

# INTERNATIONAL STANDARD

## NORME INTERNATIONALE



**Electricity metering data exchange – the DLMS/COSEM suite –  
Part 8-4: Communication profiles for narrow-band OFDM PLC PRIME  
neighbourhood networks**

**Échange des données de comptage de l'électricité – la suite DLMS/COSEM –  
Partie 8-4: Profils de communication pour réseaux de voisinage OFDM PLC  
PRIME à bande étroite**



## THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2018 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

IEC Central Office  
3, rue de Varembe  
CH-1211 Geneva 20  
Switzerland

Tel.: +41 22 919 02 11  
[info@iec.ch](mailto:info@iec.ch)  
[www.iec.ch](http://www.iec.ch)

### About the IEC

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

### About IEC publications

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

#### IEC Catalogue - [webstore.iec.ch/catalogue](http://webstore.iec.ch/catalogue)

The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

#### IEC publications search - [webstore.iec.ch/advsearchform](http://webstore.iec.ch/advsearchform)

The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,...). It also gives information on projects, replaced and withdrawn publications.

#### IEC Just Published - [webstore.iec.ch/justpublished](http://webstore.iec.ch/justpublished)

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

#### Electropedia - [www.electropedia.org](http://www.electropedia.org)

The world's leading online dictionary of electronic and electrical terms containing 21 000 terms and definitions in English and French, with equivalent terms in 16 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

#### IEC Glossary - [std.iec.ch/glossary](http://std.iec.ch/glossary)

67 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

#### IEC Customer Service Centre - [webstore.iec.ch/csc](http://webstore.iec.ch/csc)

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: [sales@iec.ch](mailto:sales@iec.ch).

### A propos de l'IEC

La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

### A propos des publications IEC

Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

#### Catalogue IEC - [webstore.iec.ch/catalogue](http://webstore.iec.ch/catalogue)

Application autonome pour consulter tous les renseignements bibliographiques sur les Normes internationales, Spécifications techniques, Rapports techniques et autres documents de l'IEC. Disponible pour PC, Mac OS, tablettes Android et iPad.

#### Recherche de publications IEC - [webstore.iec.ch/advsearchform](http://webstore.iec.ch/advsearchform)

La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études,...). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

#### IEC Just Published - [webstore.iec.ch/justpublished](http://webstore.iec.ch/justpublished)

Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

#### Electropedia - [www.electropedia.org](http://www.electropedia.org)

Le premier dictionnaire en ligne de termes électroniques et électriques. Il contient 21 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans 16 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

#### Glossaire IEC - [std.iec.ch/glossary](http://std.iec.ch/glossary)

67 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.

#### Service Clients - [webstore.iec.ch/csc](http://webstore.iec.ch/csc)

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: [sales@iec.ch](mailto:sales@iec.ch).

# INTERNATIONAL STANDARD

## NORME INTERNATIONALE



**Electricity metering data exchange – the DLMS/COSEM suite –  
Part 8-4: Communication profiles for narrow-band OFDM PLC PRIME  
neighbourhood networks**

**Échange des données de comptage de l'électricité – la suite DLMS/COSEM –  
Partie 8-4: Profils de communication pour réseaux de voisinage OFDM PLC  
PRIME à bande étroite**

INTERNATIONAL  
ELECTROTECHNICAL  
COMMISSION

COMMISSION  
ELECTROTECHNIQUE  
INTERNATIONALE

ICS 17.220.20; 35.110.01; 91.140.50

ISBN 978-2-8322-6334-1

**Warning! Make sure that you obtained this publication from an authorized distributor.  
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.**

## CONTENTS

|                                                                                     |    |
|-------------------------------------------------------------------------------------|----|
| FOREWORD.....                                                                       | 5  |
| INTRODUCTION.....                                                                   | 7  |
| 1 Scope.....                                                                        | 8  |
| 2 Normative references .....                                                        | 8  |
| 3 Terms, definitions and abbreviated terms .....                                    | 10 |
| 3.1 Terms and definitions.....                                                      | 10 |
| 3.2 Abbreviated terms.....                                                          | 10 |
| 4 Targeted communication environments.....                                          | 12 |
| 5 Use of the communication layers for this profile.....                             | 13 |
| 5.1 Information related to the use of the standard specifying the lower layers..... | 13 |
| 5.2 The structure of the communication profiles .....                               | 13 |
| 5.2.1 Overview .....                                                                | 13 |
| 5.2.2 The IEC 61334-4-32 profile.....                                               | 14 |
| 5.2.3 The TCP-UDP/IPv4 profile.....                                                 | 14 |
| 5.2.4 The TCP-UDP/IPv6 profile.....                                                 | 14 |
| 5.3 Lower protocol layers and their use.....                                        | 14 |
| 5.3.1 General .....                                                                 | 14 |
| 5.3.2 Physical layer .....                                                          | 14 |
| 5.3.3 MAC layer.....                                                                | 15 |
| 5.4 Service mapping and convergence layers .....                                    | 17 |
| 5.4.1 Overview .....                                                                | 17 |
| 5.4.2 The IEC 61334-4-32 LLC SSCS.....                                              | 17 |
| 5.4.3 The IPv4 SSCS.....                                                            | 19 |
| 5.4.4 The IPv6 SSCS.....                                                            | 21 |
| 5.5 Registration and connection management.....                                     | 22 |
| 5.5.1 Overview .....                                                                | 22 |
| 5.5.2 IEC 61334-4-32 profile.....                                                   | 22 |
| 5.5.3 TCP-UDP/IPv4 profile .....                                                    | 27 |
| 5.5.4 TCP-UDP/IPv6 profile .....                                                    | 40 |
| 6 Identification and addressing schemes .....                                       | 55 |
| 6.1 IEC 61334-4-32 profile addressing.....                                          | 55 |
| 6.1.1 Overview .....                                                                | 55 |
| 6.1.2 MAC address.....                                                              | 56 |
| 6.1.3 IEC 61334-4-32 SSCS addresses .....                                           | 56 |
| 6.1.4 LLC addresses .....                                                           | 56 |
| 6.2 TCP-UDP/IPv4 profile addressing .....                                           | 57 |
| 6.3 TCP-UDP/IPv6 profile addressing .....                                           | 57 |
| 7 Specific consideration for the application layer services .....                   | 57 |
| 7.1 Overview.....                                                                   | 57 |
| 7.2 Application Association (AA) establishment and release: ACSE services .....     | 57 |
| 7.2.1 AA establishment: IEC 61334-4-32 profile.....                                 | 57 |
| 7.2.2 AA establishment: IP based profile .....                                      | 58 |
| 7.2.3 Application association release.....                                          | 59 |
| 7.3 xDLMS services .....                                                            | 59 |
| 7.4 Security mechanisms .....                                                       | 59 |
| 7.4.1 DLMS/COSEM security.....                                                      | 59 |

|              |                                                                                         |    |
|--------------|-----------------------------------------------------------------------------------------|----|
| 7.4.2        | Lower layers security .....                                                             | 59 |
| 7.5          | Transferring long application messages .....                                            | 59 |
| 7.6          | Media access, bandwidth and timing considerations .....                                 | 60 |
| 7.7          | Other considerations .....                                                              | 60 |
| 8            | Communication configuration and management .....                                        | 60 |
| 9            | The COSEM application process .....                                                     | 60 |
| 10           | Additional considerations for the use of this profile .....                             | 60 |
| Annex A      | (informative) Examples .....                                                            | 61 |
| A.1          | Data exchange between two IP communication peers .....                                  | 61 |
| A.2          | Joining a multicast group .....                                                         | 63 |
| A.3          | PRIME encoding examples .....                                                           | 63 |
| Annex B      | (normative) New COSEM interfaces classes and OBIS codes .....                           | 76 |
| Annex C      | (informative) IEC 61334-4-32 profile: Error cases during connection establishment ..... | 77 |
| Annex D      | (normative) Convergence layer constants .....                                           | 78 |
| Bibliography | .....                                                                                   | 79 |
| Figure 1     | – Communication architecture .....                                                      | 12 |
| Figure 2     | – OFDM PLC PRIME communication profile architectures .....                              | 14 |
| Figure 3     | – IEC 61334-4-32 SSCS services .....                                                    | 18 |
| Figure 4     | – MSC for Data services in the case of logical name referencing .....                   | 19 |
| Figure 5     | – IEC 61334-4-32 SSCS .....                                                             | 22 |
| Figure 6     | – MSC for IEC 61334-4-32 SSCS services .....                                            | 26 |
| Figure 7     | – IPv4 SSCS services .....                                                              | 28 |
| Figure A.1   | – MSC of IPv4 SSCS services .....                                                       | 62 |
| Figure A.2   | – Joining MSC IPv4 profile .....                                                        | 63 |
| Figure C.1   | – Error cases during connection establishment .....                                     | 77 |
| Table 1      | – Result values for SSCS services .....                                                 | 25 |
| Table 2      | – AR_REGISTER_S message format .....                                                    | 36 |
| Table 3      | – AR_REGISTER_B message format .....                                                    | 36 |
| Table 4      | – AR_UNREGISTER_S message format .....                                                  | 36 |
| Table 5      | – AR_MCAST_REG_S message format .....                                                   | 36 |
| Table 6      | – AR_MCAST_REG_B message format .....                                                   | 37 |
| Table 7      | – AR_MCAST_UNREG_S message format .....                                                 | 37 |
| Table 8      | – AR_MCAST_UNREG_B message format .....                                                 | 37 |
| Table 9      | – AR_LOOKUP_S message format .....                                                      | 38 |
| Table 10     | – AR_LOOKUP_B message format .....                                                      | 38 |
| Table 11     | – IPv4 packet format without header compression negotiated .....                        | 38 |
| Table 12     | – IPv4 packet format with VJ header compression .....                                   | 39 |
| Table 13     | – Connection data sent by the initiator .....                                           | 39 |
| Table 14     | – Connection data sent by the responder .....                                           | 40 |
| Table 15     | – IPv6 SSCS table entry .....                                                           | 44 |
| Table 16     | – Mapping IPv6 precedence to PRIME MAC priority .....                                   | 45 |

|                                                                                           |    |
|-------------------------------------------------------------------------------------------|----|
| Table 17 – AR_REGISTERv6_S message format.....                                            | 51 |
| Table 18 – AR_REGISTERv6_B message format.....                                            | 51 |
| Table 19 – AR_UNREGISTERv6_S message format.....                                          | 51 |
| Table 20 – AR_UNREGISTERv6_B message format.....                                          | 52 |
| Table 21 – AR_LOOKUPv6_S message format.....                                              | 52 |
| Table 22 – AR_LOOKUPv6_B message format.....                                              | 52 |
| Table 23 – AR_MCAST_REGv6_S message format.....                                           | 53 |
| Table 24 – AR_MCAST_REGv6_B message format.....                                           | 53 |
| Table 25 – AR_MCAST_UNREGv6_B message format.....                                         | 53 |
| Table 26 – IPv6 Packet format without negotiated header compression.....                  | 54 |
| Table 27 – UDP/IPv6 Packet format with LOWPAN_IPHC header compression and LOWPAN_NHC..... | 54 |
| Table 28 – IPv6 Packet format with LOWPAN_IPHC negotiated header compression.....         | 54 |
| Table 29 – IPv6 Connection signalling data sent by the initiator.....                     | 55 |
| Table 30 – IPv6 Connection signalling data sent by the responder.....                     | 55 |
| Table 31 – Client service access point values.....                                        | 57 |
| Table 32 – Server service access point values.....                                        | 57 |
| Table 33 – Application associations and data exchange in the IEC 61334-4-32 profile.....  | 58 |
| Table D.1 – TYPE value assignment.....                                                    | 78 |

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## INTERNATIONAL ELECTROTECHNICAL COMMISSION

**ELECTRICITY METERING DATA EXCHANGE –  
THE DLMS/COSEM SUITE –****Part 8-4: Communication profiles for narrow-band  
OFDM PLC PRIME neighbourhood networks**

## FOREWORD

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

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this International Standard may involve the use of a maintenance service concerning the stack of protocols on which the present standard IEC 62056-8-4 is based.

The IEC takes no position concerning the evidence, validity and scope of this maintenance service.

The provider of the maintenance service has assured the IEC that he is willing to provide services under reasonable and non-discriminatory terms and conditions for applicants throughout the world. In this respect, the statement of the provider of the maintenance service is registered with the IEC. Information may be obtained from:

PRIME Alliance  
2-12, Avenue de la Renaissance  
1000, Brussels/ (BE)  
[www.prime-alliance.org.com](http://www.prime-alliance.org.com)

International Standard IEC 62056-8-4 has been prepared by IEC technical committee 13:  
Electrical energy measurement and control.

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

|             |                  |
|-------------|------------------|
| CDV         | Report on voting |
| 13/1749/CDV | 13/1763/RVC      |

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

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

A list of all parts in the IEC 62056 series, published under the general title *Electricity metering data exchange – The DLMS/COSEM suite*, can be found on the IEC website.

The committee has decided that the contents of this document will remain unchanged until the stability date indicated on the IEC website under "<http://webstore.iec.ch>" in the data related to the specific document. At this date, the document will be

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

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

## INTRODUCTION

As defined in IEC 62056-1-0, the IEC 62056 DLMS/COSEM suite provides specific communication profile standards for communication media relevant for smart metering.

Such communication profile standards specify how the COSEM data model and the DLMS/COSEM application layer can be used on the lower, communication media-specific protocol layers.

Communication profile standards refer to communication standards that are part of the IEC 62056 DLMS/COSEM suite or to any other open communication standard.

This International Standard specifies DLMS/COSEM communication profiles using Recommendation ITU-T G.9904:2012 *Narrow-band orthogonal frequency division multiplexing power line communication transceivers for PRIME networks*. It applies for devices installed on the neighbourhood network.

It follows the rules defined in IEC 62056-5-3:2017, Annex A, and in IEC 62056-1-0 and the IEC TS 62056-1-1 recommendations for its structure.

The communication profile specified in this document is based on the results of the European OPEN Meter project, Topic Energy 2008.7.1.1, Project no.: 226369, [www.openmeter.com](http://www.openmeter.com), and has been prepared by the PRIME Alliance Technical Working Group, [www.prime-alliance.org](http://www.prime-alliance.org).

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## **ELECTRICITY METERING DATA EXCHANGE – THE DLMS/COSEM SUITE –**

### **Part 8-4: Communication profiles for narrow-band OFDM PLC PRIME neighbourhood networks**

#### **1 Scope**

This part of IEC 62056 specifies DLMS/COSEM communication profiles for narrow-band OFDM power line carrier PRIME neighbourhood networks using the modulation as specified in Recommendation ITU-T G.9904:2012.

Three communication profiles are specified:

- a profile using the IEC 61334-4-32 LLC layer;
- a profile using TCP-UDP/IPv4;
- a profile using TCP-UDP/IPv6.

#### **2 Normative references**

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 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 61334-4-511:2000, *Distribution automation using distribution line carrier systems – Part 4-511: Data communication protocols – Systems management – CIASE protocol*

IEC 62056-1-0, *Electricity metering data exchange – The DLMS/COSEM suite – Part 1-0: Smart metering standardization framework*

IEC TS 62056-1-1, *Electricity metering data exchange – The DLMS/COSEM suite – Part 1-1: Template for DLMS/COSEM communication profile standards*

IEC 62056-4-7:2015, *Electricity metering data exchange – The DLMS/COSEM suite – Part 4-7: DLMS/COSEM transport layer for IP networks*

IEC 62056-5-3:2017, *Electricity metering data exchange – The DLMS/COSEM suite – Part 5-3: DLMS/COSEM application layer*

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

IEC 62056-6-2:2017, *Electricity metering data exchange – The DLMS/COSEM suite – Part 6-2: COSEM interface classes*

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

Recommendation ITU-T G.9904:2012, *Series G: Transmission systems and media, digital systems and networks Access networks – In premises networks. Narrowband orthogonal frequency division multiplexing power line communication transceivers for PRIME networks*

STD0005 – Internet Protocol

Author: J. Postel

Date: September 1981

Also: RFC0791, RFC0792, RFC0919, RFC0922, RFC0950, RFC1112

Available from: <http://www.ietf.org/rfc/rfc0791.txt>

STD0006 – User Datagram Protocol

Author: J. Postel

Date: 28 August 1980

Also: RFC 768

Available from: <http://www.ietf.org/rfc/rfc0768.txt>

STD0007 – Transmission Control Protocol

Author: J. Postel

Date: September 1981

Available from: <http://www.ietf.org/rfc/rfc0793.txt>

RFC 1144, Compressing TCP/IP Headers for Low Speed serial Link

Author: V. Jacobson

Date: February 1990

Available from: <https://tools.ietf.org/rfc/rfc1144.txt>

RFC 2460, Internet Protocol, Version 6 (IPv6) Specification

Authors: S. Deering, Cisco, R. Hinden Nokia

Date: December 1998

Available from: <http://www.ietf.org/rfc/rfc2460.txt>

RFC 2464, Transmission of IPv6 Packets over Ethernet Networks

Authors M. Crawford Fermilab

Date: December 1998

Available from: <http://www.ietf.org/rfc/rfc2464.txt>

RFC 3315, Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

Authors R. Droms, E. J. Bound, B. Volz, T. Lemon, C. Perkins, M. Carney

Date: July 2003

Available from: [www.ietf.org/rfc/rfc3315.txt](http://www.ietf.org/rfc/rfc3315.txt)

RFC 4291, IP Version 6 Addressing Architecture

Authors R. Hinden Nokia, S. Deering Cisco Systems

Date: February 2006.

