_data.request primitive is used by the application layer to initiate the data transmission process.

The semantics of the primitive are as follows:

CL_data.request (ServiceType, DestinationSAP, SourceSAP, DestinationAddress, SourceAddress, DataPriority, DataLength, Data)

The *ServiceType* parameter specifies the type of the required transmission. The parameter may have one of the following values:

0: unicast data transmission is requested;

1: non-acknowledged broadcast data transmission is requested. This type is only used by a master node.

The *DestinationSAP* and *SourceSAP* parameter contain 8-bit-values of data link layer selectors (also called "L-SAP") DSAP and SSAP respectively that specify the destination service access point and source service access point field in the data link layer PDU. For the specification of the L-SAP values (SSAP, DSAP) refer to EN 61334-4-32 and EN 61334-4-1.

The Destination*Address* parameter specifies the destination address for the transmission. If used in the master node, it contains the serial number (device identifier, DID) of the slave node. In the slave node, the field contains CIN of the master node.

The Source*Address* parameter specifies the source address of the transmission. If used in the slave node, it contains the serial number (device identifier, DID) of the slave node. In the master node the field contains CIN of the master node.

The Data*Priority* parameter indicates the priority of the data to be sent. The master node is free to select the priority value. If a slave node responds to a previously received request, then the same priority value as contained in the corresponding request should be used in the response. It may have any of the following 16 values:

0: priority level 0 (lowest priority);

1 to 14: priority level 1 to 14 respectively;

15: priority level 15 (highest priority).

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX (see Annex D of CLC/TS 50590:2015). For the *ServiceType* values 0 and 1 the *DataLength* may have any value in the range from 1 to (16 x (DBLMAX-3) – 2).

The *Data* parameter is a buffer of octets that contains the data to be transmitted.

The following Table 2 shows the mapping of the CL_data.request primitive to the DL_data.request primitive in the master node.

Table 2 – Mapping of CL_data.request to DL_data.request in the master node

CL_data.request		**DL_data.request**	
Parameter	**Contents**	**Parameter**	**Contents**
ServiceType	0, 1	ServiceType	1, if ServiceType of CL_data.request =0 0, if ServiceType of CL_data.request =1
SourceAddress	CIN	CellIdentificationNumber	CIN
DestinationAddress	Device Identifier, if ServiceType = 0 Not used, if ServiceType = 1	LinkAddress	LA = 0 to 0xFFFD, if ServiceType of CL_data.request =0 LA = 0xFFFF, if ServiceType of CL_data.request =1
		LinkChannelNumber	7
DataPriority	0 to 15	DataPriority	0 to 15
DataLength	1 to 16 x (DBLMAX-3) – 2	DataLength	3 to 16 x (DBLMAX-3)
DestinationSAP	0 to 255	1st byte of Data	DSAP
SourceSAP	0 to 255	2nd byte of Data	SSAP
Data	A-PDU	Data bytes 3 to DataLength	A-PDU

The following Table 3 shows the mapping of the CL_data.request primitive to the DL_data.request primitive in the slave node.

Table 3 – Mapping of CL_data.request to DL_data.request in the slave node

CL_data.request		DL_data.request	
Parameter	**Contents**	**Parameter**	**Contents**
ServiceType	0	ServiceType	1
DestinationAddress	CIN	CellIdentificationNumber	CIN
SourceAddress	Device Identifier	LinkAddress	LA
		LinkChannelNumber	7
DataPriority	0 to 15	DataPriority	0 to 15
DataLength	1 to 16 x (DBLMAX-3) – 2	DataLength	3 to 16 x (DBLMAX-3)
DestinationSAP	0 to 255	1^{st} byte of Data	DSAP
SourceSAP	0 to 255	2^{nd} byte of Data	SSAP
Data	A-PDU	Data bytes 3 to DataLength	A-PDU

7.4.2.2.2 CL_data.indication

The CL_data.indication primitive informs the application layer about the reception of data.

The semantics of the primitive are as follows:

CL_data.indication (ServiceType, DestinationSAP, SourceSAP, DestinationAddress, SourceAddress, DataPriority, DataLength, Data)

The *ServiceType* parameter specifies the type of the transmission. The parameter may have one of the following values:

0: unicast data transmission was used;

1: non-acknowledged broadcast data transmission was used.

The *DestinationSAP* and *SourceSAP* parameter contain 8-bit-values of data link layer selectors (also called “L-SAP”) DSAP and SSAP respectively that specify the destination service access point and source service access point field in the data link layer PDU. For the specification of the L-SAP values (SSAP, DSAP) refer to EN 61334-4-32 and EN 61334-4-1.

The Destination*Address* parameter specifies the destination address for the transmission. If used in the master node, it contains CIN of the master node. In the slave node, the field contains the serial number (device identifier, DID) of the slave node.

The Source*Address* parameter specifies the source address of the transmission. If used in the master node, it contains the serial number (device identifier, DID) of the slave node. In the slave node the field contains CIN of the master node.

The Data*Priority* parameter indicates the priority of the data that has been received. It may have any of the following 16 values:

0: priority level 0 (lowest priority)

1 to 14: priority level 1 to 14 respectively

15: priority level 15 (highest priority)

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX. For the *ServiceType* parameter values 0 and 1 the *DataLength* parameter may have any value in the range from 1 to 16 x (DBLMAX-3) – 2.

The *Data* parameter is a buffer of octets that contains the data that was received.

The following Table 4 shows the mapping of the DL_data.indication primitive to the CL_data.indication primitive in the master node.

Table 4 – Mapping of DL_data.indication to CL_data.indication in the master node

DL_data.indication		CL_data.indication	
Parameter	**Contents**	**Parameter**	**Contents**
DataIdentifier	255	ServiceType	0
CellIdentificationNumber	CIN	DestinationAddress	CIN
LinkAddress	LA	SourceAddress	Device Identifier
LinkChannelNumber	7		
DataPriority	0 to 15	DataPriority	0 to 15
DataLength	3 to 16 x (DBLMAX-3)	DataLength	1 to 16 x (DBLMAX-3) – 2
1st byte of Data	DSAP	DestinationSAP	0 to 255
2nd byte of Data	SSAP	SourceSAP	0 to 255
Data bytes 3 to DataLength	A-PDU	Data	A-PDU

The following Table 5 shows the mapping of the DL_data.indication primitive to the CL_data.indication primitive in the slave node.

Table 5 – Mapping of DL_data.indication to CL_data.indication in the slave node

DL_data.indication		CL_data.indication	
Parameter	**Contents**	**Parameter**	**Contents**
DataIdentifier	254, 255	ServiceType	0, if DataIdentifier = 255 1, if DataIdentifier = 254
CellIdentificationNumber	CIN	SourceAddress	CIN
LinkAddress	LA = 0 to 0xFFFD, if DataIdentifier = 255; LA = 0xFFFF, if DataIdentifier = 254	DestinationAddress	Device Identifier, if DataIdentifier = 255; Not used, if DataIdentifier = 254
LinkChannelNumber	7		
DataPriority	0 to 15	DataPriority	0 to 15
DataLength	3 to 16 x (DBLMAX-3)	DataLength	1 to 16 x (DBLMAX-3) - 2
1st byte of Data	DSAP	DestinationSAP	0 to 255
2nd byte of Data	SSAP	SourceSAP	0 to 255
Data bytes 3 to DataLength	A-PDU	Data	A-PDU

7.4.2.2.3 CL_data_request.confirm

The CL_data_request.confirm primitive is a local response to the application layer entity, which indicates the local success or failure of the associated previous CL_data.request primitive.

The semantics of the primitive are as follows:

CL_data_request.confirm (Status, ServiceType, DestinationSAP, SourceSAP, DestinationAddress, DataPriority, DataLength, Data)

The *Status* parameter is used to pass the status information back to the requesting application layer entity. It is used to indicate the success or failure of the associated previous CL_data.request primitive. The parameter may have one of the following values:

0: success of the associated previous CL_data.request;

1: failure of the associated previous CL_data.request. The transmit buffer of the requested entity is occupied;

2: failure of the associated previous CL_data.request. There is no communication to the addressed destination node. The node registration is lost;

3: failure of the associated previous CL_data.request. The communication to the addressed destination node is temporarily lost.

The *ServiceType* parameter specifies the type of the requested transmission. The parameter may have one of the following values:

0: unicast data transmission is requested;

1: non-acknowledged broadcast data transmission is requested. The value is only used by a master node.

The *DestinationSAP* and *SourceSAP* parameter contain 8-bit-values of data link layer selectors (also called "L-SAP") DSAP and SSAP respectively that specify the destination service access point and source service access point field in the data link layer PDU. Their values correspond to those of the previous associated CL_data.request primitive. For the specification of the L-SAP values (SSAP, DSAP) refer to EN 61334-4-32 and EN 61334-4-1.

The *DestinationAddress* parameter specifies the destination address for the transmission. Its value corresponds to that of the associated previous CL_data.request primitive. If used in the slave node, the *DestinationAddress* parameter contains the serial number (device identifier, DID) of the slave node. In the master node the field contains the CIN of the master node.

The Data*Priority* parameter indicates the priority of the data to be sent. The same priority value as contained in the corresponding request should be used. It may have any of the following 16 values:

0: priority level 0 (lowest priority);

1 to 14: priority level 1 to 14 respectivelly;

15: priority level 15 (highest priority).

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX (see Annex D of CLC/TS 50590:2015). The *DataLength* may have any value in the range from 1 to (16 x (DBLMAX-3) – 2).

The *Data* parameter is a buffer of octets that contains the data to be transmitted.

The following Table 6 shows the mapping of the DL_data_identifier.confirm primitive to the CL_data_request.confirm primitive in the master node.

Table 6 – Mapping of DL_data_identifier.confirm to CL_data_request.confirm in the master node

DL_data_identifier.confirm		CL_data_request.confirm	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 3	Status	0 to 3
DataIdentifier	254 or 255, if Status = 0; Not used, if Status = 1 to3.	ServiceType	0, if DataIdentifier = 255; 1, if DataIdentifier = 254; Not used, if Status = 1 to3.
CellIdentificationNumber	CIN		
LinkAddress	LA = 0 to 0xFFFD, if DataIdentifier = 255; LA = 0xFFFF, if DataIdentifier = 254	DestinationAddress	DeviceIdentifier, if ServiceType = 0; Not used, if ServiceType = 1
LinkChannelNumber	7		
DataPriority	0 to 15	DataPriority	0 to 15
DataLength	3 to 16 x (DBLMAX-3)	DataLength	1 to 16 x (DBLMAX-3) – 2
1st byte of Data	DSAP	DestinationSAP	0 to 255
2nd byte of Data	SSAP	SourceSAP	0 to 255
Data bytes 3 to DataLength	A-PDU	Data	A-PDU

The following Table 7 shows the mapping of the DL_data_identifier.confirm primitive to the CL_data_request.confirm primitive in the slave node.

Table 7 – Mapping of DL_data_identifier.confirm to CL_data_request.confirm in the slave node

DL_data_identifier.confirm		CL_data_request.confirm	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 3	Status	0 to 3
DataIdentifier	255, if Status = 0; Not used, if Status = 1 to 3.	ServiceType	0, if Status = 0; Not used, if Status = 1 to 3.
CellIdentificationNumber	CIN	DestinationAddress	CIN
LinkAddress	LA = 0 to 0xFFFD		
LinkChannelNumber	7		
DataPriority	0 to 15	DataPriority	0 to 15
DataLength	3 to 16 x (DBLMAX-3)	DataLength	1 to 16 x (DBLMAX-3) – 2
1st byte of Data	DSAP	DestinationSAP	0 to 255
2nd byte of Data	SSAP	SourceSAP	0 to 255
Data bytes 3 to DataLength	A-PDU	Data	A-PDU

7.4.2.2.4 Reserved SAP values

The reserved SAP values for the master node operating as DLMS-client and for the slave node operating as DLMS server are shown in following Table 8 and Table 9.

Table 8 – Reserved SAP values on the master node side

Address	Meaning
0x00	No-station
0x01	Client (Master Node) management process.
0x10	Public Client

Table 9 – Reserved SAP values on the slave node side

Address	Meaning
0x00	No-station
0x01	Management Logical Device
0x02 to 0x0F	Reserved
0xFF	All-station (Broadcast)

7.4.3 IPv4 specific convergence sub-layer

7.4.3.1 Architecture

The IPv4 specific convergence sub-layer provides an efficient method for transferring IPv4 packets over the AMC-SS networks. The IPv4 PDUs may be compressed according to RFC 2507 with the predefined compression parameters specified in Annex A. The communication architecture for IPv4 is shown in Figure 8.

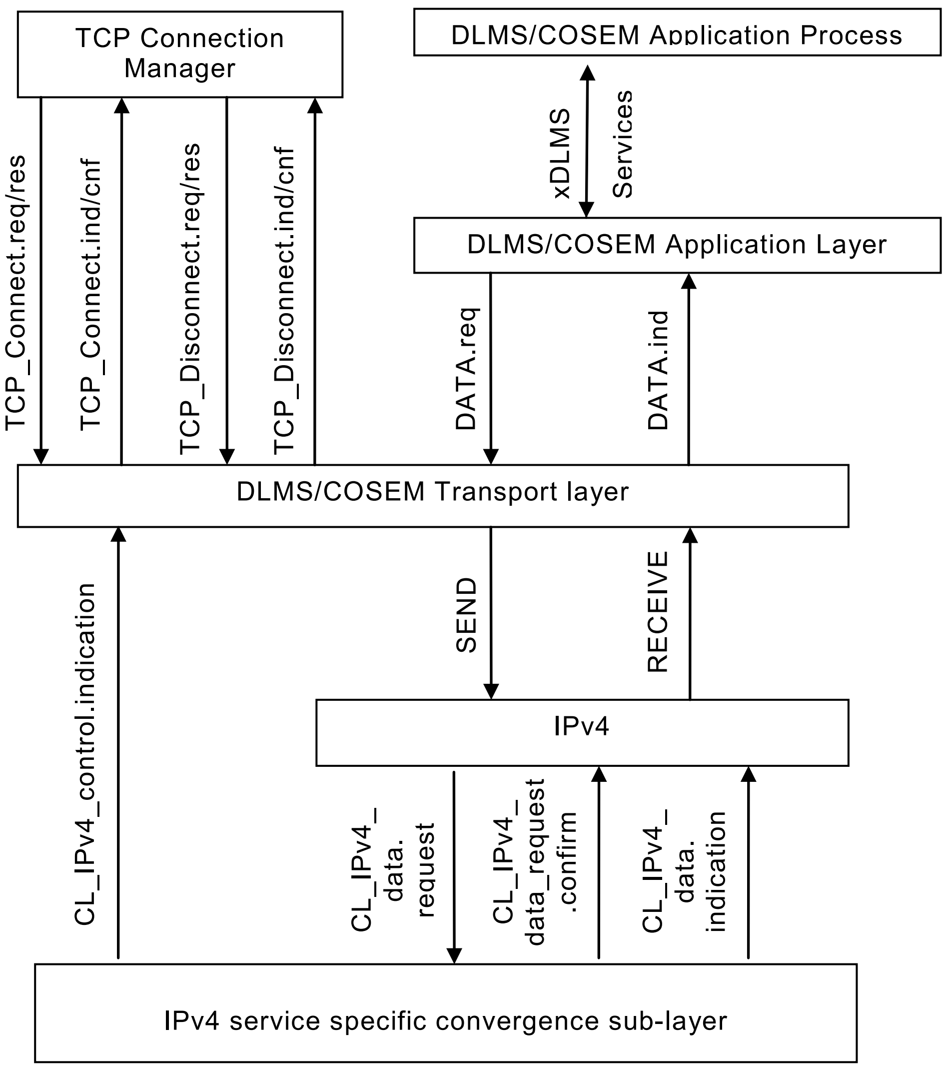

Figure 8 – The IPv4 communication architecture

7.4.3.2 The TCP connection manager

The TCP Connection Manager is specified in EN 62056-9-7.

7.4.3.3 Primitives

7.4.3.3.1 CL_IPv4_data.request

The CL_IPv4_data.request primitive is used by the IPv4 layer to initiate the data transmission process.

The semantics of the primitive are as follows:

CL_IPv4_data.request (Destination Address, SourceAddress, CompressionType, DataLength, Data)

The *DestinationAddress* parameter is an IPv4 address. It may have any values defined in 7.5.

The *SourceAddress* parameter is an IPv4 address. It may have IPv4 unicast address values defined in 7.5.

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX (see Annex D of CLC/TS 50590:2015). The *DataLength* may have any value in the range from 3 to (16 x (DBLMAX-3) – 1).

The *CompressionType* parameter specifies the type of a uncompressed or compressed IPv4 packet or of a CONTEXT_STATE packet to be transmitted. The parameter may have one of the following values:

0: data contains a regular, uncompressed, IPv4 datagram;

1: data contains an IPv4 datagram with the format FULL HEADER as specified in RFC 2507;

2: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_TCP as specified in RFC 2507;

3: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_TCP_NODELTA as specified in RFC 2507;

4: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_NON_TCP as specified in RFC 2507;

5: data contains a CONTEXT_STATE message as specified in RFC 2507.

The *Data* parameter is the CL service data unit to be transmitted. It may contain an IPv4 packet or a CONTEXT_STATE packet as specified in RFC 2507.

The following Table 10 shows the mapping of the CL_IPv4_data.request primitive to the DL_data.request primitive in the master node.

Table 10 – Mapping of CL_IPv4_data.request to DL_data.request in the master node

CL_IPv4_data.request		DL_data.request	
Parameter	**Contents**	**Parameter**	**Contents**
		ServiceType	0, if *DestinationAddress* contains an IPv4 unicast address; 1, if *DestinationAddress* contains the IPv4 broadcast address 255.255.255.255
SourceAddress	IPv4 unicast address of the master node	CellIdentificationNumber	CIN of the master node
DestinationAddress	IPv4 broadcast address or IPv4 unicast address of a slave node registered with the master node	LinkAddress	LA = 0 to 0xFFFF = lower 16 bit of the *Destination Address*
		LinkChannelNumber	5
		DataPriority	0 to 15
DataLength	3 to 16 x (DBLMAX-3)-1	DataLength	DataLength value of CL_IPv4_data.request plus one. The maximal DataLength value is limited to 16 x (DBLMAX-3)
CompressionType	0 to 5	1^{st} byte of Data	0 to 5
Data	IPv4 or CONTEXT_STATE packet	Data bytes 2 to DataLength	IPv4 or CONTEXT_STATE packet

The following Table 11 shows the mapping of the CL_ IPv4_data.request primitive to the DL_data.request primitive in the slave node.