Available from: <http://www.ietf.org/rfc/rfc4291.txt>

RFC 4862, IPv6 Stateless Address Configuration

Authors S. Thomson, Cisco, T. Narten IBM, T. Jinmei, Toshiba

Date: September 2007.

Available from: [www.ietf.org/rfc/rfc4862.txt](http://www.ietf.org/rfc/rfc4862.txt)

RFC 6282, Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks

Authors J. Hui, Ed. Arch Rock Corporation P. Thubert Cisco

Date: September 2011.

Available from: <http://www.ietf.org/rfc/rfc6282.txt>

### 3 Terms, definitions and abbreviated terms

#### 3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

NOTE See also Recommendation ITU-T G.9904:2012.

##### 3.1.1

##### **base node**

master node which controls and manages the resources of a subnetwork

##### 3.1.2

##### **registration**

process by which a service node is accepted as member of the subnetwork and allocated with an LNID

##### 3.1.3

##### **service node**

any one node of a subnetwork which is not a base node

##### 3.1.4

##### **unregistration**

process by which a service node leaves a subnetwork

#### 3.2 Abbreviated terms

| Abbreviation | Meaning                                             |
|--------------|-----------------------------------------------------|
| AA           | Application Association                             |
| AARE         | Application Association Response                    |
| AARQ         | Application Association Request                     |
| ACSE         | Application Control Service Element                 |
| AL           | Application Layer                                   |
| AP           | Application Process                                 |
| APDU         | Application Protocol Data Unit                      |
| ARQ          | Automatic Repeat Request                            |
| CL           | Convergence Layer                                   |
| .cnf         | Confirm service primitive                           |
| COSEM        | Companion Specification for Energy Metering         |
| CPCS         | Common Part Convergence Sublayer                    |
| CSMA/CA      | Carrier Sense Multiple Access – Collision Avoidance |
| D8PSK        | Differential Eight-Phase Shift Keying               |
| DBPSK        | Differential Binary Phase Shift Keying              |
| DGW          | Default Gateway                                     |

| Abbreviation | Meaning                                    |
|--------------|--------------------------------------------|
| DHCP         | Dynamic Host Configuration Protocol        |
| DLMS         | Device Language Message Specification      |
| DQPSK        | Differential Quaternary Phase Shift Keying |
| EUI-48       | 48-bit Extended Unique Identifier          |
| FU           | Firmware Upgrade                           |
| FW           | Firmware                                   |
| IANA         | Internet Assigned Numbers Authority        |
| IGMP         | Internet Group Management Protocol         |
| .ind         | Indication service primitive               |
| IP           | Internet Protocol                          |
| IPv4         | Internet Protocol, version 4               |
| IPv6         | Internet Protocol version 6                |
| LCID         | Local Connection Identifier                |
| LD           | Logical Device                             |
| LLC          | Logical Link Control (sub-layer)           |
| LNID         | Local Node Identifier                      |
| MAC          | Medium Access Control, MAC sublayer entity |
| MLME         | MAC Layer Management Entity                |
| MPDU         | MAC Protocol Data Unit                     |
| NAT          | Network Address Translation                |
| NHC          | Next Header Compression                    |
| NL           | Noise Level                                |
| OBIS         | OBject Identification System               |
| OFDM         | Orthogonal Frequency Division Multiplexing |
| OSI          | Open System Interconnection                |
| PHY          | Physical Layer entity                      |
| PLC          | Power Line Communication                   |
| PIB          | PLC Information Base                       |
| PLME         | Physical Layer Management Entity           |
| PPDU         | PHY Protocol Data Unit                     |
| .req         | Request service primitive                  |
| RFC          | Request For Comment                        |
| .rsp         | Response service primitive                 |
| SDU          | Service Data Unit                          |
| SID          | Switch Identifier                          |
| SNA          | Subnetwork Address                         |
| SNR          | Signal-to-Noise Ratio                      |
| SSCS         | Service Specific Convergence Sublayer      |

| Abbreviation | Meaning                                   |
|--------------|-------------------------------------------|
| TCP          | Transmission Control Protocol             |
| TOS          | Type Of Service                           |
| UDP          | User Datagram Protocol                    |
| xDLMS_ASE    | extended DLMS Application Service Element |
| ZCT          | Zero Crossing Time                        |

## 4 Targeted communication environments

The DLMS/COSEM communication profiles for narrow-band OFDM PLC PRIME neighbourhood networks are intended for remote data exchange on Neighbourhood Networks (NN) between Neighbourhood Network Access Points (NNAPs) and Local Network Access Points (LNAPs) or End Devices using OFDM PLC technology over the low voltage electricity distribution network as a communication medium. The functional reference architecture is shown Figure 1.

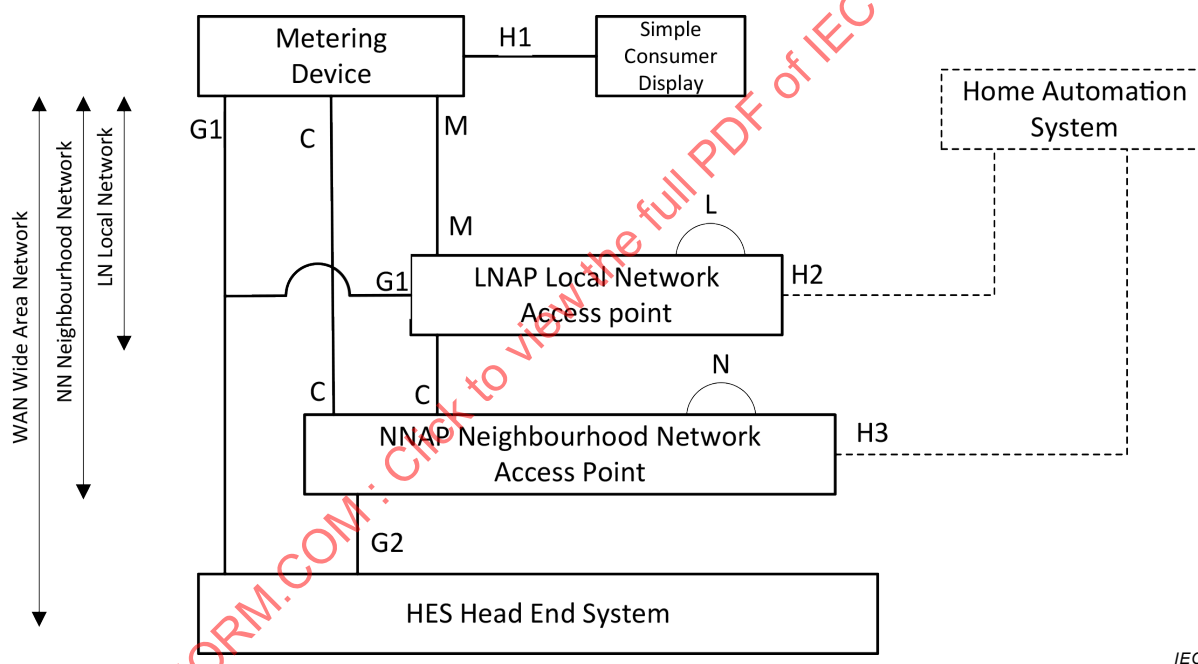


**Figure 1 – Communication architecture**

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 base node while end devices and LNAPs act as service nodes.

NNAPs and similarly LNAPs may communicate to each other, but this is out of the scope of this document, 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.

A mixed architecture is also possible, i.e. both the HES and the NNAP can act as a client.

## 5 Use of the communication layers for this profile

### 5.1 Information related to the use of the standard specifying the lower layers

Recommendation ITU-T G.9904:2012 defines PHY and MAC layers for power line communication using OFDM. At the top of the MAC layer it defines also a number of Convergence layers, the purpose of which is to match the lower protocol layers with the higher protocol layers.

This document makes use of the entire part of the PHY and MAC layers. Additionally, it defines – in 5.5 – the use of the respective Convergence layers.

### 5.2 The structure of the communication profiles

#### 5.2.1 Overview

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

- the DLMS/COSEM Application layer as specified in IEC 62056-5-3 covering the Application, Presentation and Session functionalities;
- the LLC sublayer as specified in IEC 61334-4-32, used with the DLMS/COSEM 61334-4-32 profile over PRIME networks;
- the DLMS/COSEM transport layer for IP networks as specified in IEC 62056-4-7:2015 used with the DLMS/COSEM TCP-UDP/IPv4 and TCP-UDP/IPv6 profiles over PRIME networks;
- the PRIME MAC layer, the CPCS and the corresponding SSCS, according to the selected profile (IEC 61334-4-32, TCP-UDP/IPv4 or TCP-UDP/IPv6);
- the PRIME Physical layer.

Following this reference model, three distinct profiles can be identified, all of them using the PRIME PHY, MAC layers as lower layers and the Common Part Convergence Sublayer on one hand, and the DLMS/COSEM Application layer specified in IEC 62056-5-3 and the COSEM object model specified in IEC 62056-6-1 and IEC 62056-6-2 on the other hand. Lower layers – meaning PHY and MAC – are based on the principles of IEEE 802.15.4.

NOTE The COSEM interface classes for setting up and managing data exchange over narrow-band OFDM PLC PRIME network are specified in IEC 62056-6-2.

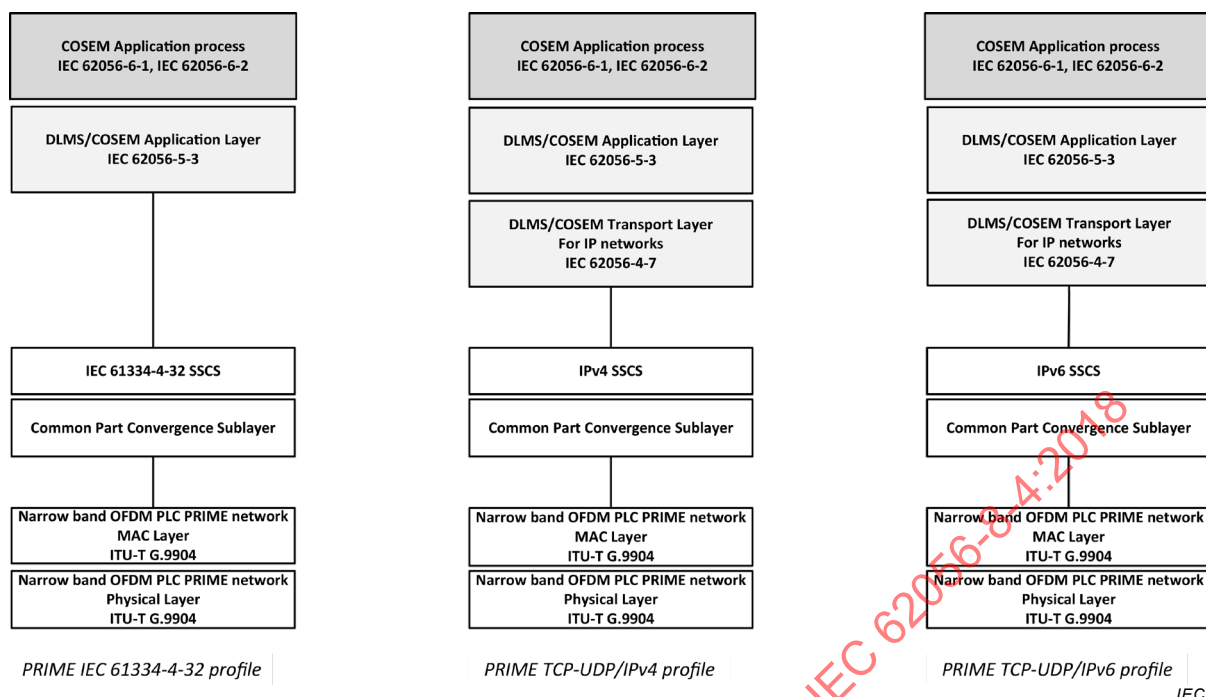


**Figure 2 – OFDM PLC PRIME communication profile architectures**

### 5.2.2 The IEC 61334-4-32 profile

The IEC 61334-4-32 profile uses the PRIME IEC 61334-4-32 SCS making the necessary adaptation between the PRIME PHY and MAC layers and the DLMS/COSEM application layer.

### 5.2.3 The TCP-UDP/IPv4 profile

The TCP-UDP/IPv4 profile uses the PRIME IPv4 SCS making the necessary adaptation between the PRIME PHY and MAC layers and the IPv4 layer, supporting the DLMS/COSEM transport layer and application layer.

### 5.2.4 The TCP-UDP/IPv6 profile

The TCP-UDP/IPv6 profile uses the PRIME IPv6 SCS making the necessary adaptation between the PRIME PHY and MAC layers and the IPv6 layer, supporting the DLMS/COSEM transport layer and application layer.

## 5.3 Lower protocol layers and their use

### 5.3.1 General

All three profiles specified in this document share the same PHY and MAC layers.

### 5.3.2 Physical layer

#### 5.3.2.1 General

This layer provides the interface between the equipment and the physical transmission medium that is the electricity distribution network. It transmits and receives MPDUs between neighbour nodes.

### 5.3.2.2 PRIME PHY data plane services

PHY DATA services are generated / used by the MAC layer entity whenever data – PPDU – have to be transmitted to / received from (a) peer MAC entity(ies) using the PHY transmission procedures. See Recommendation ITU-T G.9904:2012, 7.10.2.

### 5.3.2.3 PRIME PHY control plane services

PRIME PHY control plane services are used to control the physical layer by the MAC layer. See Recommendation ITU-T G.9904:2012, 7.10.3. They are the following:

- PHY\_AGC: allows the MAC layer entity to set or get the Automatic Gain Mode of the PHY;
- PHY\_TIMER: allows the MAC layer entity to get the time at which the transmission has to be started;
- PHY\_CD: allows the MAC layer entity to look for the carrier detect signal, in order to detect if the physical medium is free;
- PHY\_NL: allows the MAC layer entity to get the floor noise level value present on the power line;
- PHY\_SNR: allows the MAC layer entity to get the value of the signal-to-noise ratio, in order to find the appropriate degree of robustness needed for data exchange;
- PHY\_ZCT: allows the MAC layer entity to get the zero crossing time of the mains and the time between the last transmission or reception and the zero crossing of the mains.

### 5.3.2.4 PRIME PHY management plane services

PRIME PHY management plane services are used to manage the physical layer by the MAC layer. See Recommendation ITU-T G.9904:2012, 7.10.4. They are the following:

- PLME\_RESET: allows the MAC layer entity to request the PHY layer to reset its present functional state. As a result of this primitive, the PHY should reset all internal states and flush all buffers to clear any queued receive or transmit data;
- PLME\_SLEEP: allows the MAC layer entity to request the PHY layer to suspend its present activities including all reception functions. The PHY layer should complete any pending transmission before entering into a sleep state;
- PLME\_RESUME: allows the MAC layer entity to request the PHY layer to resume its suspended activities. As a result of this primitive, the PHY layer should start its normal transmission and reception functions;
- PLME\_TESTMODE: allows the MAC layer entity to put the PHY layer into some non-default functional modes. Specific functional mode out of the various possible modes is provided as an input parameter. Following the reception of this primitive, the PHY layer should complete any pending transmissions in its buffer before entering the test mode requested;
- PLME\_GET: allows the MAC layer entity to query information about a given attribute of the PRIME Information Base.

## 5.3.3 MAC layer

### 5.3.3.1 Overview – main features and functions

A subnetwork can be seen as a tree structure with two types of nodes, the base node and service nodes. The base node is at the root of the tree and acts as the master node that provides the subnetwork with connectivity. There is one and only one base node in a subnetwork. Any other subnetwork node is a service node. Service nodes are either leaves or branch points (switches) of the tree structure.

The base node is initially the subnetwork itself and all other nodes should follow a registration process to enrol themselves on the subnetwork.

Service nodes start in a “Disconnected” state and they try to find a base node or a switch node to register themselves to the subnetwork. After this, they become leaves of the tree.

Service nodes may change their state dynamically from “Terminal” functions to “Switch” functions and vice-versa. These changes occur on the basis of certain pre-defined events on the network. Service nodes in “Switch” state become branch points of the tree, capable of switching their neighbours’ data to propagate connectivity.

The three functional states of service nodes are:

- **Disconnected:** all nodes are in this state initially or after a restart. In this state, a node is not capable to participate in the network. The primary function of a service node in this state is to search for an operational network in its proximity and try to register itself on it;
- **Terminal:** In this state a service node is part of the network and is capable of communicating its traffic by establishing connections, but it is not capable of switching the traffic of any other node; and finally:
- **Switch:** In this state a service node is capable of performing all “Terminal” functions. Additionally, it is capable of forwarding data to and from other devices on the subnetwork.

The events and associated processes that trigger changes from one functional state to another are registration, unregistration, promotion and demotion.

Other functions of the MAC layer are:

- address resolution and broadcast and multicast addressing;
- CSMA/CA algorithm implementation;
- promoting service nodes from “Terminal” state to “Switch” state or demoting them from the “Switch” state to “Terminal” state;
- establishing direct connections from one service node to another;
- packet aggregation;
- security functions, such as encryption and security keys management;
- PHY robustness management in order to select the best modulation schema for a given situation;
- ARQ mechanism.

#### 5.3.3.2 Services used by base node and service nodes

The MAC connection and data services used by base nodes and service nodes are the following (see Recommendation ITU-T G.9904:2012, 8.5.2 and 8.5.4:

- MAC\_ESTABLISH: is used to manage the connection establishment at MAC layer;
- MAC\_RELEASE: is used to release a connection at MAC layer;
- MAC\_JOIN: is used to join to a broadcast or multicast connection and allow the reception of such packets;
- MAC\_LEAVE: is used to leave a broadcast or multicast connection;
- MAC\_DATA: is used to send and receiving unicast, multicast or broadcast data.

#### 5.3.3.3 Management services

The MAC management services (see Recommendation ITU-T G.9904:2012, 8.5.5) are the following:

- MLME\_REGISTER: is used to perform registration and to indicate when registration has been performed;

- MLME\_UNREGISTER: is used to perform unregistration and to indicate when it has been performed;
- MLME\_PROMOTE: is used to perform promotion of service nodes from the “Terminal” state to “Switch” state and to indicate when it has been performed;
- MLME\_DEMOTE: is used to perform demotion of service nodes in the “Switch” state to “Terminal” state and to indicate when it has been performed;
- MLME\_RESET: is used to reset the MAC into a known “good” status. After the reset the MAC is in the disconnected functional state;
- MLME\_GET: is used to retrieve individual values from the MAC such as statistics;
- MLME\_LIST\_GET: is used to retrieve a list of values from the MAC;
- MLME\_SET: is used to set configuration values in the MAC.

## 5.4 Service mapping and convergence layers

### 5.4.1 Overview

NOTE 1 The following description is based on Recommendation ITU-T G.9904:2012, 9.1.

The convergence layer is divided into two sublayers:

The Common Part Convergence Sublayer (CPCS) provides a set of generic services. The use of CPCS services is optional in that a SSCS will configure into its protocol stack those services which are required from the CPCS and omit services that are not required. In the present specification, the CPCS provides the following services to the different SSCSs: segmentation and reassembly.

The Service Specific Convergence Sublayer (SSCS) contains services that are specific to a given communication profile. There are four SSCSs specified to connect the MAC layer to the upper layer:

- the IEC 61334-4-32 SSCS, see 5.4.2;
- the IPv4 SSCS, see 5.4.3;
- the IPv6 SSCS, see 5.4.4;
- the Null Convergence Sublayer.

NOTE 2 The Null Convergence Sublayer is outside the scope of this document. Information can be found in Recommendation ITU-T G.9904:2012, Clause 9.

### 5.4.2 The IEC 61334-4-32 LLC SSCS

#### 5.4.2.1 General

The IEC 61334-4-32 LLC convergence sublayer (SSCS) provides convergence functions for the communication profile using IEC 61334-4-32 LLC services. Implementations conforming to this SSCS shall offer all LLC services specified in this document. Additionally, the PRIME IEC 61334-4-32 SSCS shall provide services that help the mapping of the connection-less LLC protocol into the connection-oriented nature of PRIME MAC, meaning that before data exchange can take place, a connection has to be established. See Recommendation ITU-T G.9904:2012, 9.5.2.

Main features:

- a service node can only exchange data with the base node and not with other service nodes;
- each IEC 61334-4-32 SSCS session establishes a dedicated PRIME MAC connection for exchanging unicast data with the base node;

- the service node SSCS session is responsible for initiating this connection to the base node. The base node SSCS cannot initiate a connection to a service node;
- each IEC 61334-4-32 SSCS listens to a PRIME broadcast MAC connection dedicated to the transfer of IEC 61334-4-32 broadcast data from the base node to the service nodes. This broadcast connection is used when the base node application process using the IEC 61334-4-32 communication profile makes a transmission request with the destination address used for broadcast or when the broadcast SAP functions are used. When there are multiple SSCS sessions within a service Node, one PRIME broadcast MAC connection is shared by all the SSCS sessions.

The IEC 61334-4-32 SSCS services are shown in Figure 3 and specified in 5.4.2.2.

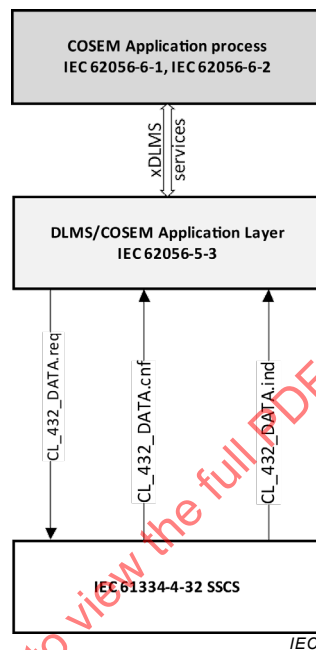


**Figure 3 – IEC 61334-4-32 SSCS services**

#### **5.4.2.2 CL\_432\_DATA services**

##### **5.4.2.2.1 General**

IEC 61334-4-32 specifies the following LLC layer data services:

- DL\_Data services;
- DL\_Broadcast services;
- DL\_Reply services; and
- DL\_Update\_Reply services.

In the DLMS/COSEM PRIME IEC 61334-4-32 profile, only the DL\_Data services are used. They are mapped to CL\_432\_DATA services. All other services are also mapped into CL\_432\_DATA services as defined below.

##### **5.4.2.2.2 CL\_432\_DATA services**

For DL\_Data data services, refer to IEC 61334-4-32. In this document, DL\_Data services are mapped into CL\_432\_DATA services.

xDLMS LN services are shown as an example here. PRIME does not impose any limitation on using any of the xDLMS services.

Figure 4 shows an example of the use of the CL\_432\_Data services.

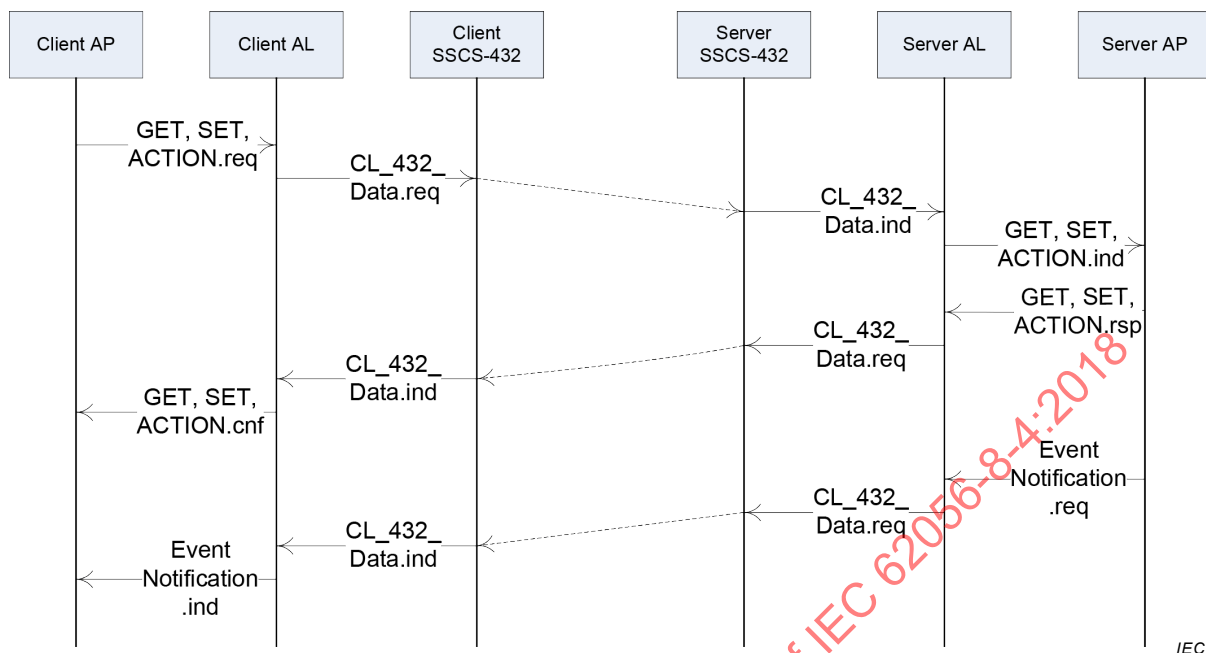


**Figure 4 – MSC for Data services in the case of logical name referencing**

#### 5.4.2.2.3 DL\_Broadcast services

The DL\_Broadcast.request service specified in IEC 61334-4-32 is not used in this profile. Instead, the CL\_432\_DATA.request primitive is used with a broadcast address as destination address.

#### 5.4.2.2.4 DL\_Reply and DL\_Update\_Reply services

In IEC 61334-4-32, the DL\_Reply and Update\_Reply services are used for unsolicited service management. In the DLMS/COSEM PRIME IEC 61334-4-32 profile, these services are replaced by CL\_432\_DATA services at the SSCS level.

### 5.4.3 The IPv4 SSCS

#### 5.4.3.1 Data services

##### 5.4.3.1.1 General

The following primitives are used for sending and receiving IPv4 packets.

##### 5.4.3.1.2 CL\_IPv4\_DATA.request

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.5.2.

##### Function

This primitive is passed from the IPv4 layer to the IPv4 SSCS. It contains one IPv4 packet to be sent.

##### Service parameters

The semantics of this primitive are as follows:

```
CL_IPv4_DATA.request(IPv4_PDU)
```

The IPv4\_PDU is the IPv4 packet to be sent.

#### *Use*

This primitive is used when an IPv4 packet has to be transmitted to the peer IPv4 layer.

#### **5.4.3.1.3 CL\_IPv4\_DATA.confirm**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.5.3.

#### *Function*

This primitive is passed from the IPv4 SSCS to the IPv4 layer. It contains a status indication and an IPv4 packet that has just been sent.

#### *Service parameters*

The semantics of this primitive are as follows:

```
CL_IPv4_DATA.confirm(IPv4_PDU, Result)
```

The IPv4\_PDU is the IPv4 packet that was to be sent.

The status is described by the field "Result". The Result value indicates whether the packet was sent or an error occurred. It takes a value from Table 1.

#### *Use*

This primitive is generated locally for signalling to the requestor the result of the data transfer request.

#### **5.4.3.1.4 CL\_IPv4\_DATA.indicate**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.5.4.

#### *Function*

This primitive is passed from the IPv4 SSCS to the IPv4 layer. It contains an IPv4 packet that has just been received.

#### *Service parameters*

The semantics of this primitive are as follows:

```
CL_IPv4_DATA.indicate(IPv4_PDU)
```

The IPv4\_PDU is the IPv4 packet that was received.

#### *Use*

Upon the arrival of a new IPv4 packet from the peer, the primitive is generated.

#### **5.4.4 The IPv6 SSCS**

##### **5.4.4.1 Data services**

###### **5.4.4.1.1 General**

The following primitives are used for sending and receiving IPv6 packets.

###### **5.4.4.1.2 CL\_IPv6\_DATA.request**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.6.9.5.2.

###### *Function*

This primitive is passed from the IPv6 layer to the IPv6 SSCS. It contains one IPv6 packet to be sent.

###### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv6\_DATA.request(IPv6\_PDU)

The IPv6\_PDU is the IPv6 packet to be sent.

###### *Use*

This primitive is used when an IPv6 packet has to be transmitted to the peer IPv6 layer.

###### **5.4.4.1.3 CL\_IPv6\_DATA.confirm**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.6.9.5.3.

###### *Function*

This primitive is passed from the IPv6 SSCS to the IPv6 layer. It contains a status indication and an IPv6 packet that has just been sent.

###### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv6\_DATA.confirm(IPv6\_PDU, Result)

The IPv6\_PDU is the IPv6 packet that was to be sent.

The Result value indicates whether the packet was sent or an error occurred. It takes a value from Table 1.

###### *Use*

This primitive is generated locally for signalling to the requestor the result of the data transfer request.

#### 5.4.4.1.4 CL\_IPv6\_DATA.indicate

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.6.9.5.4.

##### Function

This primitive is passed from the IPv6 SSCS to the IPv6 layer. It contains an IPv6 packet that has just been received.

##### Service parameters

The semantics of this primitive are as follows:

CL\_IPv6\_DATA.indicate(IPv6\_PDU)

The IPv6\_PDU is the IPv6 packet that was received.

##### Use

Upon the arrival of a new IPv6 packet from the peer, the primitive is generated.

### 5.5 Registration and connection management

#### 5.5.1 Overview

This subclause provides elements for device registration and connection management. All PRIME SSCSs are connection oriented requiring the establishment of a connection before data can be exchanged.

#### 5.5.2 IEC 61334-4-32 profile

##### 5.5.2.1 Connection management services

##### 5.5.2.1.1 Overview

The following services are used for IEC 61334-4-32 SSCS connection management; see Figure 5.

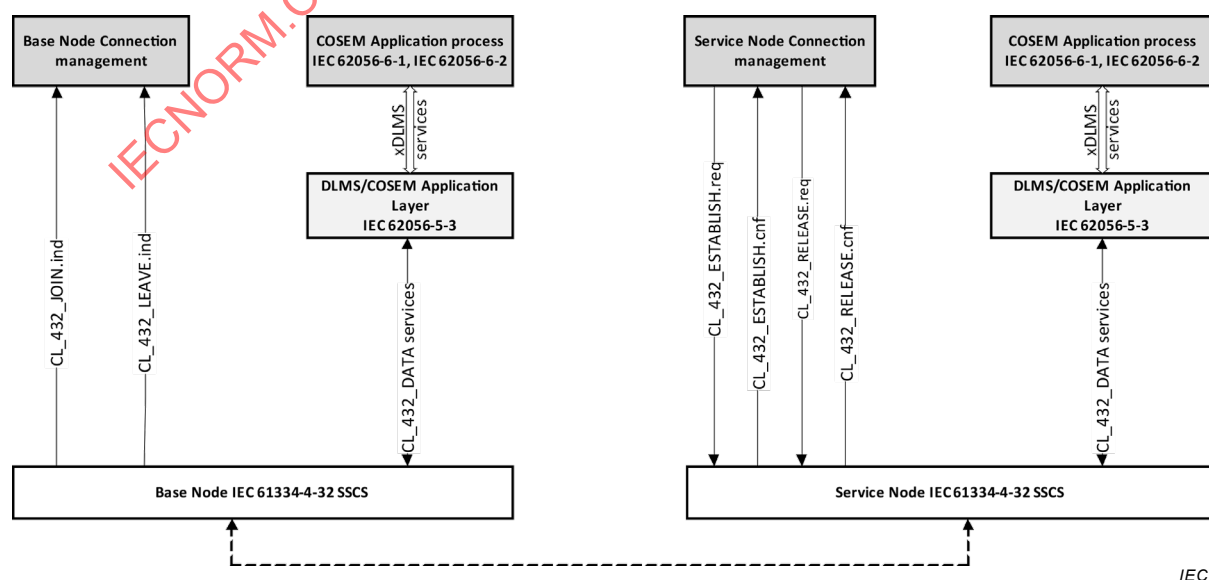


Figure 5 – IEC 61334-4-32 SSCS

#### 5.5.2.1.2 CL\_432\_ESTABLISH.request

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.5.6.1.1.