Table 11 – Mapping of CL_IPv4_data.request to DL_data.request in the slave node

CL_IPv4_data.request		DL_data.request	
Parameter	**Contents**	**Parameter**	**Contents**
		ServiceType	1
Destination Address	IPv4 unicast address of the master node	CellIdentificationNumber	CIN of the master node the slave is registered with
SourceAddress	IPv4 unicast address of the slave node	LinkAddress	LA = 0 to 0xFFFD= lower 16 bit of the *SourceAddress*
		LinkChannelNumber	5
		DataPriority	0 to 15
DataLength	3 to 16 x (DBLMAX-3)-1	DataLength	DataLength value of CL_IPv4_data.request plus one. The maximal DataLength value is limited to 16 x (DBLMAX-3)
CompressionType	0 to 5	1st byte of Data	0 to 5
Data	IPv4 or CONTEXT_STATE packet	Data bytes 2 to DataLength	IPv4 or CONTEXT_STATE packet

7.4.3.3.2 CL_IPv4_data.indication

The CL_data.indication primitive informs the IPv4 layer about the reception of data.

The semantics of the primitive are as follows:

CL_IPv4_data.indication (Destination Address, SourceAddress, CompressionType, DataLength, Data)

The *DestinationAddress* parameter is an IPv4 address. It may have any values defined in 7.5.

The *SourceAddress* parameter is an IPv4 address. It may have IPv4 unicast address values defined in 7.5.

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX (see Annex D of CLC/TS 50590:2015). The *DataLength* may have any value in the range from 3 to (16 x (DBLMAX-3) – 1).

The *CompressionType* parameter specifies the type of the received uncompressed or compressed IPv4 packet or CONTEXT_STATE packet. The parameter may have one of the following values:

0: data contains a regular, uncompressed, IPv4 datagram;

1: data contains an IPv4 datagram with the format FULL HEADER as specified in RFC 2507;

2: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_TCP as specified in RFC 2507;

3: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_TCP_NODELTA as specified in RFC 2507;

4: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_ NON_TCP as specified in RFC 2507;

5: data contains a CONTEXT_STATE message as specified in RFC 2507.

The *Data* parameter is the CL service data unit that was received. It may contain an IPv4 packet or a CONTEXT_STATE packet as specified in RFC 2507.

The following Table 12 shows the mapping of the DL_data.indication primitive to the CL_IPv4_data.indication primitive in the master node.

Table 12 – Mapping of DL_data.indication to CL_IPv4_data.indication in the master node

DL_data.indication		CL_IPv4_data.indication	
Parameter	**Contents**	**Parameter**	**Contents**
DataIdentifier	255		
CellIdentificationNumber	CIN	DestinationAddress	IPv4 unicast address of the master node
LinkAddress	LA = 0 to 0xFFFD	SourceAddress	IPv4 unicast address of the slave node
LinkChannelNumber	5		
DataPriority	0 to 15		
DataLength	4 to 16 x (DBLMAX-3)	DataLength	3 to 16 x (DBLMAX-3)-1 = DataLength value of DL_data.indication minus one.
1st byte of Data	0 to 5	CompressionType	0 to 5
Data bytes 2 to DataLength	IPv4 or CONTEXT_STATE packet	Data	IPv4 or CONTEXT_STATE packet

The following Table 13 shows the mapping of the DL_data.indication primitive to the CL_IPv4_data.indication primitive in the slave node.

Table 13 – Mapping of DL_data.indication to CL_IPv4_data.indication in the slave node

DL_data.indication		CL_IPv4_data.indication	
Parameter	**Contents**	**Parameter**	**Contents**
DataIdentifier	254, 255		
CellIdentificationNumber	CIN	SourceAddress	IPv4 unicast address of the master node
LinkAddress	LA = 0 to 0xFFFD, if DataIdentifier = 255; LA = 0xFFFF, if DataIdentifier = 254	DestinationAddress	IPv4 unicast address, if DataIdentifier=255; IPv4 broadcast address, if if DataIdentifier=254
LinkChannelNumber	5		
DataPriority	0 to 15		
DataLength	4 to 16 x (DBLMAX-3)	DataLength	3 to 16 x (DBLMAX-3)-1 = DataLength value of DL_data.indication minus one.
1st byte of Data	0 to 5	CompressionType	0 to 5
Data bytes 2 to DataLength	IPv4 or CONTEXT_STATE packet	Data	IPv4 or CONTEXT_STATE packet

7.4.3.3.3 CL_IPv4_data_request.confirm

The CL_IPv4_data_request.confirm primitive is a local response to the IPv4 layer entity, which indicates the local success or failure of the associated previous CL_IPv4_data.request primitive.

The semantics of the primitive are as follows:

CL_IPv4_data_request.confirm (Status, Destination Address, CompressionType, DataLength, Data)

The *Status* parameter is used to pass the status information back to the requesting IPv4 layer entity. It is used to indicate the success or failure of the associated previous CL_IPv4_data.request primitive. The parameter may have one of the following values:

0: success of the associated previous CL_IPv4_data.request;

1: failure of the associated previous CL_IPv4_data.request. The transmit buffer of the requested entity is occupied;

2: failure of the associated previous CL_IPv4_data.request. There is no communication to the addressed destination node. The node registration is lost;

3: failure of the associated previous CL_IPv4_data.request. The communication to the addressed destination node is temporarily lost.

The *DestinationAddress* parameter is an IPv4 address. It may have any values defined in 7.5.

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX (see Annex D of CLC/TS 50590:2015). The *DataLength* may have any value in the range from 3 to (16 x (DBLMAX-3) – 1).

The *CompressionType* parameter specifies the type of the uncompressed or compressed IPv4 packet or CONTEXT_STATE packet. The parameter may have one of the following values:

0: data contains a regular, uncompressed, IPv4 datagram;

1: data contains an IPv4 datagram with the format FULL HEADER as specified in RFC 2507;

2: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_TCP as specified in RFC 2507;

3: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_TCP_NODELTA as specified in RFC 2507;

4: data contains an IPv4 datagram with a compressed header with the format COMPRESSED_ NON_TCP as specified in RFC 2507;

5: data contains a CONTEXT_STATE message as specified in RFC 2507.

The *Data* parameter is the CL service data unit that was contained in the associated previous CL_IPv4_data.request primitive. It may contain an IPv4 packet or a CONTEXT_STATE packet as specified in RFC 2507.

The following Table 14 shows the mapping of the DL_data_identifier.confirm primitive to the CL_IPv4_data_request.confirm primitive in the master node.

Table 14 – Mapping of DL_data_identifier.confirm to CL_IPv4_data_request.confirm in the master node

DL_data_identifier.confirm		CL_IPv4_data_request.confirm	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 3	Status	0 to 3
DataIdentifier	254 or 255, if Status = 0; Not used, if Status = 1 to3.		
LinkAddress	LA = 0 to 0xFFFD, if DataIdentifier = 255; LA = 0xFFFF, if DataIdentifier = 254	DestinationAddress	IPv4 unicast address, if DataIdentifier=255; IPv4 broadcast address, if if DataIdentifier=254
LinkChannelNumber	5		
DataPriority	0 to 15		
DataLength	4 to 16 x (DBLMAX-3)	DataLength	3 to 16 x (DBLMAX-3)-1 = DataLength value of DL_data.identifier.confirm minus one.
1^{st} byte of Data	0 to 5	CompressionType	0 to 5
Data bytes 2 to DataLength	IPv4 or CONTEXT_STATE packet	Data	IPv4 or CONTEXT_STATE packet

The following Table 15 shows the mapping of the DL_data_identifier.confirm primitive to the CL_IPv4_data_request.confirm primitive in the slave node.

Table 15 – Mapping of DL_data_identifier.confirm to CL_IPv4_data_request.confirm in the slave node

DL_data_identifier.confirm		CL_IPv4_data_request.confirm	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 3	Status	0 to 3
DataIdentifier	255, if Status = 0; Not used, if Status = 1 to 3.		
CellIdentificationNumber	CIN	DestinationAddress	IPv4 unicast address of the master node
LinkChannelNumber	5		
DataPriority	0 to 15		
DataLength	4 to 16 x (DBLMAX-3)	DataLength	3 to 16 x (DBLMAX-3)-1 = DataLength value of DL_data.identifier.confirm minus one.
1^{st} byte of Data	0 to 5	CompressionType	0 to 5
Data bytes 2 to DataLength	IPv4 or CONTEXT_STATE packet	Data	IPv4 or CONTEXT_STATE packet

7.4.3.4 Address resolution

The IPv4 layer will present the convergence sub-layer with an IPv4 packet to be transferred. The convergence sub-layer is responsible for determining which node the packet should be delivered to using the IPv4 addresses in the packet.