##### *Function*

The purpose of this service is to request a connection opening with the base node.

##### *Service parameters*

The semantics of the primitive is the following:

```
CL_432_ESTABLISH.request  
(  
    Device_Identifier  
)
```

Device\_Identifier is the serial number of the service node.

##### *Use*

This service primitive is used by the service node IEC 61334-4-32 Connection Manager to request a connection establishment with the base node.

#### 5.5.2.1.3 CL\_432\_ESTABLISH.confirm

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.5.6.1.2.

##### *Function*

The purpose of this service is to inform the service node IEC 61334-4-32 Connection Manager about the result of the previous CL\_432\_ESTABLISH.request.

##### *Service parameters*

The semantics of the primitive is the following:

```
CL_432_ESTABLISH.confirm  
(  
    Device_Identifier,  
    Destination_Address,  
    Base_Address  
)
```

Device\_Identifier is the serial number of the service node. It is the same value than the one carried by the CL\_432\_ESTABLISH.req primitive.

Destination\_Address is the address assigned by the base node to the concerned service node during the connection establishment process. Its scope is the subnetwork managed by the base node. During the registration process, the base node allocates a unique subnetwork destination address to the service nodes.

The Base\_Address is the address assigned to the base node.

##### *Use*

On receiving the connection establishment confirm primitive, the IEC 61334-4-32 SSCS session confirms to the service node IEC 61334-4-32 Connection Manager that the IEC 61334-4-32 SSCS session has been opened and it indicates the Destination\_Address allocated to the service node IEC 61334-4-32 SSCS session, as well as the address of the base node.

The base node IEC 61334-4-32 SSCS is responsible for allocating these addresses dynamically and associating the device identifier of the service node IEC 61334-4-32 SSCS session device with the allocated Destination\_Address, according to IEC 61334-4-511.

The service node also opens a MAC broadcast connection if no other IEC 61334-4-32 SSCS session has already opened such a broadcast connection. This connection is used to receive broadcast packets sent by the base node IEC 61334-4-32 SSCS to all service node IEC 61334-4-32 SSCS sessions.

In the case when the CL\_432\_ESTABLISH.request fails, a CL\_432\_Release.confirm primitive is sent back. The related reason of the failure is notified in the result.

#### **5.5.2.1.4 CL\_432\_JOIN.indicate**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.5.6.3.2.

##### *Function*

The purpose of this service is to inform the base node IEC 61334-4-32 Connection Manager about a new connection establishment.

##### *Service parameters*

The semantics of the service is the following:

```
CL_432_JOIN.indicate
(
    Device_Identifier,
    Destination_Address
)
```

Device\_Identifier is the serial number of the service node, which joined the PRIME PLC subnetwork.

Destination\_Address is the destination address provided by the base node SSCS to the service node which has requested the connection establishment.

See also 6.1.

##### *Use*

The base node IEC 61334-4-32 SSCS uses this service to inform its IEC 61334-4-32 Connection Manager that the concerned destination address is assigned to the service node identified by this Device\_Identifier.

#### **5.5.2.1.5 CL\_432\_RELEASE.request**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.5.6.1.3.

##### *Function*

This service primitive is passed from the service node IEC 61334-4-32 Connection Manager to the IEC 61334-4-32 SSCS. It is used to close the SSCS and to release any resource it may be holding.

*Service parameters*

The semantics of the service is the following:

```
CL_432_RELEASE.request
(
    Destination_Address
)
```

Destination\_Address is the destination address provided by the base node SSCS to the service node during the connection establishment.

*Use*

This primitive is used by the service node IEC 61334-4-32 Connection Manager to request the IEC 61334-4-32 SSCS to close the connection.

**5.5.2.1.6 CL\_432\_RELEASE.confirm**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.5.6.1.4.

*Function*

This service primitive is passed from the service node IEC 61334-4-32 SSCS to the IEC 61334-4-32 Connection Manager to inform it that the SSCS has been closed. This could be due to a previous CL\_432\_RELEASE.request or due to an error occurred, forcing the closure of the SSCS.

*Service parameters*

The semantics of this primitive are as follows:

```
CL_432_RELEASE.confirm
(
    Destination_Address,
    Result
)
```

Destination\_Address identifies the connection, which has been closed. The possible values of the Result parameter are specified in Table 1.

*Use*

This primitive is used by the service node IEC 61334-4-32 SSCS to inform the IEC 61334-4-32 Connection Manager about the result of the release request.

**Table 1 – Result values for SSCS services**

| Result             | Description                                                       |
|--------------------|-------------------------------------------------------------------|
| Success = 0        | The SSCS service was successfully performed.                      |
| Reject = 1         | The SSCS service failed because it was rejected by the base node. |
| Timeout = 2        | A time out occurs during the SSCS service processing.             |
| Not Registered = 6 | The service node is not currently registered to a subnetwork.     |

### 5.5.2.1.7 CL\_432\_LEAVE.indicate

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.5.6.3.3.

#### Function

This service primitive is used by the base node SSCS to inform the IEC 61334-4-32 Connection Manager that the service node identified by the Destination\_Address has left the subnetwork.

#### Service parameters

The semantics of this primitive are as follows:

```
CL_432_LEAVE.indicate
(
    Destination_Address
)
```

Destination\_Address is address of the service node, which left the subnetwork.

#### Use

The base node IEC 61334-4-32 SSCS uses this service to inform the IEC 61334-4-32 Connection Manager that the service node identified by the Destination\_Address has left the subnetwork.

### 5.5.2.2 Summary of the connection management services

The MSC of the service primitives exchanged between the IEC 61334-4-32 Connection Manager and the IEC 61334-4-32 SSCS is shown in Figure 6. Error cases are shown in Annex C.

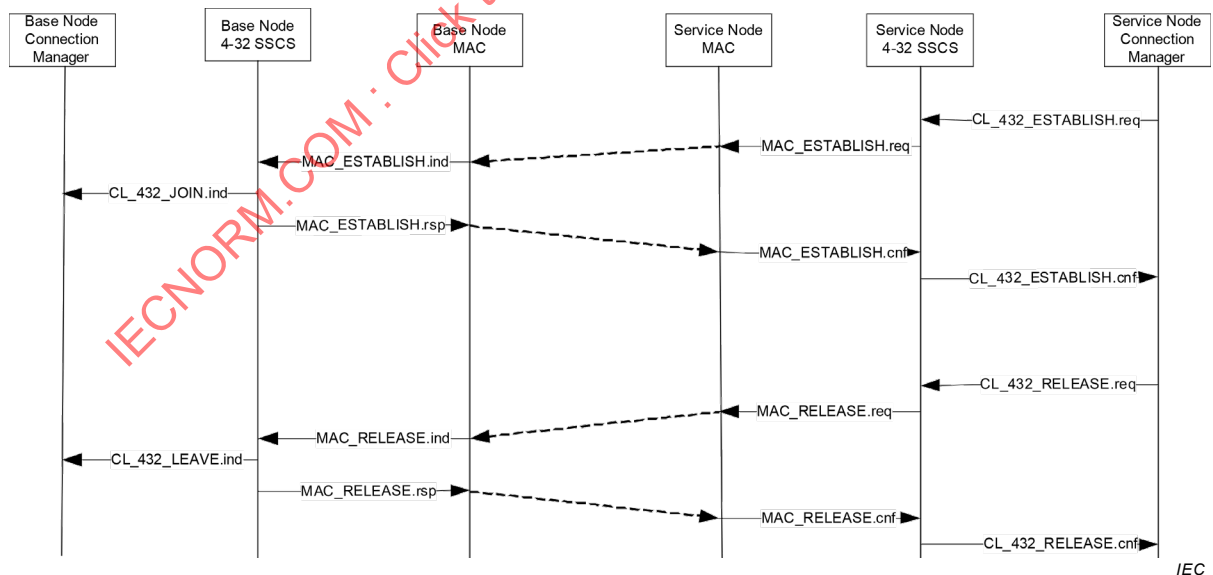


Figure 6 – MSC for IEC 61334-4-32 SSCS services

### 5.5.3 TCP-UDP/IPv4 profile

#### 5.5.3.1 Overview

##### 5.5.3.1.1 General architecture

The TCP-UDP/IPv4 based communication profile is fully in line with the DLMS/COSEM communication profile for TCP-UDP/IP, as specified in IEC 62056-9-7:2013. Refer to that standard for more information. The present subclause provides information related to the binding of TCP-UDP/IPv4 layers with the IPv4 SSCS only.

The IPv4 SSCS provides an efficient method for transporting IPv4 packets over the PRIME network.

Although IPv4 is a connectionless protocol, the IPv4 SSCS is connection-oriented meaning that a connection is established between the source and destination service nodes for the transfer of IP packets. This connection is maintained while traffic is being transferred and may be removed after a period of inactivity. The SSCS has two connection types:

- for address resolution: there is an address resolution connection to the base node;
- for IPv4 data transfer there is a connection per destination node. The destination may be the base node, another service node on the same subnetwork, or a destination outside of the subnetwork.

Optionally TCP/IPv4 headers may be compressed. The compression is negotiated as part of the connection establishment phase.

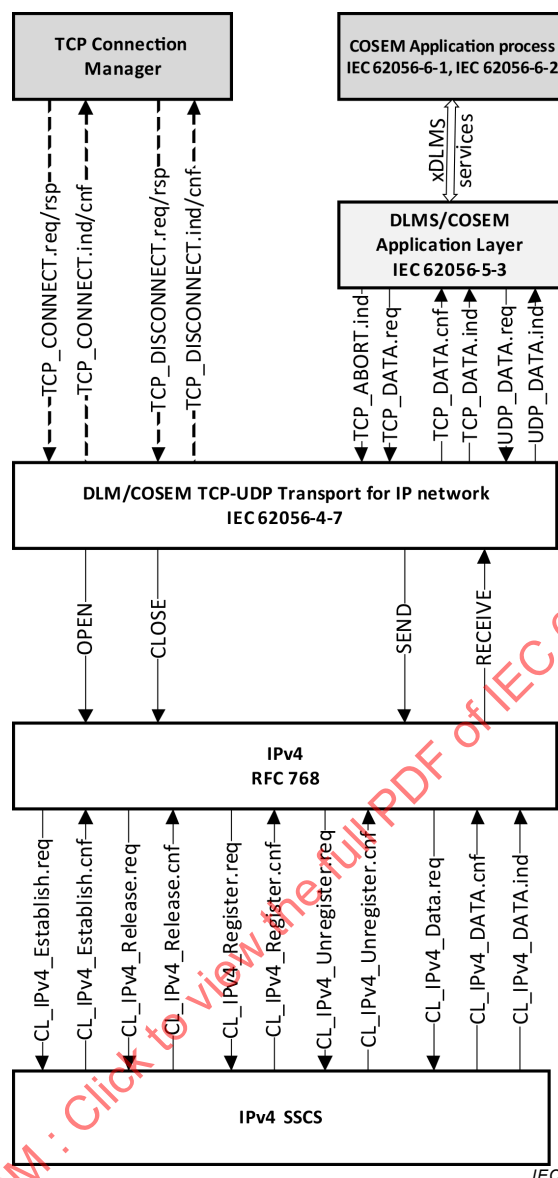
Broadcasting of IPv4 packets is supported using the MAC broadcast mechanism.

Multicasting of IPv4 packets is supported using the MAC multicast mechanism.

Segmentation and reassembly services are provided by the CPCS.

For more information see Recommendation ITU-T G.9904:2012.

The general architecture is shown in Figure 7.



NOTE Details related to the node which issues the primitives are provided in the service description. See 5.5.3.2 and 5.5.3.5.3.

**Figure 7 – IPv4 SCS services**

In this figure, only upper layers are shown. Lower layers and their relationship are described in Recommendation ITU-T G.9904:2012.

#### 5.5.3.1.2 The TCP connection manager

The TCP Connection Manager is specified in IEC 62056-9-7:2013.

#### 5.5.3.1.3 TCP-UDP/IPv4

The IPv4, TCP, and UDP layers implement the standard IPv4, TCP and UDP services respectively as specified in STD0005, STD0006 and STD0007. Specifying how the IPv4 layer is bound to the IPv4 SCS is the purpose of the following subclauses.

#### 5.5.3.1.4 Subnetwork gateway

The base node usually acts as a gateway to the PRIME subnetwork providing IP connectivity to the service nodes. Its main features are the following:

- it can exchange IPv4 packets with service nodes, and it can also forward packets from one service node to another service node;
- it may perform NAT functions;
- it may perform DHCP functions;

It is noted that IPv4 addresses can be statically preconfigured or dynamic. When the IPv4 address is dynamically configured, then a DHCP server is used. When using DHCP, it is performed using the broadcast/multicast services. Then the IPv4 address is registered using CL\_IPv4\_REGISTER service. It is not the scope of the present document to describe the DHCP or NAT process.

- it performs IPv4 to EUI-48 address resolution, maintaining a database of the EUI-48 address and IPv4 addresses of each service node registered to it. Service nodes – provided that they are registered with the base node – can then query the base node to resolve an IPv4 address to an EUI-48 address;
- it checks in each packet the IPv4 address to determine if the packet should be processed by a node in the subnetwork or should be sent to a destination outside the subnetwork.

In order to keep the implementation simple, only one single route is supported per local IPv4 address.

### 5.5.3.2 Opening and closing the IPv4 SSCS

#### 5.5.3.2.1 General

The following services are used to open and close the IPv4 SSCS. At a given moment, only one SSCS connection can exist, therefore the SSCS shall be opened once only. The IPv4 layer may close the SSCS when the IPv4 interface is brought down. The SSCS will also close when the underlying MAC connection to the base node is lost.

For MSC of IPv4 SSCS services please refer to Annex A.

#### 5.5.3.2.2 CL\_IPv4\_ESTABLISH.request

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.2.2.

##### *Function*

This service primitive is issued by the service node IPv4 layer to the IPv4 SSCS. This is done whenever the IPv4 layer brings the interface up.

##### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_ESTABLISH.request()

##### *Use*

On receiving this service primitive, the service node IPv4 SSCS will form the address resolution connection to the base node and join the broadcast group used for receiving/transmitting broadcast packets. See Figure A.1.

#### 5.5.3.2.3 CL\_IPv4\_ESTABLISH.confirm

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.2.3.

##### *Function*

This service primitive is passed from the service node IPv4 SSCS to the IPv4 layer. In case of success, it signals that the IPv4 SSCS is ready to accept IPv4 packets to be sent to peers.

##### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_ESTABLISH.confirm()

##### *Use*

This service primitive is used to provide the result of the CL\_IPv4\_ESTABLISH.request service. Once the IPv4 SSCS has established all the necessary connections and is ready to transmit and receive IPv4 packets, this primitive is passed to the IPv4 layer.

If the IPv4 SSCS encounters an error while opening, it responds with a CL\_IPv4\_RELEASE.confirm primitive instead of a CL\_IPv4\_ESTABLISH.confirm primitive.

#### 5.5.3.2.4 CL\_IPv4\_RELEASE.request

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.2.4.

##### *Function*

This service primitive is used by the service node IPv4 layer to the IPv4 SSCS to request closing the IPv4 SSCS.

##### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_RELEASE.request()

##### *Use*

This primitive is used by the service node when the SSCS has to be closed for any reason. The IPv4 SSCS closes all connections. All resources are released and no more IPv4 packets can be received or sent. When the IPv4 SSCS has released all its connections and resources, it returns a CL\_IPv4\_RELEASE.confirm service primitive.

#### 5.5.3.2.5 CL\_IPv4\_RELEASE.confirm

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.2.5.

##### *Function*

This service primitive is used by the service node IPv4 SSCS to indicate to the IPv4 layer that the IPv4 SSCS has been closed. This can be the result of a CL\_IPv4\_RELEASE.request primitive, a failed CL\_IPv4\_ESTABLISH.request primitive, or because the MAC layer indicates that the address resolution connection has been lost (i.e., this may happen because of an occurrence of a communication error, such as corrupted messages, too many ARQ retransmissions, etc.), or the service node itself is no longer registered, meaning no longer IP connectivity.

*Service parameters*

The semantics of this primitive are as follows:

```
CL_IPv4_RELEASE.confirm(Result)
```

The possible values of Result parameter are specified in Table 1.