The IPv4 addresses are mapped to the layer 2 addresses in the following way (see also 7.5 for IPv4 address generation):

- An IPv4 unicast address of a slave node is mapped to an unicast Link Address (LA = 0 to 0xFFFD). The value of the Link Address is the value of the lower 16 bits of the IPv4 address;
- The IPv4 broadcast address 255.255.255.255 is mapped to the layer 2 broadcast address (LA = 0xFFFF). It is only used for broadcast transmission from a master node to the slave nodes that are registered with it.

After the IP address has been mapped to a corresponding layer 2 address the packet is sent. See also section 7.5.

7.4.3.5 Segmentation and reassembly

The Logical Link Control sub-layer will perform segmentation and reassembly, if the packet is too long. The coding details are defined in Annex B of CLC/TS 50590:2015. An example diagram can be found in 6.5.12 of CLC/TS 50590:2015.

7.4.4 IPv6 specific convergence sub-layer

7.4.4.1 Architecture

The IPv6 specific convergence sub-layer provides an efficient method for transferring IPv6 packets over the AMC-SS networks. The IPv6 PDUs may be compressed according to RFC 2507 with the predefined compression parameters specified in Annex A. The communication architecture for IPv6 is shown in Figure 9.

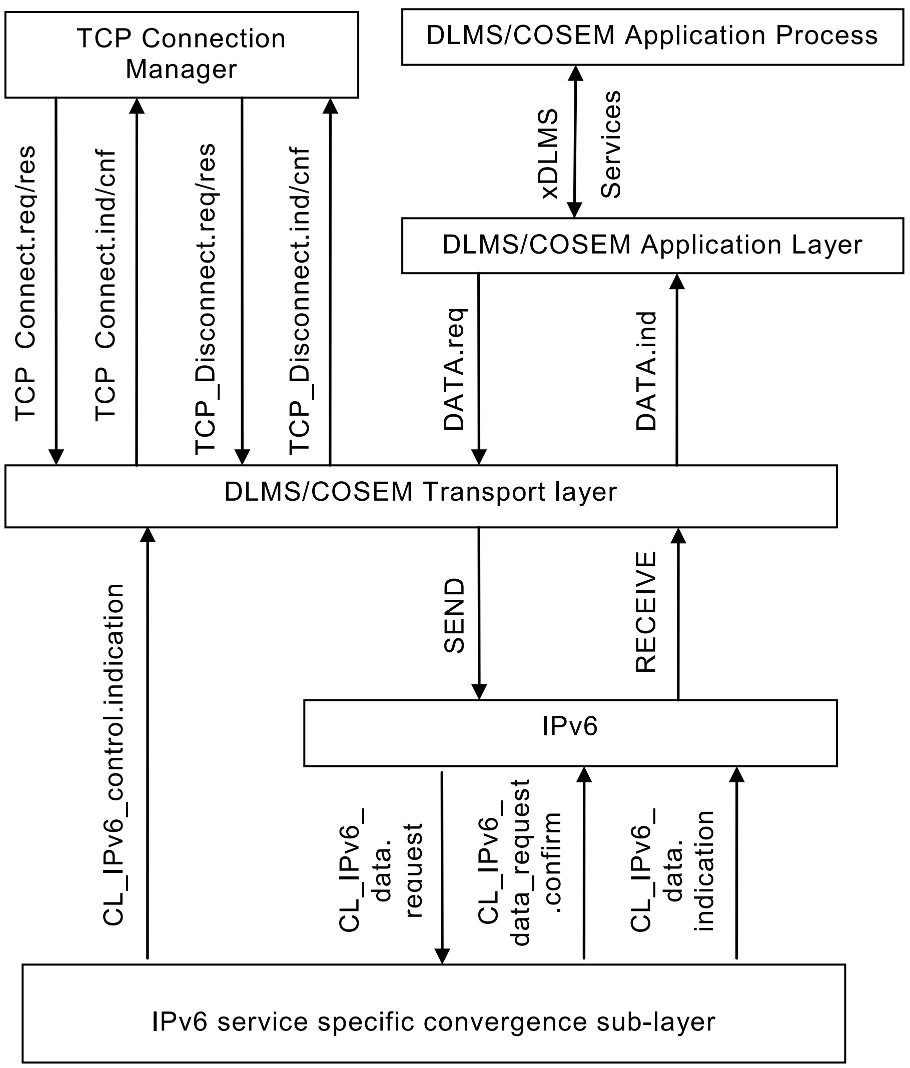

Figure 9 – The IPv6 communication architecture

7.4.4.2 Primitives

7.4.4.2.1 CL_IPv6_data.request

The CL_IPv6_data.request primitive is used by the IPv6 layer to initiate the data transmission process.

The semantics of the primitive are as follows:

CL_IPv6_data.request (Destination Address, SourceAddress, CompressionType, DataLength, Data)

The *DestinationAddress* parameter is an IPv6 address. It may have any values defined in 7.5.

The *SourceAddress* parameter is an IPv6 address. It may have IPv6 unicast address values defined in 7.5.

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX (see Annex D of CLC/TS 50590:2015). The *DataLength* may have any value in the range from 3 to (16 x (DBLMAX-3) – 1).

The *CompressionType* parameter specifies the type of a uncompressed or compressed IPv6 packet or of a CONTEXT_STATE packet to be transmitted. The parameter may have one of the following values:

- 0: data contains a regular, uncompressed, IPv6 datagram;
- 1: data contains an IPv6 datagram with the format FULL HEADER as specified in RFC 2507;
- 2: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_TCP as specified in RFC 2507;
- 3: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_TCP_NODELTA as specified in RFC 2507;
- 4: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_ NON_TCP as specified in RFC 2507;
- 5: data contains a CONTEXT_STATE message as specified in RFC 2507.

The *Data* parameter is the CL service data unit to be transmitted. It may contain an IPv6 packet or a CONTEXT_STATE packet as specified in RFC 2507.

The following Table 16 shows the mapping of the CL_IPv6_data.request primitive to the DL_data.request primitive in the master node.

Table 16 – Mapping of CL_IPv6_data.request to DL_data.request in the master node

CL_IPv6_data.request		DL_data.request	
Parameter	**Contents**	**Parameter**	**Contents**
		ServiceType	0, if the *Destination Address* contains an IPv6 unicast address; 1, if the *Destination Address* contains an IPv6 multicast address
SourceAddress	IPv6 unicast address of the master node	CellIdentificationNumber	CIN of the master node
DestinationAddress	IPv6 multicast address or IPv6 unicast address of a slave node registered with the master node	LinkAddress	LA = 0 to 0xFFFF = lower 16 bit of the *DestinationAddress*
		LinkChannelNumber	6
		DataPriority	0 to 15
DataLength	3 to 16 x (DBLMAX-3)-1	DataLength	DataLength value of CL_IPv6_data.request plus one. The maximal DataLength value is limited to 16 x (DBLMAX-3)
CompressionType	0 to 5	1st byte of Data	0 to 5
Data	IPv6 or CONTEXT_STATE packet	Data bytes 2 to DataLength	IPv6 or CONTEXT_STATE packet

The following Table 17 shows the mapping of the CL_IPv6_data.request primitive to the DL_data.request primitive in the slave node.

Table 17 – Mapping of CL_IPv6_data.request to DL_data.request in the slave node

CL_IPv6_data.request		DL_data.request	
Parameter	**Contents**	**Parameter**	**Contents**
		ServiceType	1
Destination Address	IPv6 unicast address of the master node	CellIdentificationNumber	CIN of the master node the slave is registered with
SourceAddress	IPv6 unicast address of the slave node	LinkAddress	LA = 0 to 0xFFFD= lower 16 bit of the *SourceAddress*
		LinkChannelNumber	5
		DataPriority	0 to 15
DataLength	3 to 16 x (DBLMAX-3)-1	DataLength	DataLength value of CL_IPv6_data.request plus one. The maximal DataLength value is limited to 16 x (DBLMAX-3)
CompressionType	0 to 5	1st byte of Data	0 to 5
Data	IPv6 or CONTEXT_STATE packet	Data bytes 2 to DataLength	IPv6 or CONTEXT_STATE packet

7.4.4.2.2 CL_IPv6_data.indication

The CL_data.indication primitive informs the IPv6 layer about the reception of data.

The semantics of the primitive are as follows:

CL_IPv6_data.indication (Destination Address, SourceAddress, CompressionType, DataLength, Data)

The *DestinationAddress* parameter is an IPv6 address. It may have any values defined in 7.5.

The *SourceAddress* parameter is an IPv6 address. It may have IPv6 unicast address values defined in 7.5.

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX (see Annex D of CLC/TS 50590:2015). The *DataLength* may have any value in the range from 3 to (16 x (DBLMAX-3) – 1).

The *CompressionType* parameter specifies the type of the received uncompressed or compressed IPv6 packet or CONTEXT_STATE packet. The parameter may have one of the following values:

0: data contains a regular, uncompressed, IPv6 datagram;

1: data contains an IPv6 datagram with the format FULL HEADER as specified in RFC 2507;

2: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_TCP as specified in RFC 2507;

3: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_TCP_NODELTA as specified in RFC 2507;

4: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_ NON_TCP as specified in RFC 2507;

5: data contains a CONTEXT_STATE message as specified in RFC 2507.

The *Data* parameter is the CL service data unit that was received. It may contain an IPv6 packet or a CONTEXT_STATE packet as specified in RFC 2507.

The following Table 18 shows the mapping of the DL_data.indication primitive to the CL_IPv6_data.indication primitive in the master node.

Table 18 – Mapping of DL_data.indication to CL_IPv6_data.indication in the master node

DL_data.indication		CL_IPv6_data.indication	
Parameter	**Contents**	**Parameter**	**Contents**
DataIdentifier	255		
CellIdentificationNumber	CIN	DestinationAddress	IPv6 unicast address of the master node
LinkAddress	LA = 0 to 0xFFFD	SourceAddress	IPv6 unicast address of the slave node
LinkChannelNumber	6		
DataPriority	0 to 15		
DataLength	4 to 16 x (DBLMAX-3)	DataLength	3 to 16 x (DBLMAX-3)-1 = DataLength value of DL_data.indication minus one.
1st byte of Data	0 to 5	CompressionType	0 to 5
Data bytes 2 to DataLength	IPv6 or CONTEXT_STATE packet	Data	IPv6 or CONTEXT_STATE packet

The following Table 19 shows the mapping of the DL_data.indication primitive to the CL_IPv6_data.indication primitive in the slave node.

Table 19 – Mapping of DL_data.indication to CL_IPv6_data.indication in the slave node

DL_data.indication		CL_IPv6_data.indication	
Parameter	**Contents**	**Parameter**	**Contents**
DataIdentifier	254, 255		
CellIdentificationNumber	CIN	SourceAddress	IPv6 unicast address of the master node
LinkAddress	LA = 0 to 0xFFFD, if DataIdentifier = 255; LA = 0xFFFF, if DataIdentifier = 254	DestinationAddress	IPv6 unicast address, if DataIdentifier=255; IPv6 multicast address, if if DataIdentifier=254
LinkChannelNumber	6		
DataPriority	0 to 15		
DataLength	4 to 16 x (DBLMAX-3)	DataLength	3 to 16 x (DBLMAX-3)-1 = DataLength value of DL_data.indication minus one.
1st byte of Data	0 to 5	CompressionType	0 to 5
Data bytes 2 to DataLength	IPv6 or CONTEXT_STATE packet	Data	IPv6 or CONTEXT_STATE packet

7.4.4.2.3 CL_IPv6_data_request.confirm

The CL_IPv6_data_request.confirm primitive is a local response to the IPv6 layer entity, which indicates the local success or failure of the associated previous CL_IPv6_data.request primitive.

The semantics of the primitive are as follows:

CL_IPv6_data_request.confirm (Status, Destination Address, CompressionType, DataLength, Data)

The *Status* parameter is used to pass the status information back to the requesting IPv6 layer entity. It is used to indicate the success or failure of the associated previous CL_IPv6_data.request primitive. The parameter may have one of the following values:

0: success of the associated previous CL_IPv6_data.request;

1: failure of the associated previous CL_IPv6_data.request. The transmit buffer of the requested entity is occupied;

2: failure of the associated previous CL_IPv6_data.request. There is no communication to the addressed destination node. The node registration is lost;

3: failure of the associated previous CL_IPv6_data.request. The communication to the addressed destination node is temporarily lost.

The *DestinationAddress* parameter is an IPv6 address. It may have any values defined in 7.5.

The *DataLength* parameter is the length of the *Data* parameter in octets. It depends on the value of the parameter DBLMAX (see Annex D of CLC/TS 50590:2015). The *DataLength* may have any value in the range from 3 to (16 x (DBLMAX-3) – 1).

The *CompressionType* parameter specifies the type of the uncompressed or compressed IPv6 packet or CONTEXT_STATE packet. The parameter may have one of the following values:

0: data contains a regular, uncompressed, IPv6 datagram;

1: data contains an IPv6 datagram with the format FULL HEADER as specified in RFC 2507;

2: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_TCP as specified in RFC 2507;

3: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_TCP_NODELTA as specified in RFC 2507;

4: data contains an IPv6 datagram with a compressed header with the format COMPRESSED_ NON_TCP as specified in RFC 2507;

5: data contains a CONTEXT_STATE message as specified in RFC 2507.

The *Data* parameter is the CL service data unit that was contained in the associated previous CL_IPv6_data.request primitive. It may contain an IPv6 packet or a CONTEXT_STATE packet as specified in RFC 2507

The following Table 20 shows the mapping of the DL_data_identifier.confirm primitive to the CL_IPv6_data_request.confirm primitive in the master node.

Table 20 – Mapping of DL_data_identifier.confirm to CL_IPv6_data_request.confirm in the master node

DL_data_identifier.confirm		**CL_IPv6_data_request.confirm**	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 3	Status	0 to 3
DataIdentifier	254 or 255, if Status = 0; Not used, if Status = 1 to3.		
LinkAddress	LA = 0 to 0xFFFD, if DataIdentifier = 255; LA = 0xFFFF, if DataIdentifier = 254	DestinationAddress	IPv6 unicast address, if DataIdentifier=255; IPv6 multicast address, if if DataIdentifier=254
LinkChannelNumber	6		
DataPriority	0 to 15		
DataLength	4 to 16 x (DBLMAX-3)	DataLength	3 to 16 x (DBLMAX-3)-1 = DataLength value of DL_data.identifier.confirm minus one.
1st byte of Data	0 to 5	CompressionType	0 to 5
Data bytes 2 to DataLength	IPv6 or CONTEXT_STATE packet	Data	IPv6 or CONTEXT_STATE packet

The following Table 21 shows the mapping of the DL_data_identifier.confirm primitive to the CL_IPv6_data_request.confirm primitive in the slave node.

Table 21 – Mapping of DL_data_identifier.confirm to CL_IPv6_data_request.confirm in the slave node

DL_data_identifier.confirm		CL_IPv6_data_request.confirm	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 3	Status	0 to 3
DataIdentifier	255, if Status = 0; Not used, if Status = 1 to 3.		
CellIdentificationNumber	CIN	DestinationAddress	IPv6 unicast address of the master node
LinkChannelNumber	6		
DataPriority	0 to 15		
DataLength	4 to 16 x (DBLMAX-3)	DataLength	3 to 16 x (DBLMAX-3)-1 = DataLength value of DL_data.identifier.confirm minus one.
1st byte of Data	0 to 5	CompressionType	0 to 5
Data bytes 2 to DataLength	IPv6 or CONTEXT_STATE packet	Data	IPv6 or CONTEXT_STATE packet

7.4.4.3 Address resolution

The IPv6 layer will present the convergence sub-layer with an IPv6 packet to be transferred. The convergence sub-layer is responsible for determining which node the packet should be delivered to using the IPv6 addresses in the packet.

The IPv6 addresses are mapped to the layer 2 addresses in the following way (see also 7.5 for IPv6 address generation):

- An IPv6 unicast address of a slave node is mapped to an unicast Link Address (LA = 0 to 0xFFFD). The value of the Link Address is the value of the lower 16 bits of the IPv6 address;
- A multicast IPv6 address is mapped to the layer 2 broadcast address (LA = 0xFFFF). It is only used for broadcast transmission from a master node to the slave nodes that are registered with it. The value of the master node CIN is contained in the bits 16 to 31 of the IPv6 address.

After the IP address has been mapped to a corresponding layer 2 address the packet is sent.

7.4.4.4 Segmentation and reassembly

The Logical Link Control sub-layer will perform segmentation and reassembly, if the packet is too long. The coding details are defined in Annex B of CLC/TS 50590:2015. An example diagram can be found in 6.5.12 of CLC/TS 50590:2015.

7.5 Identification and addressing

A slave node is identified by its serial number (device identifier, DID), which is the ‘Multivendor Identification Number’ based on DIN 43864-5 (cf. DID definition in CLC/TS 50590:2015) and which contains the FLAG ID of DLMS. During the registration process the ‘Multivendor Identification Number’ is provided to the master node and the Channel Identification Number (CIN) and the Link Address (LA) are assigned to the slave node. In addition the Network Identification Number (NIN) is provided to the slave node. The registration process is specified in 7.2 of CLC/TS 50590:2015. The addresses and the identification numbers are specified in 6.2 of CLC/TS 50590:2015. Using LA, CIN and NIN a master node and the slave node registered with it are able to generate the same IPv4 or IPv6 address as defined below, without necessity to exchange further information with each other.

If an IPv4 unicast address is used by a master node to address a slave node registered with it, then the IPv4 address has the following value:

IPv4-Address = 10.0.xx.yy,

xx = LA_{high}, the higher byte of the Link Address (LA) of the slave node;
yy = LA_{low}, the lower byte of the Link Address (LA) of the slave node.

The IPv4 unicast address 10.0.255.255 is used by a slave node to address the master node, it is registered with.

The IPv4 broadcast address 255.255.255.255 is used by a master node to address all slave nodes registered with it.

If a unicast IPv6 address is used to address a slave node, then the IPv6 address has the following value:

IPv6-Address = FE80:0000:0000:0000:0000:xxxx:yyyy:zzzz,

xxxx = N_NIN, the Network Identification Number of the slave node;
yyyy = CIN, the Channel Identification Number;
zzzz = LA = 0 to 0xFFFD, layer 2 unicast address of the slave node.