*Use*

The IPv4 SSCS uses this service to inform the IPv4 layer that the service node has left the subnetwork at the IP level.

**5.5.3.3 Unicast address management****5.5.3.3.1 General**

The services defined here are used for unicast address management, i.e. the registration and unregistration of unicast IPv4 addresses with the IPv4 SSCS.

When there are no unicast IPv4 addresses registered with the IPv4 SSCS, the IPv4 SSCS can only send and receive broadcast and multicast packets. However, this is sufficient for BOOTP/DHCP operation to allow the device to obtain an IPv4 address. Once an IPv4 address has been registered, the IPv4 layer can transmit unicast packets that have a source address equal to one of its registered IPv4 addresses.

**5.5.3.3.2 CL\_IPv4\_REGISTER.request**

NOTE 1 For details, see Recommendation ITU-T G.9904:2012, 9.4.8.3.2.

*Function*

This primitive is passed from the service node IPv4 layer to the IPv4 SSCS to request registering an IPv4 address.

*Service parameters*

The semantics of this primitive are as follows:

```
CL_IPv4_REGISTER.req(
    IPv4_Address,
    Netmask,
    Gateway)
```

Where:

- the IPv4\_Address is the address to be registered;

NOTE 2 A service node can register several unicast IPv4 addresses.

- the Netmask is the network mask. It is used by the IPv4 SSCS to determine whether the packet should be delivered directly or the gateway should be used;
- the Gateway is an IPv4 address of the gateway to be used when the destination address is outside the subnetwork.

See also 5.5.3.1.4.

As each PRIME subnetwork has only one base node, the Gateway address is generally the IPv4 address of the base node, but this is not mandatory.

NOTE 3 The IPv4 Address, the Netmask and the Gateway address are provided by the IP layer and are configured inside the device by using an "IPv4 setup" object. See IEC 62056-6-2.

#### *Use*

From the service node side this registration is remote up to the base node.

If the service node has successfully registered its IPv4 address with the base node, a CL\_IPv4\_REGISTER.cnf primitive is used. If the registration fails, the CL\_IPv4\_RELEASE.cnf primitive will be used.

From the base node side, the registration is only local and used for notifying to the IPv4 SSCS which packets are to be passed to the upper layer.

#### **5.5.3.3.3 CL\_IPv4\_REGISTER.confirm**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.3.3.

#### *Function*

This service primitive is passed from the service node IPv4 SSCS to the IPv4 layer to indicate that the IPv4 address registration succeeded.

#### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_REGISTER.cnf(IPv4\_Address)

The IPv4\_Address is the address that has been successfully registered.

#### *Use*

The CL\_IPv4\_REGISTER.cnf service primitive is used by the service node IPv4 SSCS once the registration has been completed. The IPv4 layer can then make use of this address for sending IPv4 packets.

#### **5.5.3.3.4 CL\_IPv4\_UNREGISTER.request**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.3.4.

#### *Function*

This service primitive is passed from the service node IPv4 layer to the IPv4 SSCS to request unregistering an IPv4 address.

#### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_UNREGISTER.req(IPv4\_Address)

The IPv4\_Address is the address to be unregistered.

#### *Use*

Once the IPv4 address has been unregistered with the base node, a CL\_IPv4\_UNREGISTER.confirm primitive is used. In the case the CL\_IPv4\_UNREGISTER.request fails, a CL\_IPv4\_RELEASE.confirm primitive is sent back. The related reason of the failure is notified in the result.

Because several IPv4 addresses can be registered by the service node, therefore only the one concerned is unregistered by this request. The addresses not unregistered are still valid, until their unregistration or until the SSCS is closed.

#### **5.5.3.3.5 CL\_IPv4\_UNREGISTER.confirm**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.3.5.

##### *Function*

This service primitive is passed from the service node IPv4 SSCS to the IPv4 layer to indicate that unregistering of the IPv4 address has been successful.

##### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_UNREGISTER.confirm(IPv4\_Address)

The IPv4\_Address is the unicast IPv4 address that has been unregistered.

##### *Use*

Once unregistration has been completed, the IPv4 layer cannot send IPv4 packets using this source address.

#### **5.5.3.4 Multicast group management**

##### **5.5.3.4.1 General**

This subclause describes the service primitives used to manage multicast groups.

NOTE Notice the different nature of CL\_432\_JOIN primitives and the CL\_IPv4\_IGMP\_JOIN primitives. The first group informs the base node of the new service nodes that have joined the network. The second group will manage the subscription to a certain IP multicast group. Also note that this set of primitives have the IGMP in its name so the distinction is clear.

##### **5.5.3.4.2 CL\_IPv4\_IGMP\_Join.req**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.4.2.

##### *Function*

This service primitive is passed from the service node or base node IPv4 layer to the IPv4 SSCS. It contains an IPv4 multicast group address that is to be joined. The base node and the service node can use then multicast services.

##### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_IGMP\_JOIN.req(IPv4\_Address)

The IPv4\_Address is the IPv4 multicast group to be joined.

### *Use*

When the IPv4 SSCS receives this primitive, it will arrange for IPv4 packets sent to this group to be multicast in the subnetwork and to receive packets using this address to be passed to the IPv4 layer. If the SSCS can join the group, it shall use CL\_IPv4\_IGMP\_JOIN.confirm primitive to indicate success. Otherwise, it shall use the CL\_IPv4\_IGMP\_LEAVE.confirm service primitive.

#### **5.5.3.4.3 CL\_IPv4\_IGMP\_JOIN.confirm**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.4.3.

### *Function*

This service primitive is passed from the service node or the base node IPv4 SSCS to the IPv4 layer.

### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_IGMP\_JOIN.confirm(IPv4\_Address)

The IPv4\_Address is the IPv4 multicast group that has been joined.

### *Use*

The SSCS will start forwarding IPv4 multicast packets for the given multicast group.

#### **5.5.3.4.4 CL\_IPv4\_IGMP\_LEAVE.request**

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.4.4.

### *Function*

This service primitive is passed from the service node or base node IPv4 layer to the IPv4 SSCS. It contains an IPv4 multicast address to be left.

### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_IGMP\_LEAVE.req(IPv4\_Address)

The IPv4\_Address is the IPv4 multicast group to be left.

### *Use*

The SSCS will stop forwarding IPv4 multicast packets for this group and may leave the MAC multicast group.

#### 5.5.3.4.5 CL\_IPv4\_IGMP\_LEAVE.confirm

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.4.8.4.5.

##### *Function*

This service primitive is passed from the service node or the base node IPv4 SSCS to the IPv4 layer to indicate the result of the CL\_IPv4\_IGMP\_LEAVE.request.

##### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv4\_IGMP\_LEAVE.confirm(IPv4\_address, Result)

The IPv4\_Address is the address of IPv4 multicast group that has been left. The IPv4 SSCS will stop forwarding IPv4 multicast packets for the given multicast group.

For the Result values, see Table 1.

##### *Use*

This primitive can be used by the service node IPv4 SSCS:

- as a result of a CL\_IPv4\_IGMP\_LEAVE.request, or
- as a result of a CL\_IPv4\_IGMP\_JOIN.request in the case of failure, or
- due to an error condition resulting in the loss of the MAC multicast connection.

#### 5.5.3.5 IPv4 SSCS PDUs

##### 5.5.3.5.1 General

The following subclauses specify the IPv4 SSCS PDUs. The services which carry them are MAC sub layer services as shown on Figure 7 and Figure A.1.

##### 5.5.3.5.2 Address resolution PDUs

###### 5.5.3.5.2.1 General

The following messages are used over the address resolution connection between the service node and the base node. See Recommendation ITU-T G.9904:2012, 9.4.7.

AR.MSG values from 6 to 7 are reserved.

###### 5.5.3.5.2.2 AR\_REGISTER\_S

Table 2 shows the address resolution register message built by the service node SSCS and sent from the service node to the base node.

**Table 2 – AR\_REGISTER\_S message format**

| Name      | Length  | Description                                              |
|-----------|---------|----------------------------------------------------------|
| AR.MSG    | 8-bits  | Address Resolution Message Type<br>For AR_REGISTER_S = 0 |
| AR.IPv4   | 32-bits | IPv4 address to be registered                            |
| AR.EUI-48 | 48-bits | EUI-48 to be registered                                  |

**5.5.3.5.2.3 AR\_REGISTER\_B**

Table 3 shows the address resolution register acknowledgement message built by the base node SSCS and sent from the base node to the service node.

**Table 3 – AR\_REGISTER B message format**

| Name      | Length  | Description                                              |
|-----------|---------|----------------------------------------------------------|
| AR.MSG    | 8-bits  | Address Resolution Message Type<br>For AR_REGISTER_B = 1 |
| AR.IPv4   | 32-bits | IPv4 address registered                                  |
| AR.EUI-48 | 48-bits | EUI-48 registered                                        |

**5.5.3.5.2.4 AR\_UNREGISTER\_S**

Table 4 shows the address resolution unregister message sent from the service node to the base node.

**Table 4 – AR\_UNREGISTER\_S message format**

| Name      | Length  | Description                                                      |
|-----------|---------|------------------------------------------------------------------|
| AR.MSG    | 8-bits  | Address Resolution Message Type<br>For AR_UNREGISTER_S = 2       |
| AR.IPv4   | 32-bits | IPv4 address to be unregistered                                  |
| AR.EUI-48 | 48-bits | EUI-48 of the device which requested the IPv4 to be unregistered |

**5.5.3.5.2.5 AR\_MCAST\_REG\_S**

Table 5 shows the multicast address resolution register message sent from the service node to the base node.

**Table 5 – AR\_MCAST\_REG\_S message format**

| Name    | Length  | Description                                               |
|---------|---------|-----------------------------------------------------------|
| AR.MSG  | 8-bits  | Address Resolution Message Type<br>For AR_MCAST_REG_S = 8 |
| AR.IPv4 | 32-bits | IPv4 multicast address to be registered                   |

**5.5.3.5.2.6 AR\_MCAST\_REG\_B**

Table 6 shows the multicast address resolution register acknowledgment message sent from the base node to the service node.

The AR.IPv4 field is included in the AR\_MCAST\_REG\_B message so that the service node can perform multiple overlapping registrations.

**Table 6 – AR\_MCAST\_REG\_B message format**

| Name     | Length  | Description                                               |
|----------|---------|-----------------------------------------------------------|
| AR.MSG   | 8-bits  | Address Resolution Message Type<br>For AR_MCAST_REG_B = 9 |
| AR.IPv4  | 32-bits | IPv4 multicast address registered                         |
| Reserved | 2-bits  | Reserved. Shall be encoded as 0                           |
| AR.LCID  | 6-bits  | LCID assigned to this IPv4 multicast address              |

#### 5.5.3.5.2.7 AR\_MCAST\_UNREG\_S

Table 7 shows the multicast address resolution unregister message sent from the service node to the base node.

**Table 7 – AR\_MCAST\_UNREG\_S message format**

| Name    | Length  | Description                                                  |
|---------|---------|--------------------------------------------------------------|
| AR.MSG  | 8-bits  | Address Resolution Message Type<br>For AR_MCAST_UNREG_S = 10 |
| AR.IPv4 | 32-bits | IPv4 multicast address to be unregistered                    |

#### 5.5.3.5.2.8 AR\_MCAST\_UNREG\_B

Table 8 shows the multicast address resolution unregister acknowledgement message sent from the base node to the service node.

**Table 8 – AR\_MCAST\_UNREG\_B message format**

| Name    | Length  | Description                                                  |
|---------|---------|--------------------------------------------------------------|
| AR.MSG  | 8-bits  | Address Resolution Message Type<br>For AR_MCAST_UNREG_B = 11 |
| AR.IPv4 | 32-bits | IPv4 multicast address unregistered                          |

The AR.IPv4 field is included in the AR\_MCAST\_UNREG\_B message so that the service node can perform multiple overlapping unregistrations.

### 5.5.3.5.3 Data connection establishment

#### 5.5.3.5.3.1 Overview

The following PDUs are sent during the data connection establishment.

#### 5.5.3.5.3.2 AR\_LOOKUP\_S

Table 9 shows the address resolution lookup message sent from the service node to the base node.

**Table 9 – AR\_LOOKUP\_S message format**

| Name    | Length  | Description                                            |
|---------|---------|--------------------------------------------------------|
| AR.MSG  | 8-bits  | Address Resolution Message Type<br>For AR_LOOKUP_S = 4 |
| AR.IPv4 | 32-bits | IPv4 address to lookup                                 |

### 5.5.3.5.3.3 AR\_LOOKUP\_B

Table 10 shows the address resolution response message sent from the base node to the service node.

**Table 10 – AR\_LOOKUP\_B message format**

| Name      | Length  | Description                                                                                                                                     |
|-----------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| AR.MSG    | 8-bits  | Address Resolution Message Type<br>For AR_LOOKUP_B = 5                                                                                          |
| AR.IPv4   | 32-bits | IPv4 address looked up                                                                                                                          |
| AR.EUI-48 | 48-bits | EUI-48 for IPv4 address                                                                                                                         |
| AR.Status | 8-bits  | Lookup status, indicating if the address was found or an error occurred.<br><br>0 = found, AR.EUI-48 valid.<br>1 = unknown, AR.EUI-48 undefined |

The lookup may fail if the requested address has not been registered. In that case, AR.Status will have a value other than zero and the contents of AR.EUI-48 will be undefined. The lookup is only successful when AR.Status is zero. In that case, the EUI-48 field contains the resolved address.

### 5.5.3.6 IPv4 SSCS packet format

#### 5.5.3.6.1 General

The following PDU formats are used for transferring IPv4 packets between service nodes. Two formats are defined. The first format is used without header compression. The second format is used with Van Jacobsen header compression.

#### 5.5.3.6.2 IPv4 packet format without header compression

When no header compression has been negotiated, the IP packet is simply sent as it is, without any header compression; see Table 11. For compression negotiation, see 5.5.3.7.2.

**Table 11 – IPv4 packet format without header compression negotiated**

| Name    | Length   | Description     |
|---------|----------|-----------------|
| IPv4PKT | n-octets | The IPv4 Packet |

#### 5.5.3.6.3 IPv4 packet format with Van Jacobsen header compression

Table 12 shows the IPv4 packet format when Van Jacobsen header compression has been negotiated. See RFC 1144:1990, 3.2.

**Table 12 – IPv4 packet format with VJ header compression**

| Name      | Length   | Description                                                                                                                                               |
|-----------|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| IPv4.Type | 2-bits   | Type of compressed packet.<br>IPv4.Type = 0 – TYPE_IP<br>IPv4.Type = 1 – UNCOMPRESSED_TCP<br>IPv4.Type = 2 – COMPRESSED_TCP<br>IPv4.Type = 3 – TYPE_ERROR |
| IPv4.Seq  | 6-bits   | Packet sequence number                                                                                                                                    |
| IPv4.PKT  | n-octets | The IPv4 Packet (with the header compressed)                                                                                                              |

The IPv4.Type value TYPE\_ERROR is never sent. It is a pseudo packet type used to tell the decompressor that a packet has been lost.

### 5.5.3.7 Connection data

#### 5.5.3.7.1 General

The following PDUs are exchanged during the data connection establishment phase. When a connection is established between service nodes for the transfer of IPv4 packets, data is also transferred in the connection request packets. This data allows the negotiation of compression and notification of the IPv4 address. See Recommendation ITU-T G.9904:2012, 8.6.6.1.

#### 5.5.3.7.2 Connection data from the initiator

Table 13 shows the connection data as it has to be sent by the connection initiator.

**Table 13 – Connection data sent by the initiator**

| Name      | Length  | Description                                                                                                                                                              |
|-----------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Reserved  | 6-bits  | Should be encoded as zero in this version of the SSCS protocol                                                                                                           |
| Data.HC   | 2-bits  | Header Compression<br>Data.HC = 0 – No compression requested<br>Data.HC = 1 – VJ Compression requested<br>Data.HC = 2, 3 – Reserved for future versions of this document |
| Data.IPv4 | 32-bits | IPv4 address of the initiator                                                                                                                                            |

If the device accepts the connection, it should copy the Data.IPv4 address into a new table entry along with the negotiated Data.HC value.

#### 5.5.3.7.3 Connection data from the responder

Table 14 shows the connection data sent in response to the connection request.

**Table 14 – Connection data sent by the responder**

| Name     | Length | Description                                                                                                                                    |
|----------|--------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Reserved | 6-bits | Should be encoded as zero in this version of the SSCS protocol                                                                                 |
| Data.HC  | 2-bit  | Header Compression negotiated<br>Data.HC = 0 – No compression permitted<br>Data.HC = 1 – VJ Compression negotiated<br>Data.HC = 2,3 – Reserved |

A header compression scheme can only be used when it is supported by both service nodes. The responder may only set Data.HC to 0 or the same value as the one received from the initiator. When the same value is used, it indicates that the requested compression scheme has been negotiated and will be used for the connection. Setting Data.HC to 0 allows the responder to deny the request for that header compression scheme or force the use of no header compression.

## **5.5.4 TCP-UDP/IPv6 profile**

### **5.5.4.1 Overview**

#### **5.5.4.1.1 General architecture**

The general TCP-UDP/IPv6 architecture is the same as that of the TCP-UDP/IPv4, meaning according to IEC 62056-9-7:2013. The lower layers are according to Recommendation ITU-T G.9904:2012. The present subclause provides information related to the binding of TCP-UDP/IPv6 layers with the IPv6 SSCS. See Figure 2.

The IPv6 SSCS provides an efficient method for transferring IPv6 packets over the PRIME network.

A service node can pass IPv6 packets to the base node or directly to other service nodes.

By default, the base node acts as a router between the PRIME subnet and the backbone network. A base node shall have at least this connectivity capability. Any other node inside the subnetwork can also act as a gateway. The base node can also act as a NAT router. However given the abundance of IPv6 addresses this is not expected. How the base node connects to the backbone is beyond the scope of this document.

#### **5.5.4.1.2 IPv6 unicast addressing assignment**

IPv6 service nodes (and base nodes) shall support the standard IPv6 protocol, as described in RFC 2460.

IPv6 service nodes (and base nodes) shall support the standard IPv6 addressing architecture, as described in RFC 4291.

IPv6 service nodes (and base nodes) shall support global unicast IPv6 addresses, link-local IPv6 addresses and multicast IPv6 addresses, as described in RFC 4291.

IPv6 service nodes (and base nodes) shall support automatic address configuration using stateless address configuration as specified in RFC 4862. They may also support automatic address configuration using stateful address configuration as specified in RFC 3315 and they may support manual configuration of IPv6 addresses. The decision of which address configuration scheme to use is deployment specific.

Service nodes shall support DHCPv6 client, when base nodes support DHCPv6 server as specified in RFC 3315 for stateless address configuration.

#### **5.5.4.1.3 Role of the base node**

At the IPv6 SSCS level, the base node maintains a table containing all the IPv6 unicast addresses and the EUI-48 addresses related to them. One of the roles of the base node is to perform IPv6 to EUI-48 address resolution. Each service node belonging to the subnetwork managed by the base node registers its IPv6 address and EUI-48 address with the base node. Other service nodes can then query the base node to resolve an IPv6 address into a EUI-48 address. This requires the establishment of a dedicated connection to the base node for address resolution.

Optionally, UDP/IPv6 headers may be compressed. Compression is negotiated as part of the connection establishment phase. Currently one header compression technique is described in this document that is used for transmission of IPv6 packets over IEEE 802.15.4-based networks, as defined in RFC 6282. This is also known as LOWPAN\_IPHC.

The multicasting of IPv6 packets is supported using the MAC multicast mechanism.

#### **5.5.4.2 IPv6 SSCS**

##### **5.5.4.2.1 General**

The IPv6 SSCS has two types of connection types. For address resolution, there is a connection to the base node. For IPv6 data transfer, there is one connection per destination node. The base node may act as the IPv6 gateway to the outside world or another node in the same subnetwork.

##### **5.5.4.2.2 Routing in the subnetwork**

Routing IPv6 packets is the scope of the IPv6 SSCS. In other words, the IPv6 SSCS will decide whether the packet should be sent directly to another service node or forwarded to the configured gateway depending on the IPv6 destination address.

Although IPv6 is a connectionless protocol, the IPv6 SSCS is connection-oriented. Once address resolution has been performed, a connection is established between the source and destination service nodes for the transfer of IPv6 packets. See 5.5.4.11.4. This connection is maintained all the time the traffic is being transferred and may be removed after a period of inactivity.

##### **5.5.4.2.3 CPCS: Segmentation and reassembly**

The CPCS sublayer shall always be present with the IPv6 SSCS allowing segmentation and reassembly facilities. Thus, the MSDUs generated by the IPv6 SSCS are always less than CMTUSize bytes and application messages are expected to be no longer than CIMaxAppPktSize.

#### **5.5.4.3 IPv6 Address Configuration**

##### **5.5.4.3.1 General**

The service nodes may use manually configured IPv6 addresses, link local addresses, stateless auto-configuration according to RFC 4862, or DHCPv6 to obtain IPv6 addresses. All nodes shall support the unicast link local address in addition to other configured addresses below, and multicast addresses if ever the node belongs to multicast groups.

#### **5.5.4.3.2 Interface identifier**

In order to make use of stateless address auto configuration and link local addresses it is necessary to define how the interface identifier, as defined in RFC 4291, is derived. Each PRIME node has a unique EUI-48. This EUI-48 is converted into a EUI-64 in the same way as for Ethernet networks as defined in RFC 2464. This EUI-64 is then used as the interface identifier.

#### **5.5.4.3.3 IPv6 Link local address configuration**

The IPv6 link local address of a PRIME interface is formed by appending the interface identifier as defined above to the Prefix FE80::/64.

#### **5.5.4.3.4 Stateless address auto configuration**

An IPv6 address prefix of a PRIME interface used for stateless auto configuration, as defined in RFC 4862, shall have a length of 64 bits. The IPv6 prefix is obtained by the service nodes from the base node via Router Advertisement messages, which are sent periodically by routers or on request by the base node.

#### **5.5.4.3.5 Stateful address auto configuration**

Alternatively, IPv6 addresses can be configured using DHCPv6 as described in RFC 3315. DHCPv6 can provide a device with addresses assigned by a DHCPv6 server and other configuration information which are carried in options.

#### **5.5.4.3.6 Multicast address**

IPv6 service nodes (and base nodes) shall support multicast IPv6 addressing, as described in RFC 4291:2006, 2.7.

#### **5.5.4.3.7 Address resolution**

##### **5.5.4.3.7.1 General**

The IPv6 layer will present the IPv6 SSCS with an IPv6 packet to be transferred. The IPv6 SSCS is responsible for determining which service node the packet should be delivered to, using the IPv6 addresses in the packet. The IPv6 SSCS shall then establish a connection to the destination if one does not already exist so that the packet can be transferred. Two classes of IPv6 addresses can be used and the following subclause describes how these addresses are resolved into PRIME EUI-48 addresses. It should be noted that IPv6 does not have a broadcast address. However, broadcasting is possible using multicast to all nodes addresses.

##### **5.5.4.3.7.2 Unicast address**

###### **5.5.4.3.7.2.1 General**

IPv6 unicast addresses shall be resolved into PRIME unicast EUI-48 addresses. The base node maintains a central database of IPv6 addresses and EUI-48 addresses. Address resolution functions are performed by querying this database. The service node shall establish a connection to the address resolution service running on the base node, using the TYPE value TYPE\_CL\_IPv6\_AR (see Recommendation ITU-T G.9904:2012, Table C.1). No data should be passed in the connection establishment signalling.

###### **5.5.4.3.7.2.2 Address registration and unregistration**

A service node uses the AR\_REGISTERv6\_S message to register an IPv6 address and the corresponding EUI-48 address. The base node will acknowledge an AR\_REGISTERv6\_B message. The service node may register multiple IPv6 addresses for the same EUI-48.

A service node uses the AR\_UNREGISTERv6\_S message to unregister an IPv6 address and the corresponding EUI-48 address. The base node will acknowledge with an AR\_UNREGISTERv6\_B message.

When the address resolution connection between the service node and the base node is closed, the base node should remove all addresses associated with that connection.

#### **5.5.4.3.7.2.3 Address lookup**

A service node uses the AR\_LOOKUPv6\_S message to perform a lookup. The message contains the IPv6 address to be resolved. The base node shall respond with an AR\_LOOKUPv6\_B message that contains an error code and, if there is no error, the EUI-48 associated with the IPv6 address. If the base node has multiple entries in its database for the same IPv6 address, the possible EUI-48 returned is undefined.

It should be noted that in the case of link local addresses – due to the fact that the EUI-48 can be obtained from the IPv6 address – the lookup can simply return this value by extracting it from the IPv6 address.

#### **5.5.4.3.7.2.4 Multicast address**

Multicast IPv6 addresses are mapped to connection handles (ConnHandle) by the IPv6 SSCS.

To join a multicast group, the IPv6 SSCS uses the MAC\_JOIN.request primitive with the IPv6 address specified in the data field. A corresponding MAC\_JOIN.confirm primitive will be generated by the MAC layer after completion of the join process. The MAC\_Join.confirm primitive will contain the result (success/failure) and the corresponding ConnHandle to be used by the IPv6 SSCS. The MAC layer will handle the transfer of data for this connection using the appropriate LCIDs. To leave the multicast group, the IPv6 SSCS at the service node shall use the MAC\_LEAVE.request(ConnHandle) primitive.

To send an IPv6 multicast packet, the IPv6 SSCS will simply send the packet to the group, using the allocated ConnHandle. The ConnHandle is maintained while there are more packets to be sent. However, after Tmcast\_reg seconds of not sending an IPv6 multicast packet to the group, the node should release the ConnHandle by using the MAC\_LEAVE.request primitive. The nominal value of Tmcast\_reg is 10 min; however, other values may be used.

NOTE Tmcast\_reg is the timeout for unregistering from the multicast group. See Recommendation ITU-T G.9904:2012, 9.4.2.4.

#### **5.5.4.3.7.2.5 Retransmission of address resolution packets**

The connection between the service node and the base node for address resolution is not reliable. The MAC ARQ service is not used. The service node is responsible for making retransmissions if the base node does not respond in one second. It is not considered an error when the base node receives the same registration requests multiple times or is asked to remove a registration that does not exist. These conditions can be the result of retransmissions.

#### **5.5.4.4 IPv6 packet transfer**

For packets to be transferred, a connection needs to be established between the source and destination nodes (see Figure A.1). The IPv6 SSCS will examine each IP packet to determine the destination EUI-48 address. If a connection to the destination has already been established, the packet is simply sent. For this purpose, the IPv6 SSCS keeps a table for each connection it has with information shown in Table 15. To use this table, it is first necessary to determine if the remote address is in the local subnet or if ever a gateway has to be used. The netmask associated with the local IPv6 address is used to determine this. If the destination address is not in the local subnetwork, the address of the gateway is used instead of the destination address when the table is searched.

**Table 15 – IPv6 SSCS table entry**

| Parameter             | Description                                   |
|-----------------------|-----------------------------------------------|
| CL_IPv6_Con.Remote_IP | Remote IP address of this connection          |
| CL_IPv6_Con.ConHandle | MAC Connection handle for the connection      |
| CL_IPv6_Con.LastUsed  | Timestamp of last packet received/transmitted |
| CL_IPv6_Con.HC        | Header Compression scheme being used          |

The IPv6 SSCS may close a connection when it has not been used for an implementation-defined time period. When the connection is closed the entry for the connection is removed at both ends of the connection.

When a connection to the destination does not exist, more work is necessary. The address resolution service is used to determine the EUI-48 address of the remote IP address if it is local or the gateway associated with the local address if the destination address is in another subnet. When the base node replies with the EUI-48 address of the destination service node, a MAC connection is established to the remote device. The TYPE value of this established connection is TYPE\_CL\_IPv6\_UNICAST (see Recommendation ITU-T G.9904:2012, Table C.1). The data passed in the request message is defined in 5.5.4.11.3. The local IPv6 address is provided so that the remote device can add the new connection to its cache of connections for sending data in the opposite direction. The use of header compression is also negotiated as part of the connection establishment. Once the MAC connection has been established, the IPv6 packet can be sent.

#### **5.5.4.5 Segmentation and reassembly**

The IPv6 SSCS shall support IPv6 packets with an MTU of 1 500 bytes. This requires the use of the common part convergence sublayer segmentation and reassembly service.

#### **5.5.4.6 Compression**

It is assumed that any PRIME device is capable of LOWPAN\_IPHC IPv6 header compression/decompression. It may also be capable of performing UDP compression/decompression. Thus UDP/IPv6 compression is negotiated.

No negotiation can take place for multicast packet. Nodes can only make use of mandatory compression capabilities.

Depending on the type of IPv6 address carried by the packet and the capabilities which are negotiated between the nodes involved in the data exchanges, IPv6 header compression is performed.

All the service nodes and the base node shall support IPv6 Header Compression using source and destination Addresses stateless compression as defined in RFC 6282. Source and destination IPv6 addresses using stateful compression and IPv6 Next header compression are negotiable.

#### **5.5.4.7 Quality of Service Mapping**

In the specification of the PRIME MAC layer it is specified that the contention-based access mechanism supports 4 priority levels (1-4). Level 1 is used for MAC signalling messages, but not exclusively so.

IPv6 packets include a traffic class field in the header to indicate the QoS the packet would like to receive. This traffic class can be used in the same way that IPv4 ToS, see Recommendation ITU-T G.9904:2012, 9.4.6. That is, three bits of the ToS indicate the IP

Precedence. Table 16 specifies how the IP Precedence is mapped into the PRIME MAC priority.

**Table 16 – Mapping IPv6 precedence to PRIME MAC priority**

| IP Precedence                                                                                                                                                          | MAC Priority |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| 000 – Routine                                                                                                                                                          | 4            |
| 001 – Priority                                                                                                                                                         | 4            |
| 010 – Immediate                                                                                                                                                        | 3            |
| 011 – Flash                                                                                                                                                            | 3            |
| 100 – Flash Override                                                                                                                                                   | 2            |
| 101 – Critical                                                                                                                                                         | 2            |
| 110 – Internetwork Control                                                                                                                                             | 1            |
| 111 – Network Control                                                                                                                                                  | 1            |
| NOTE At the MAC layer level the priority as set in the Packet header field is the value assigned in this table minus 1, as the range of PKT.PRIO field is from 0 to 3. |              |

#### 5.5.4.8 Opening and closing the IPv6 SSCS

##### 5.5.4.8.1 General

The following services are used to open and close the IPv6 SSCS. At a given moment, only one IPv6 SSCS connection can exist. Therefore, the IPv6 SSCS shall be opened once only. The IPv6 layer may close the IPv6 SSCS when the IPv6 interface is brought down. The IPv6 SSCS will also close when the underlying MAC connection to the base node is lost.

NOTE For details, see Recommendation ITU-T G.9904:2012, 9.6.9.2.

##### 5.5.4.8.2 CL\_IPv6\_ESTABLISH.request

###### Function

This service primitive is issued by the service node IPv6 layer to the IPv6 SSCS. This is performed whenever the IPv6 layer brings the interface up.

###### Service parameters

The semantics of this primitive are as follows:

CL\_IPv6\_ESTABLISH.request()

###### Use

On receiving this service primitive, the service node IPv6 SSCS will form the address resolution connection to the base node.

##### 5.5.4.8.3 CL\_IPv6\_Establish.confirm

###### Function

This service primitive is passed from the service node IPv6 SSCS to the IPv6 layer. In case of success, it signals that the IPv6 SSCS is ready to accept IPv6 packets to be sent to peers.

### *Service parameters*

The semantics of this primitive are as follows:

`CL_IPv6_Establish.confirm()`

### *Use*

This service primitive is used to provide the result of the `CL_IPv6_Establish.request` service. Once the IPv6 SSCS has established all the connections and is ready to transmit and receive IPv6 packets, this primitive is passed to the IPv6 layer.

If the IPv6 SSCS encounters an error while opening, it responds with a `CL_IPv6_Release.confirm` primitive, instead of a `CL_IPv6_Establish.confirm` primitive

#### **5.5.4.8.4 CL\_IPv6\_Release.request**

### *Function*

This service primitive is used by the service node IPv6 layer to the IPv6 SSCS to request closing the IPv6 SSCS connection.

### *Service parameters*

The semantics of this primitive are as follows:

`CL_IPv6_Release.request()`

### *Use*

This primitive is used by the service node when the connection has to be closed for any reason. The IPv6 SSCS closes all connections. All resources are released and no more IPv6 packets can be received or sent. When the IPv6 SSCS has released all its connections and resources, it returns a `CL_IPv6_Release.confirm` service primitive.