If a unicast IPv6 address is used to address a master node, then the IPv6 address has the following value:

IPv6-Address = FE80:0000:0000:0000:0000:xxxx:yyyy:FFFF,

xxxx = N_NIN, the Network Identification Number of the master node;
yyyy = CIN, the Channel Identification Number of the master node.

If a multicast IPv6 address is used to address all slave node registered with a master node, then the IPv6 address has the following value:

IPv6-Address = FF18:0000:0000:0000:0000:xxxx:yyyy:aaaa,

xxxx = N_NIN, the Network Identification Number of the network node;
yyyy = CIN, the Channel Identification Number of the master node;
aaaa = LA = 0xFFFF, layer 2 broadcast address of the salve node.

The master node maintains a table, which contains the device identifier (DID) and the allocated Link Address (LA) and the corresponding IP address of each registered slave node.

8 Upper layers (Session, Presentation, Application)

8.1 Overview

The DLMS/COSEM Application layer as specified in EN 62056-5-3 is used as the application layer. It provides services to the DLMS/COSEM application process (AP).

For the establishment of Application Associations the COSEM-OPEN and COSEM-RELEASE services may be used or the Application Associations may be pre-established. It is possible to handle several Application Associations simultaneously because the slave nodes registered with a master node are always connected with it (see 6.5.10 and 6.5.11 of CLC/TS 50590:2015). The maximal number of slave nodes registered with the master node is limited by the parameter of the master node DLLCNMAX defined in the Annex D of CLC/TS 50590:2015.

The services provided for the data transfer are specified in EN 62056-5-3 and include[GK3]:

- GET;
- SET;
- ACTION;
- EventNotification;
- DataNotification[GK4];
- Read;
- Write;
- UnconfirmedWrite;

- InformationReport.

The DLMS/COSEM Application layer services are mapped into the CL data services. In the case of the DLMS/COSEM AL specific convergence layer the CL_data.request and CL_data.indication primitives are used. In the case of the IPv4 specific convergence layer the CL_IPv4_data.request and CL_IPv4_data.indication primitives are used. In the case of the IPv6 specific convergence layer the CL_IPv6_data.request and CL_IPv6_data.indication primitives are used. In the case of IPv4 and IPv6 the Application layer is not directly connected to the Convergence layer, because the TCP / UDP layers are in between.

Each invocation of the CL_data.request, CL_IPv4_data.request or CL_IPv6_data.request services by higher layer is always confirmed by CL using the local primitives CL_data_request.confirm, CL_IPv4_data_request.confirm or CL_IPv6_data_request.confirm respectively (cf. 7.1 and Figure 10).

8.2 Mapping of application layer services

The Figure 10 below shows how the DLMS/COSEM Application layer services are mapped into the CL data services.

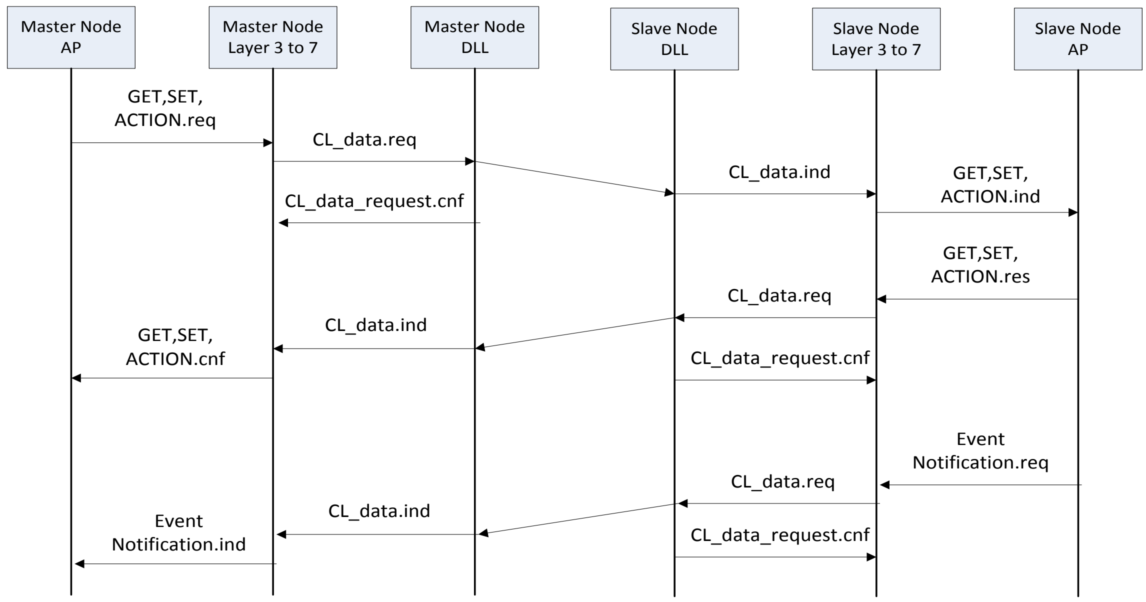

Figure 10 – Mapping of the DLMS/COSEM Application layer services

In the master node the data link layer receives from the higher layer the GET/SET/ACTION service request within a CL_data.request primitive or CL_IPv4_data.request primitive (in the case of IPv4) or CL_IPv6_data.request primitive (in the case of IPv6), which contains the serial number of the slave node as the destination address or in the case of IP the IP-destination-address of the slave node (inside the IP packet). The convergence layer maps the destination address or IP-destination-address to the corresponding layer 2 address, which is the Link Address (LA) of the addressed slave node and sends the service request to the slave node.

In the slave node the data link layer forwards the GET/SET/ACTION service request within a CL_data.indication primitive or CL_IPv4_data.indication primitive (in the case of IPv4) or CL_IPv6_data.indication primitive (in the case of IPv6) to the higher layer.

If the data link layer of the slave node receives from the higher layer the response to the GET/SET/ACTION service request within a CL_data.request primitive or CL_IPv4_data.request primitive (in the case of IPv4) or CL_IPv6_data.request primitive (in the case of IPv6), then it sends the response to the master node, In the master node the data link layer provides to the higher layer the response to the GET/SET/ACTION service request within a CL_data.indication primitive or CL_IPv4_data.indication primitive (in the case of IPv4) or CL_IPv6_data.indication primitive (in the case of IPv6), which in the non IP case contains the serial number of the slave node as the source address, which is inserted by the convergence layer, based on the received layer 2 address. In the case of IP the complete IP packet is provided, without any additional parameter.

In the slave node the data link layer receives from the higher layer the EventNotification service within a CL_data.request primitive or CL_IPv4_data.request primitive (in the case of IPv4) or CL_IPv6_data.request primitive (in the case of IPv6) and sends it to the master node,

In the master node the data link layer provides to the higher layer the EventNotification service within a CL_data.indication or CL_IPv4_data.indication primitive (in the case of IPv4) or CL_IPv6_data.indication primitive (in the case of IPv6) primitive, which in the non IP case contains the serial number of the slave node as the source address, which is inserted by the convergence layer, based on the received layer 2 address. In the case of IP the complete IP packet is provided, without any additional parameter.

8.3 Registration services

8.3.1 Overview

When a new (preconfigured) slave node is installed, it will register with a master node. The registration process is specified in 7.2 of CLC/TS 50590:2015. The CL of the master node and the slave node indicate each successful registration of the slave node to the corresponding higher layer entities. The master node monitors the connections to all registered nodes by polling them using request-respond services of the LLC sub-layer (see 6.5.10 and 7 of CLC/TS 50590:2015). If a registered slave node is not reachable for a certain amount of time (see parameter S_DLLCTO in Annex D of CLC/TS 50590:2015) or it becomes deregistered, this is also indicated to the corresponding higher layer entities.

8.3.2 CL_control.indication

The data link layer informs the higher layer with a CL_control.indication primitive about the registration or deregistration of a slave node.

The semantics of the primitive are as follows:

CL_control.indication (Status, DeviceIdentifier, SourceAddress)

The Status parameter indicates the registration status or the reachability of a slave node. The parameter can have one of the following values:

0: the slave node is registered with the master node;

1: the slave node is no longer registered with the master node;

2: communication between the slave node and the master node is temporarily lost.

The DeviceIdentifier parameter contains the serial number (device identifier, DID) of the slave node.

The Source*Address* parameter specifies the channel identification number (CIN) of the master node, the slave node has been registered with.

The following Table 22 shows the mapping of the DL_control.indication primitive to the CL_control.indication primitive.

Table 22 – Mapping of DL_control.indication to CL_control.indication

DL_control.indication		CL_control.indication	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 2	Status	0 to 2
DeviceIdentifier	Serial number	DeviceIdentifier	Serial number
CellIdentificationNumber	CIN	SourceAddress	CIN
LinkAddress	0 to 0xFFFD		
LinkChannelMap	Bitmap of 16 bit		
MulticastAddressTotalNumber	0 to 31		
MulticastAddressArray	Array with up to 31 multicast addresses		

The following Figure 11 shows the primitive exchange for the registration process in the case of the DLMS/COSEM AL.