#### **5.5.4.8.5 CL\_IPv6\_Release.confirm**

### *Function*

This service primitive is used by the service node IPv6 SSCS to indicate to the IPv6 layer that the IPv6 SSCS has been closed. This can be the result of a `CL_IPv6_Release.request` primitive, a failed `CL_IPv6_Establish.request` primitive, or because the MAC layer indicates that the address resolution connection has been lost, or the service node itself is no longer registered.

### *Service parameters*

The semantics of this primitive are as follows:

`CL_IPv6_Release.confirm(Result)`

The possible values of Result parameter are specified in Table 1.

*Use*

The IPv6 SSCS uses this service to inform the IPv6 layer that the service node has left the subnetwork.

**5.5.4.9 Unicast address management****5.5.4.9.1 General**

The services defined here are used for address management, i.e. the registration and unregistration of unicast IPv6 addresses with the IPv6 SSCS.

**5.5.4.9.2 CL\_IPv6\_Register.request***Function*

This primitive is passed from the service node IPv6 layer to the IPv6 SSCS to request registering an IPv6 address.

*Service parameters*

The semantics of this primitive are as follows:

```
CL_IPv6_REGISTER.req(
    IPv6_Address,
    Netmask,
    Gateway)
```

Where:

- the IPv6\_Address is the address to be registered;
- NOTE A service node can register several unicast IPv6 addresses.
- the Netmask is the network mask. It is used by the IPv6 SSCS to determine whether the packet should be delivered directly or the gateway should be used;
  - the Gateway is an IPv6 address of the gateway to be used when the destination address is outside the subnetwork.

As each PRIME OFDM subnetwork may have only one base node, the base node may act as a Gateway between the PRIME subnetwork and the backbone. All the base nodes shall have this connectivity capability. Any other node inside the subnetwork can also act as a gateway.

*Use*

From the service node side this registration is remote up to the base node.

If the service node has successfully registered its IPv6 address with the base node, a CL\_IPv6\_REGISTER.cnf primitive is used. If the registration fails, the CL\_IPv6\_Release.cnf primitive will be used.

**5.5.4.9.3 CL\_IPv6\_Register.confirm***Function*

This service primitive is passed from the service node IPv6 SSCS to the IPv6 layer to indicate that the IPv6 address registration succeeded.

### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv6\_REGISTER.cnf(IPv6\_Address)

The IPv6\_Address is the address that has been successfully registered.

### *Use*

The CL\_IPv6\_Register.cnf service primitive is used by the service node IPv6 SCS once the registration has been completed. The IPv6 layer can then make the use of this address for sending send IPv6 packets.

#### **5.5.4.9.4 CL\_IPv6\_UNREGISTER.request**

### *Function*

This service primitive is passed from the service node IPv6 layer to the IPv6 SCS to request unregistering an IPv6 address.

### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv6\_UNREGISTER.req(IPv6\_Address)

The IPv6\_Address is the address to be unregistered.

### *Use*

Once the IPv6 address has been unregistered with the base node, a CL\_IPv6\_UNREGISTER.confirm primitive is used. In the case the CL\_IPv6\_Unregister.request fails, a CL\_IPv6\_Release.confirm primitive is sent back. The related reason of the failure is notified in the result.

As several IPv6 addresses can be registered by the service node, therefore only the one concerned is unregistered by this request. The addresses not unregistered are still valid, until their unregistration or a release primitive takes place.

#### **5.5.4.9.5 CL\_IPv6\_Unregister.confirm**

### *Function*

This service primitive is passed from the service node IPv6 SCS to the IPv6 layer to indicate that unregistering of the IPv6 address has been successful.

### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv6\_UNREGISTER.confirm(IPv6\_Address)

The IPv6\_Address is the unicast IPv6 address that has been unregistered.

### *Use*

Once unregistration has been completed, the IPv6 layer will not send IPv6 packets using this source address.

## **5.5.4.10 Multicast group management**

### **5.5.4.10.1 General**

This subclause describes the primitives used to manage multicast groups.

NOTE Note the different nature of CL\_432\_JOIN primitives and the CL\_IPv6\_MUL\_JOIN primitives. The first group informs the base node of the new service nodes that have joined the network. The second group will manage the subscription to a certain IP multicast group. Also note that this set of primitives have the MUL in their name so the distinction is clear.

### **5.5.4.10.2 CL\_IPv6\_MUL\_Join.request**

#### *Function*

This primitive is passed from the IPv6 layer to the IPv6 convergence layer. It contains an IPv6 multicast address that is to be joined.

#### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv6\_MUL\_JOIN.request(IPv6\_Address)

The IPv6\_Address is the IPv6 multicast group address that is to be joined.

#### *Use*

When the convergence layer receives this primitive, it will arrange for IP packets sent to this group to be multicast in the PRIME network and receive packets using this address to be passed to the IPv6 stack. If the IPv6 SSCS cannot join the group, it uses the CL\_IPv6\_MUL\_LEAVE.confirm primitive. Otherwise, the CL\_IPv6\_MUL\_JOIN.confirm primitive is used to indicate success.

### **5.5.4.10.3 CL\_IPv6\_MUL\_Join.confirm**

#### *Function*

This primitive is passed from the IPv6 convergence layer to the IPv6 layer. It contains a result status and an IPv6 multicast address that has been joined.

#### *Service parameters*

The semantics of this primitive are as follows:

CL\_IPv6\_MUL\_JOIN.confirm(IPv6\_Address)

The IPv6\_Address is the IPv6 multicast group address that was joined.

#### *Use*

The convergence layer will start forwarding IPv6 multicast packets for the given multicast group.

#### 5.5.4.10.4 CL\_IPv6\_MUL\_Leave.request

##### *Function*

This primitive is passed from the IPv6 layer to the IPv6 convergence layer. It contains an IPv6 multicast address to be left.

##### *Service parameters*

The semantics of this primitive are as follows:

```
CL_IPv6_MUL_LEAVE.request(IPv6_address)
```

The IPv6\_address is the IPv6 multicast group address to be left.

##### *Use*

The convergence layer will stop forwarding IPv6 multicast packets for this group and may leave the PRIME MAC multicast group.

#### 5.5.4.10.5 CL\_IPv6\_MUL\_Leave.confirm

##### *Function*

This primitive is passed from the IPv6 convergence layer to the IPv6. It contains a result status and an IPv6 multicast address that was left.

##### *Service parameters*

The semantics of this primitive are as follows:

```
CL_IPv6_MUL_LEAVE.confirm(IPv6_address, Result)
```

The IPv6\_address is the IPv6 multicast group that was left.

##### *Use*

The convergence layer will stop forwarding IPv6 multicast packets for the given multicast group.

The Result takes a value from Table 1.

This primitive can be used by the convergence layer as a result of a CL\_IPv6\_MUL\_JOIN.request, CL\_IPv6\_MUL\_LEAVE.request or because of an error condition resulting in the loss of the PRIME MAC multicast connection.

#### 5.5.4.11 IPv6 Packet formats and PDUs

##### 5.5.4.11.1 Overview

This subclause defines the format of convergence layer PDUs.

### 5.5.4.11.2 Address resolution PDUs

#### 5.5.4.11.2.1 General

The following PDUs are transferred over the address resolution connection between the service node and the base node. The following subclause defines a number of AR.MSG values.

#### 5.5.4.11.2.2 AR\_REGISTERv6\_S

Table 17 shows the address resolution register message sent from the service node to the base node.

**Table 17 – AR\_REGISTERv6\_S message format**

| Name      | Length   | Description                                                 |
|-----------|----------|-------------------------------------------------------------|
| AR.MSG    | 8-bits   | Address Resolution Message Type<br>For AR_REGISTERv6_S = 16 |
| AR.IPv6   | 128-bits | IPv6 address to be registered                               |
| AR.EUI-48 | 48-bits  | EUI-48 to be registered                                     |

#### 5.5.4.11.2.3 AR\_REGISTERv6\_B

Table 18 shows the address resolution register acknowledgment message sent from the base node to the service node.

**Table 18 – AR\_REGISTERv6\_B message format**

| Name      | Length   | Description                                                 |
|-----------|----------|-------------------------------------------------------------|
| AR.MSG    | 8-bits   | Address Resolution Message Type<br>For AR_REGISTERv6_B = 17 |
| AR.IPv6   | 128-bits | IPv6 address registered                                     |
| AR.EUI-48 | 48-bits  | EUI-48 registered                                           |

The AR.IPv6 and AR.EUI-48 fields are included in the AR\_REGISTERv6\_B message so that the service node can perform multiple overlapping registrations.

#### 5.5.4.11.2.4 AR\_UNREGISTERv6\_S

Table 19 shows the address resolution unregister message sent from the service node to the base node.

**Table 19 – AR\_UNREGISTERv6\_S message format**

| Name      | Length   | Description                                                   |
|-----------|----------|---------------------------------------------------------------|
| AR.MSG    | 8-bits   | Address Resolution Message Type<br>For AR_UNREGISTERv6_S = 18 |
| AR.IPv6   | 128-bits | IPv6 address to be unregistered                               |
| AR.EUI-48 | 48-bits  | EUI-48 to be unregistered                                     |

#### 5.5.4.11.2.5 AR\_UNREGISTERv6\_B

Table 20 shows the address resolution unregister acknowledgment message sent from the base node to the service node.

**Table 20 – AR\_UNREGISTERv6\_B message format**

| Name      | Length   | Description                                                   |
|-----------|----------|---------------------------------------------------------------|
| AR.MSG    | 8-bits   | Address Resolution Message Type<br>For AR_UNREGISTERv6_B = 19 |
| AR.IPv6   | 128-bits | IPv6 address unregistered                                     |
| AR.EUI-48 | 48-bits  | EUI-48 unregistered                                           |

The AR.IPv6 and AR.EUI-48 fields are included in the AR\_UNREGISTERv6\_B message so that the service node can perform multiple overlapping unregistrations.

#### 5.5.4.11.2.6 AR\_LOOKUPv6\_S

Table 21 shows the address resolution lookup message sent from the service node to the base node.

**Table 21 – AR\_LOOKUPv6\_S message format**

| Name    | Length   | Description                                               |
|---------|----------|-----------------------------------------------------------|
| AR.MSG  | 8-bits   | Address Resolution Message Type<br>For AR_LOOKUPv6_S = 20 |
| AR.IPv6 | 128-bits | IPv6 address to lookup                                    |

#### 5.5.4.11.2.7 AR\_LOOKUPv6\_B

Table 22 shows the address resolution lookup response message sent from the base node to the service node.

**Table 22 – AR\_LOOKUPv6\_B message format**

| Name      | Length   | Description                                                                                                                                 |
|-----------|----------|---------------------------------------------------------------------------------------------------------------------------------------------|
| AR.MSG    | 8-bits   | Address Resolution Message Type<br>For AR_LOOKUPv6_B = 21                                                                                   |
| AR.IPv6   | 128-bits | IPv6 address looked up                                                                                                                      |
| AR.EUI-48 | 48-bits  | EUI-48 for IPv6 address                                                                                                                     |
| AR.Status | 8-bits   | Lookup status, indicating if the address was found or an error occurred.<br>0 = found, AR.EUI-48 valid.<br>1 = unknown, AR.EUI-48 undefined |

The lookup may fail if the requested address has not been registered. In that case, AR.Status will have a value equal to 1, and the contents of AR.EUI-48 will be undefined. The lookup is only successful when AR.Status is zero. In that case, the EUI-48 field contains the resolved address.

#### 5.5.4.11.2.8 AR\_MCAST\_REGv6\_S

Table 23 shows the multicast address resolution register message sent from the service node to the base node.

**Table 23 – AR\_MCAST\_REGv6\_S message format**

| Name    | Length   | Description                                                  |
|---------|----------|--------------------------------------------------------------|
| AR.MSG  | 8-bits   | Address Resolution Message Type<br>For AR_MCAST_REGv6_S = 24 |
| AR.IPv6 | 128-bits | IPv6 multicast address to be registered                      |

#### 5.5.4.11.2.9 AR\_MCAST\_REGv6\_B

Table 24 shows the multicast address resolution register acknowledgment message sent from the base node to the service node.

**Table 24 – AR\_MCAST\_REGv6\_B message format**

| Name     | Length   | Description                                                  |
|----------|----------|--------------------------------------------------------------|
| AR.MSG   | 8-bits   | Address Resolution Message Type<br>For AR_MCAST_REGv6_B = 25 |
| AR.IPv6  | 128-bits | IPv6 multicast address registered                            |
| Reserved | 2-bits   | Reserved. Should be encoded as 0.                            |
| AR.LCID  | 6-bits   | LCID assigned to this IPv6 multicast address                 |

The AR.IPv6 field is included in the AR\_MCAST\_REGv6\_B message so that the service node can perform multiple overlapping registrations.

#### 5.5.4.11.2.10 AR\_MCAST\_UNREGv6\_S

Table 25 shows the multicast address resolution unregister acknowledgment message sent from the base node to the service node.

**Table 25 – AR\_MCAST\_UNREGv6\_B message format**

| Name    | Length   | Description                                                    |
|---------|----------|----------------------------------------------------------------|
| AR.MSG  | 8-bits   | Address Resolution Message Type<br>For AR_MCAST_UNREGv6_B = 27 |
| AR.IPv6 | 128-bits | IPv6 multicast address unregistered                            |

The AR.IPv6 field is included in the AR\_MCAST\_UNREGv6\_B message so that the service node can perform multiple overlapping unregistrations.

### 5.5.4.11.3 IPv6 Packet format

#### 5.5.4.11.3.1 General

The following PDU formats are used for transferring IPv6 packets between service nodes.

### 5.5.4.11.3.2 No negotiated header compression

When no header compression takes place, the IPv6 packet is simply sent as it is, without any header. See Table 26.

**Table 26 – IPv6 Packet format without negotiated header compression**

| Name     | Length   | Description     |
|----------|----------|-----------------|
| IPv6.PKT | n-octets | The IPv6 Packet |

### 5.5.4.11.3.3 Header compression

When LOWPAN\_IPHC header compression takes place and the next header compression is negotiated, the UDP/IPv6 packet is sent as shown in Table 27.

Note, that these fields are not necessarily aligned to byte boundaries. For example, the IPv6.ncIPv6 field can be any number of bits. The IPv6.HC\_UDP field follows directly afterwards, without any padding. Padding is only applied at the end of the complete compressed UDP/IPv6 header such that the UDP data is byte aligned.

**Table 27 – UDP/IPv6 Packet format with LOWPAN\_IPHC header compression and LOWPAN\_NHC**

| Name        | Length     | Description                                                                                                   |
|-------------|------------|---------------------------------------------------------------------------------------------------------------|
| IPv6.IPHC   | 2-octet    | Dispatch + LOWPAN_IPHC encoding. With bit 5=1 indicating that the next is compressed, using LOWPAN_NHC format |
| IPv6.ncIPv6 | n.m-octets | Non-Compressed IPv6 fields (or elided)                                                                        |
| IPv6.HC_UDP | 1-octet    | Next header encoding                                                                                          |
| IPv6.ncUDP  | n.m-octets | Non-Compressed UDP fields                                                                                     |
| Padding     | 0.m-octets | Padding to byte boundary                                                                                      |
| IPv6.DATA   | n-octets   | UDP data                                                                                                      |

When the IPv6 packet contains data other than UDP the following packet format is used as shown in Table 28.

**Table 28 – IPv6 Packet format with LOWPAN\_IPHC negotiated header compression**

| Name        | Length     | Description                                                                      |
|-------------|------------|----------------------------------------------------------------------------------|
| IPv6.IPHC   | 2-octet    | HC encoding. Bits 5 contain 0 indicating the next header byte is not compressed. |
| IPv6.ncIPv6 | n.m-octets | Non-Compressed IPv6 fields                                                       |
| Padding     | 0.m-octets | Padding to byte boundary                                                         |
| IPv6.DATA   | n-octets   | IP Data                                                                          |

### 5.5.4.11.4 Connection data

#### 5.5.4.11.4.1 Overview

When a connection is established between service nodes for the transfer of IPv6 packets, data is also transferred in the connection request packets. This data allows the negotiation of compression and notification of the IPv6 address.

#### 5.5.4.11.4.2 Connection data from the initiator

Table 29 shows the connection data sent by the initiator.

**Table 29 – IPv6 Connection signalling data sent by the initiator**

| Name            | Length   | Description                                                                                                                                                                                                    |
|-----------------|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Reserved</i> | 6-bits   | Should be encoded as zero in this version of the convergence layer protocol                                                                                                                                    |
| Data.HCNH       | 2-bit    | Header compression negotiated<br>Data.HC = 0 – No compression requested<br>Data.HC = 1 – LOWPAN_NH<br>Data.HC = 2 – stateful address compression.<br>Data.HC = 3 – LOWPAN_NH and stateful address compression. |
| Data.IPv6       | 128-bits | IPv6 address of the initiator                                                                                                                                                                                  |

If the device accepts the connection, it should copy the Data.IPv6 address into a new table entry along with the negotiated Data.HC value.