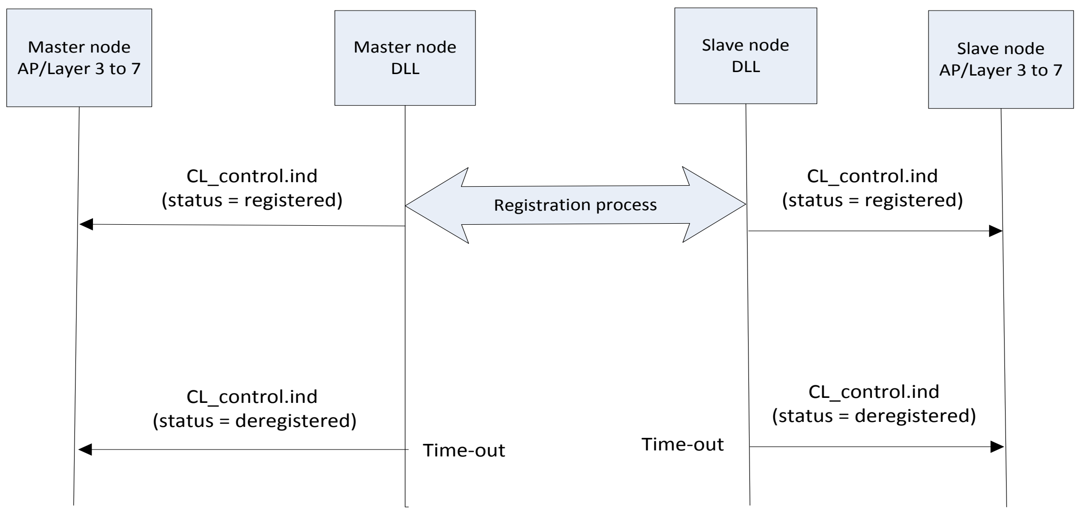

Figure 11 – Registration services specific for DLMS/COSEM AL

8.3.3 CL_IPv4_control.indication

The data link layer informs the IPv4 layer with a CL_IPv4_control.indication primitive about the registration or deregistration of a slave node.

The semantics of the primitive are as follows:

CL_IPv4_control.indication (Status, DeviceIdentifier, CellIdentificationNumber, IPv4Address)

The Status parameter indicates the registration status or the reachability of a slave node. The parameter can have one of the following values:

0: the slave node is registered with the master node;

1: the slave node is no longer registered with the master node;

2: communication between the slave node and the master node is temporarily lost.

The DeviceIdentifier parameter contains the serial number (device identifier, DID) of the slave node.

The *CellIdentificationNumber* parameter specifies the channel identification number (CIN) of the master node, the slave node has been registered with.

The IPv4Address parameter contains the IPv4 unicast address of the slave node (see 7.5).

The following Table 23 shows the mapping of the DL_control.indication primitive to the CL_IPv4_control.indication primitive.

Table 23 – Mapping of DL_control.indication to CL_IPv4_control.indication

DL_control.indication		CL_IPv4_control.indication	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 2	Status	0 to 2
DeviceIdentifier	Serial number	DeviceIdentifier	Serial number
CellIdentificationNumber	CIN	CellIdentificationNumber	CIN
LinkAddress	0 to 0xFFFD	IPv4Address	IPv4 unicast address of the slave node as defined in the sub-section 7.5.
LinkChannelMap	Bitmap of 16 bit		
MulticastAddressTotalNumber	0 to 31		
MulticastAddressArray	Array with up to 31 multicast addresses		

The following Figure 12 shows the primitive exchange for the registration process in the case of IPv4.

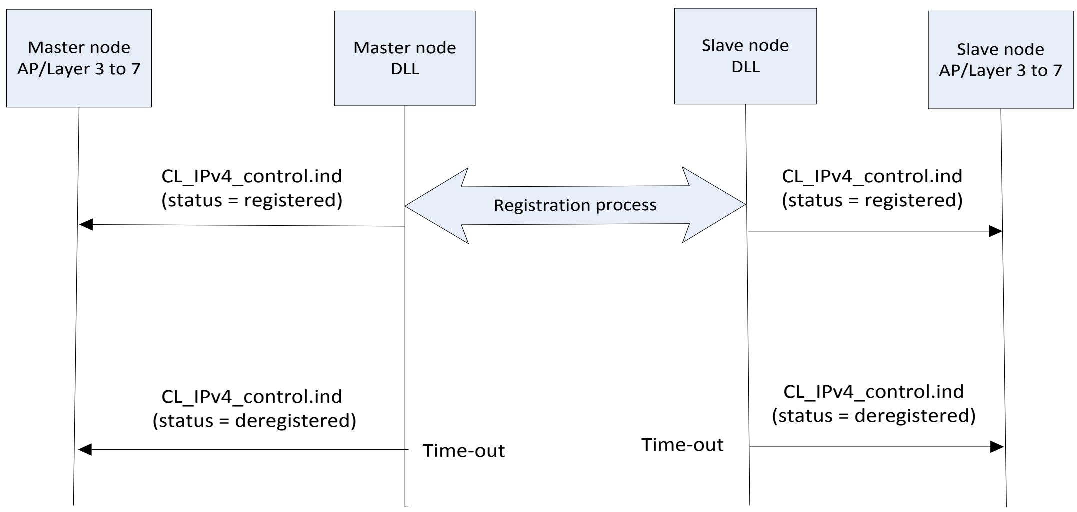

Figure 12 – IPv4 registration services

8.3.4 CL_IPv6_control.indication

The data link layer informs the IPv6 layer with a CL_IPv6_control.indication primitive about the registration or deregistration of a slave node.

The semantics of the primitive are as follows:

CL_IPv6_control.indication (Status, DeviceIdentifier, CellIdentificationNumber, IPv6Address)

The Status parameter indicates the registration status or the reachability of a slave node. The parameter can have one of the following values:

0: the slave node is registered with the master node;

1: the slave node is no longer registered with the master node;

2: communication between the slave node and the master node is temporarily lost.

The DeviceIdentifier parameter contains the serial number (device identifier, DID) of the slave node.

The *CellIdentificationNumber* parameter specifies the channel identification number (CIN) of the master node, the slave node has been registered with.

The IPv6Address parameter contains the IPv6 unicast address of the slave node (see 7.5).

The following Table 24 shows the mapping of the DL_control.indication primitive to the CL_IPv6_control.indication primitive.

Table 24 – Mapping of DL_control.indication to CL_IPv6_control.indication

DL_control.indication		CL_IPv6_control.indication	
Parameter	**Contents**	**Parameter**	**Contents**
Status	0 to 2	Status	0 to 2
DeviceIdentifier	Serial number	DeviceIdentifier	Serial number
CellIdentificationNumber	CIN	CellIdentificationNumber	CIN
LinkAddress	0 to 0xFFFD	IPv6Address	IPv6 unicast address of the slave node as defined in 7.5.
LinkChannelMap	Bitmap of 16 bit		
MulticastAddressTotalNumber	0 to 31		
MulticastAddressArray	Array with up to 31 multicast addresses		

The following Figure 13 shows the primitive exchange for the registration process in the case of IPv6.

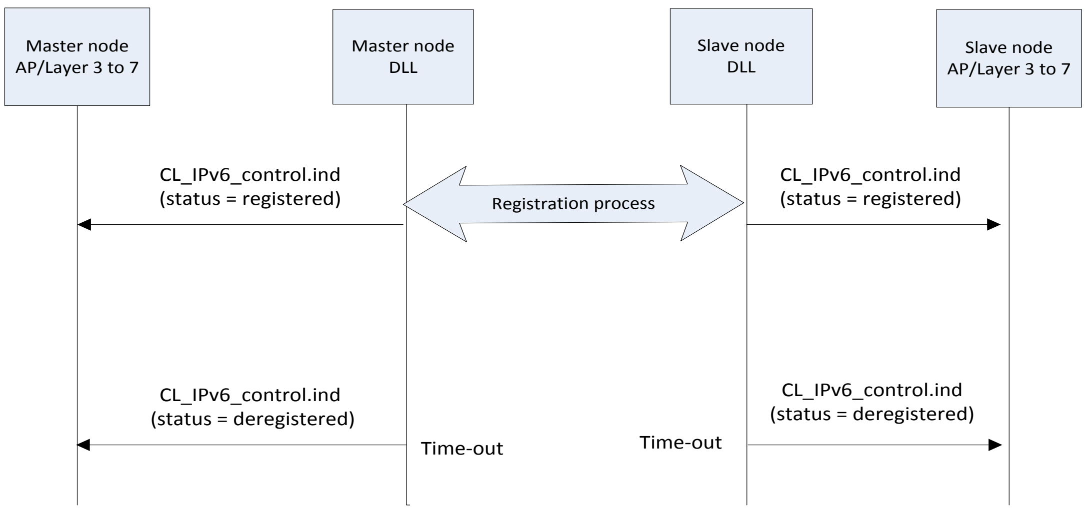

Figure 13 – IPv6 registration services

9 Data Model

The COSEM data model as specified in EN 62056-6-1 and EN 62056-6-2 shall be used as the data model.

Annex A
(normative)
IP header compression parameters.

The header of IPv4 or IPv6 PDUs may be compressed according to RFC 2507 with the following default configuration parameters partly defined in RFC 2507:

F_MAX_PERIOD = 256 : Largest number of compressed non-TCP headers that may be sent without sending a full header. DEFAULT is 256

F_MAX_TIME = 180 : Compressed headers may not be sent more than F_MAX_TIME seconds after sending last full header

MAX_HEADER = 168 : The largest header size in octets that may be compressed.

TCP_SPACE = 15 : Maximum CID value for TCP.

NON_TCP_SPACE =15 : Maximum CID value for non-TCP.

EXPECT_REORDERING = no, i. e. : the mechanisms in section 11 of RFC 2507 are not used.

Annex B
(informative)
Data exchange examples

B.1 Data retrieval from slave node

This example shows the information exchange when a master node retrieves data from a slave node using the GET command.

With the 'Node enquiry' message the master node continuously polls the registered slave nodes. The 'Data enquiry' message is used by the master node to request the data of a dedicated link channel from the addressed slave node. The coding of the messages is defined in Annex B of CLC/TS 50590:2015.

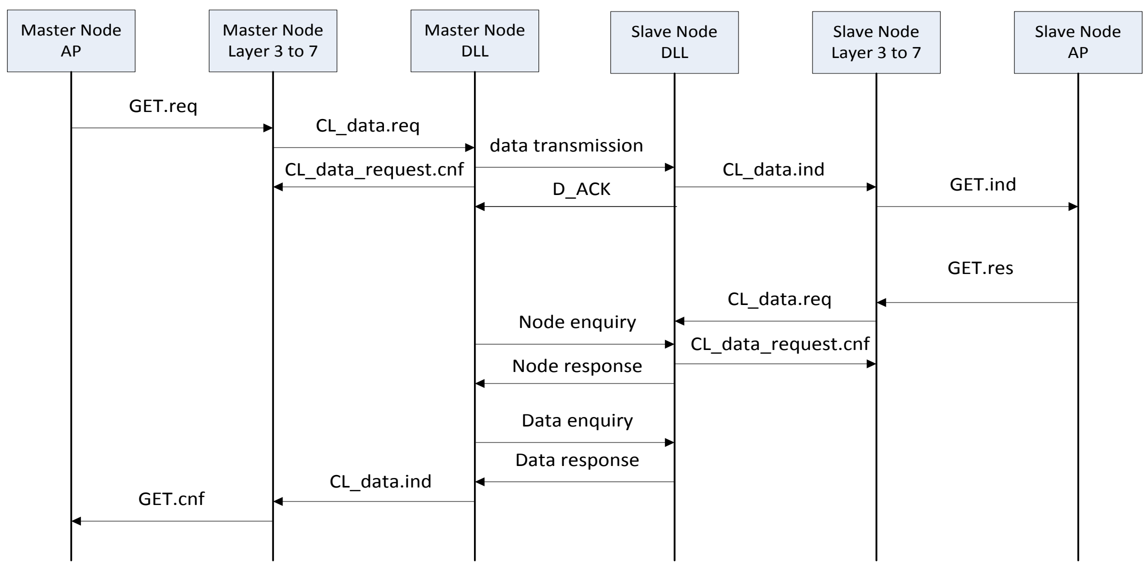

Figure B 1 – Successful GET command specific for DLMS/COSEM AL

The Figure B 1 above shows the primitives for the DLMS/COSEM AL specific case.

For the case of IPv4 the CL_IPv4_data.req primitive is used instead of the CL_data.req primitive, the CL_IPv4_data.ind primitive is used instead of the CL_data.ind primitive and the CL_IPv4_data_request.cnf primitive is used instead of the CL_data_request.cnf primitive. The Application layer is not directly connected to the Convergence layer, because the TCP / UDP layers are in between.

For the case of IPv6 the CL_IPv6_data.req primitive is used instead of the CL_data.req primitive, the CL_IPv6_data.ind primitive is used instead of the CL_data.ind primitive and the CL_IPv6_data_request.cnf primitive is used instead of the CL_data_request.cnf primitive. The Application layer is not directly connected to the Convergence layer, because the TCP / UDP layers are in between.

B.2 Data retrieval error case

This example shows the information exchange when a master node tries to retrieve data from a slave node and no response is received. With the 'Node enquiry' message the master node continuously polls the registered slave nodes. The coding of the 'Node enquiry' message is defined in Annex B of CLC/TS 50590:2015. The master node repeats the 'Node enquiry' message until a time out is reached. The status of the slave node is reported to the higher layer with a CL_control.indication primitive.

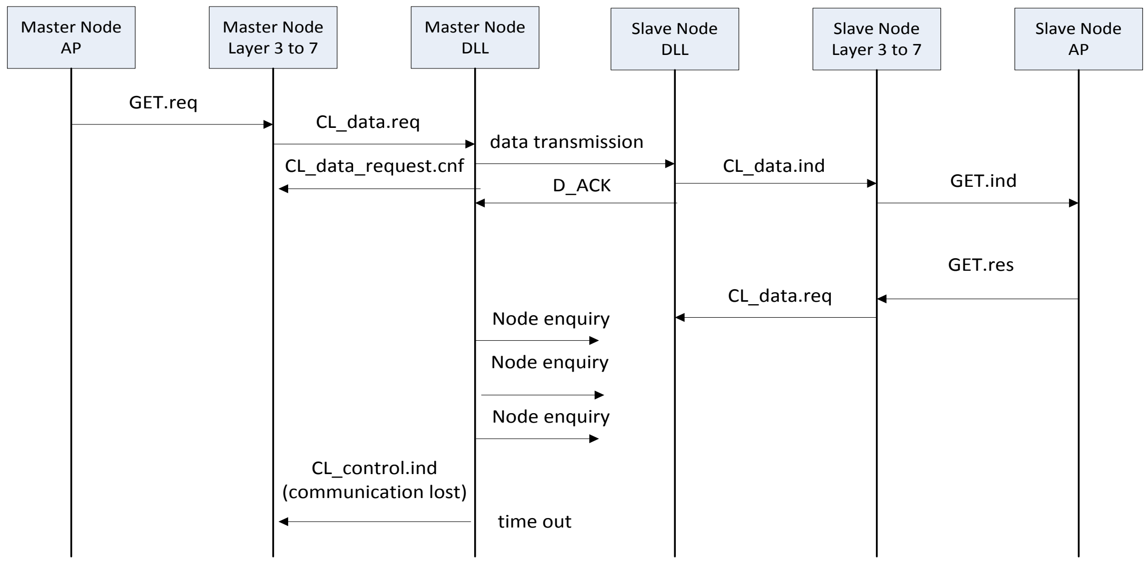

Figure B 2 – Data retrieval error case specific for DLMS/COSEM AL

The Figure B 2 above shows the primitives for the DLMS/COSEM AL specific case.

For the case of IPv4 the CL_IPv4_data.req primitive is used instead of the CL_data.req primitive, the CL_IPv4_data.ind primitive is used instead of the CL_data.ind primitive and the CL_IPv4_data_request.cnf primitive is used instead of the CL_data_request.cnf primitive. The Application layer is not directly connected to the Convergence layer, because the TCP / UDP layers are in between.

For the case of IPv6 the CL_IPv6_data.req primitive is used instead of the CL_data.req primitive, the CL_IPv6_data.ind primitive is used instead of the CL_data.ind primitive and the CL_IPv6_data_request.cnf primitive is used instead of the CL_data_request.cnf primitive. The Application layer is not directly connected to the Convergence layer, because the TCP / UDP layers are in between.

Bibliography

STD0005 – Internet Protocol
Author: J. Postel
Date: September 1981
Also: RFC0791, RFC0792, RFC0919, RFC0922, RFC0950, RFC1112
Available from: http://www.rfc-editor.org/rfc/pdfrfc/rfc791.txt.pdf

STD0006 – User Datagram Protocol
Author: J. Postel
Date: 28 August 1980
Also: RFC0768
Available from: http://www.rfc-editor.org/rfc/pdfrfc/rfc768.txt.pdf

STD0007 – Transmission Control Protocol
Author: J. Postel
Date: September 1981
Also: RFC0793
Available from: http://www.rfc-editor.org/rfc/pdfrfc/rfc793.txt.pdf

RFC 2460 - Internet Protocol, Version 6 (Ipv6) Specification. Authors: S. Deering, R. Hinden. December 1998. Available from: http://www.ietf.org/rfc/rfc2460

RFC 2464 - Transmission of Ipv6 Packets over Ethernet Networks. Authors: M. Crawford Fermilab. December 1998. Available from http://tools.ietf.org/html/rfc2464

RFC 4291 - IP Version 6 Addressing Architecture. Authors: R. Hinden, S. Deering, February 2006. Available from http://tools.ietf.org/html/rfc4291

This page deliberately left blank

NO COPYING WITHOUT BSI PERMISSION EXCEPT AS PERMITTED BY COPYRIGHT LAW

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email bsmusales@bsigroup.com.

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Copyright

All the data, software and documentation set out in all British Standards and other BSI publications are the property of and copyrighted by BSI, or some person or entity that owns copyright in the information used (such as the international standardization bodies) and has formally licensed such information to BSI for commercial publication and use. Except as permitted under the Copyright, Designs and Patents Act 1988 no extract may be reproduced, stored in a retrieval system or transmitted in any form or by any means – electronic, photocopying, recording or otherwise – without prior written permission from BSI. Details and advice can be obtained from the Copyright & Licensing Department.

Useful Contacts:

Customer Services
Tel: +44 845 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 845 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

bsi.

...making excellence a habit.™