#### 5.5.4.11.4.3 Connection data from the responder

Table 30 shows the connection data sent in response to the connection request.

**Table 30 – IPv6 Connection signalling data sent by the responder**

| Name            | Length | Description                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Reserved</i> | 6-bits | Should be encoded as zero in this version of the convergence layer protocol                                                                                                                                                                                                                                                                                                                                         |
| Data.HC         | 2-bit  | Header Compression negotiated<br>Data.HC = 0 – No compression requested:<br>NOTE – When stateless address compression is used all nodes shall support it. When the stateless address compression is not used then the node notify by this value, its compression capability.<br>Data.HC = 1 – LOWPAN_NH<br>Data.HC = 2 – Stateful address compression.<br>Data.HC = 3 – LOWPAN_NH and stateful address compression. |

All nodes support stateless address compression.

The next header compression scheme and stateful address compression can only be used when it is supported by both service nodes. The responder may only set Data.HC to the same value as the one received from the initiator or a value lower than the one received. When the same value is used, it indicates that the requested compression scheme has been negotiated and will be used for the connection. Setting Data.HC to a lower value allows the responder to deny the request for that header compression scheme.

## 6 Identification and addressing schemes

### 6.1 IEC 61334-4-32 profile addressing

#### 6.1.1 Overview

Three levels of addresses are defined in DLMS/COSEM PRIME IEC 61334-4-32 profile:

- at the MAC sublayer level, MAC addresses are processed for accessing the CL;

- at the IEC 61334-4-32 SSCS level, service node destination addresses and base node addresses for accessing the LLC sublayer;
- at the LLC sublayer, source and destination addresses hold the DLMS/COSEM Application layer Client SAP and Server SAP and vice versa depending on the direction of the communication flow.

### 6.1.2 MAC address

The MAC addresses are used to identify non-ambiguously the nodes in a subnetwork participating in the data exchange.

During data transmission, this MAC address is provided by the upper layer to the MAC sublayer and it is consumed by the MAC layer after inserting it in the PPDU.

During reception, the MAC sublayer checks if the packet is for this node and if so, it consumes the address and takes into account the concerned packet.

The full MAC addressing structure is the following:

| MSB     |        |         | LSB    |
|---------|--------|---------|--------|
| 48 bits | 8 bits | 14 bits | 9 bits |
| SNA     | LSID   | LNID    | LCID   |

Where:

- SNA is the subnetwork address, identified by the EUI-48 of the base node;
- LSID is the local switch identifier inside this subnetwork;
- LNID is the local node identifier of the related service node; and
- LCID is the connection identifier.

### 6.1.3 IEC 61334-4-32 SSCS addresses

During connection establishment, the base node provides to the service node its address at the SSCS level. This address identifies the SSCS access point.

NOTE Due to the fact that the IEC 61334-4-32 is embedded in the IEC 61334-4-32 SSCS, these addresses are not part of the frames exchanged; therefore they are not visible outside the device.

### 6.1.4 LLC addresses

The LLC addresses identify the client and the server SAPs at the application layer level. For the client, this address is the client Id, see Table 31. For the server, the SAP identifies the logical device concerned, see Table 32. These addresses are of one byte each.

NOTE Service nodes act as DLMS/COSEM servers. The base node acts as a DLMS/COSEM client or as a gateway for a client external to the PRIME subnetwork.

**Table 31 – Client service access point values**

| Value        | Purpose           |
|--------------|-------------------|
| 0            | No station        |
| 1            | Management client |
| 2... 0x0F    |                   |
| 0x10         | Public client     |
| 0x11... 0xFF |                   |

**Table 32 – Server service access point values**

| Value         | Purpose                   |
|---------------|---------------------------|
| 0             | No station                |
| 1             | Management logical device |
| 0x02... 0x0F  | Reserved for future use   |
| 0x10... 0x0FE |                           |
| 0xFF          | All station (Broadcast)   |

## 6.2 TCP-UDP/IPv4 profile addressing

As specified in IEC 62056-9-7:2013, Clause 6, the wrapper addresses are specified in IEC 62056-4-7:2015, 5.3.4, Table 1.

The addressing scheme is the one defined in the IEC 62056-9-7:2013, Figure 3. The devices are addressed by their respective IPv4 address, the COSEM application by the TCP or UDP port and the client and server by their SAP. Resolving the IPv4 address is the scope of the convergence layer.

## 6.3 TCP-UDP/IPv6 profile addressing

As defined in IEC 62056-9-7:2013, Clause 6 except that the IPv4 address is replaced by the IPv6 address. The wrapper addresses are specified in IEC 62056-4-7:2015, 5.3.4, Table 1. The devices are addressed by their IPv6 address. The other addresses are the same as for the IPv4 profile.

# 7 Specific consideration for the application layer services

## 7.1 Overview

This clause is applicable for all the 3 profiles: the IEC 61334-4-32 profile, the IPv4 and IPv6 ones.

The main feature of PRIME lower layers is the fact that the convergence layers are connection oriented. This means that before any ACSE or xDLMS APDUs can be exchanged, a data connection shall exist between the two devices.

## 7.2 Application Association (AA) establishment and release: ACSE services

### 7.2.1 AA establishment: IEC 61334-4-32 profile

In the IEC 61334-4-32 profile an AA can only be established when the two convergence layers have a data connection.

Table 33 shows the rules for establishing confirmed and unconfirmed AAs. According to this table, it is not allowed to request an xDLMS service in a confirmed way (Service\_Class = Confirmed) within an unconfirmed AA. This is prevented by the Client AL. Servers receiving such APDUs shall simply discard them, or shall send back a ConfirmedServiceError APDU or – if the feature is implemented – send back the optional ExceptionResponse APDU.

The Service\_Class parameter of the COSEM-OPEN service is linked to the response-allowed parameter of the xDLMS InitiateRequest APDU. When the COSEM-OPEN service is invoked with Service\_Class == Confirmed, the response-allowed parameter shall be set to TRUE, and the server is expected to respond. When the COSEM-OPEN service is invoked with Service\_Class == Unconfirmed, the response-allowed parameter shall be set to FALSE, and the server shall not send back a response.

**Table 33 – Application associations and data exchange in the IEC 61334-4-32 profile**

| Application association establishment  |                          |                                                           |                        | Data exchange           |                      |
|----------------------------------------|--------------------------|-----------------------------------------------------------|------------------------|-------------------------|----------------------|
| Protocol connection parameters         | COSEM-OPEN service class | Use                                                       | Type of established AA | Service class           | Use                  |
| Id:<br>LLC addresses,<br>MAC addresses | Confirmed                | Exchange AARQ/AARE APDU-s transported by DL-Data services | Confirmed              | Confirmed               | CL_432_Data services |
|                                        |                          |                                                           |                        | Unconfirmed             | CL_432_Data services |
|                                        | Unconfirmed              | Send AARQ transported by DL-Data services                 | Unconfirmed            | Confirmed (not allowed) | –                    |
|                                        |                          |                                                           |                        | Unconfirmed             | CL_432_Data services |

Only the COSEM-OPEN service has communication profile specific parameters, the Protocol\_Connection\_Parameters parameter. It contains the following data:

- Profile Identifier IEC 61334-4-32;
- Server\_Address service node MAC address;
- Server\_Id COSEM Logical Device Address;
- Client\_Address base node Address;
- Client\_Id COSEM Client application process identifier.

### 7.2.2 AA establishment: IP based profile

For IP based profiles IEC 62056-9-7:2013, 9.1, including the related Table 1 is fully applicable:

- when using TCP as transport layer, only confirmed AA is allowed. xDLMS services may then be confirmed or unconfirmed;
- when using UDP as transport layer, AAs may be confirmed or unconfirmed but confirmed xDLMS services cannot be used within unconfirmed AAs.

Only the COSEM-OPEN service has communication profile specific parameters, the Protocol\_Connection\_Parameters parameter. This contains the following data:

- Profile Identifier TCP/IP or UDP/IP;
- Server\_IP\_Address COSEM Physical Device IP Address;
- Server\_TCP\_or\_UDP\_Port The TCP or UDP port used for DLMS/COSEM;

- Server\_wrapper\_Port COSEM Logical Device Address;
- Client\_IP\_Address COSEM Client's Physical Device IP Address;
- Client\_TCP\_or\_UDP\_Port The TCP or UDP port used for DLMS/COSEM;
- Client\_wrapper\_Port COSEM application process identifier.

### 7.2.3 Application association release

#### 7.2.3.1 General

The IEC 61334-4-32 convergence layer, the IPv4 and the IPv6 convergence layers are connection oriented convergence layers. When the data connection is lost, the application association using this connection data has to be gracefully released.

#### 7.2.3.2 IEC 61334-4-32 profile

Due to the fact that the IEC 61334-4-32 LLC is a connectionless protocol, the use of the A-RELEASE services of the ACSE is mandatory for IEC 61334-4-32 profile.

#### 7.2.3.3 TCP-UDP/IP profile

In accordance with the IEC 62056-9-7:2013, the use of the A-RELEASE services of the ACSE is mandatory for TCP-UDP/IP profile.

### 7.3 xDLMS services

All the xDLMS services as defined in the IEC 62056-5-3 are applicable without any restriction.

The Service\_Class parameter of the xDLMS services is linked to the service-class bit of the Invoke-Id-And-Priority byte or Long-Invoke-Id-And-Priority. If the service is invoked with Service\_Class = Confirmed, the service-class bit shall be set to 1, otherwise it shall be set to 0.

### 7.4 Security mechanisms

#### 7.4.1 DLMS/COSEM security

DLMS/COSEM security applies at the application layer level. As a consequence, the application layer security does not depend on the structure of this communication profile. All the security mechanisms as defined in the IEC 62056-5-3 are applicable without any restrictions. The security suites and the security policies chosen and the PKI to use are project specific. They depend on the project specific companion specification.

#### 7.4.2 Lower layers security

In addition to the DLMS/COSEM security, the MAC layer also provides security features addressing confidentiality, data authenticity and integrity, and protection against replay attacks. These security features are out of the scope of this document.

### 7.5 Transferring long application messages

The maximum length of the APDUs is limited by the MaxAPDU size. When the size of the data to transport is greater than the MaxAPDU size, then the application layer block transfer shall be used.

Segmentation and reassembly capability is also available at the CPCS level. This segmentation and reassembly shall NOT interact with long application message management.

Their only purpose is to form the IEC 61334-4-32 SSCS, IPv4, or IPv6 SSCS PDU from the MAC PDUs.

#### **7.6 Media access, bandwidth and timing considerations**

There are no specific time considerations related to the medium and the lower layers which may affect the behaviour the operation of the DLMS/COSEM profile,

#### **7.7 Other considerations**

There are no other specific considerations regarding this profile.

### **8 Communication configuration and management**

The PRIME PIB attributes are mapped to attributes and methods of DLMS/COSEM interface classes. These may be accessed directly by the means of the PRIME management plane, or through the interface objects specified in IEC 62056-6-2:2017, 5.11.

### **9 The COSEM application process**

All the features as defined in the IEC 62056-5-3, IEC 62056-6-1 and IEC 62056-6-2 are applicable without any restrictions.

### **10 Additional considerations for the use of this profile**

This clause is irrelevant for this profile.

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## **Annex A**

(informative)

### **Examples**

#### **A.1 Data exchange between two IP communication peers**

This example shows the primitive exchange between a service node (192.168.0.100/24) and a base node when the former wants to exchange IP packets with a third service node (192.168.0.101/24) whose IP address is in the same IP subnetwork. This example concerning an IPv4 SSCS can be applied also to an IPv6 SSCS.

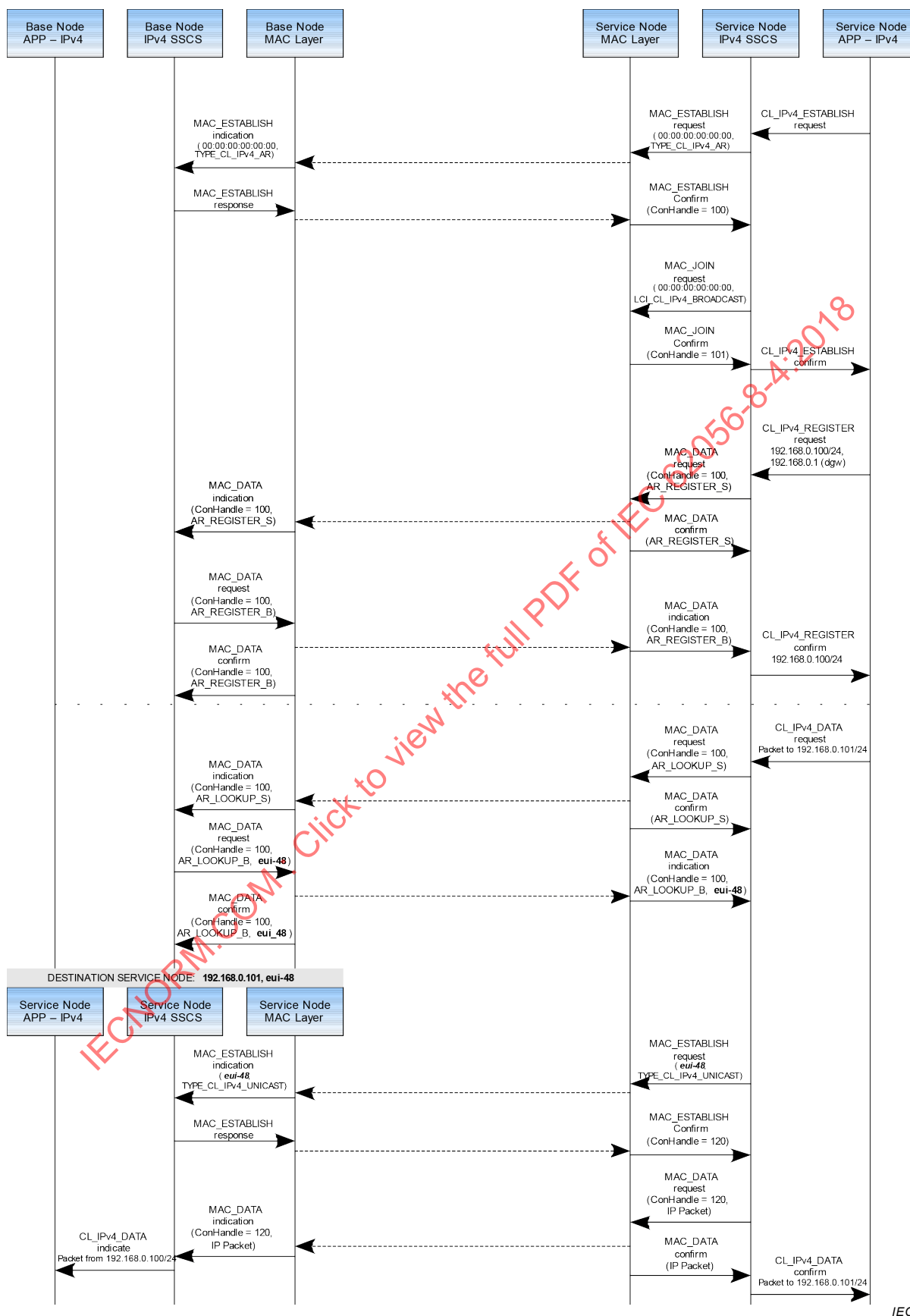
This example makes the following assumptions:

- Service node (192.168.0.100) IPv4 SSCS does not exist so, the service node needs to start an IPv4 SSCS and register its IP address with the base node prior to the exchange of IP packets.
- Service node (192.168.0.101) has already registered its IP address with the base node.

The steps illustrated in Figure A.1 are:

- a) The IPv4 layer of the service node (192.168.0.100) invokes the CL\_IPv4\_ESTABLISH.request primitive. To establish IPv4 SSCS, it is required:
  - 1) to establish a connection with the base node so all address resolution messages can be exchanged over it.
  - 2) to inform the service node MAC layer that IPv4 SSCS is ready to receive IPv4 broadcast packets. Note the difference between broadcast and multicast. To join a multicast group, the service node needs to inform the base node of the group it wants to join. This is illustrated in Clause A.2.
- b) The IPv4 layer, once the IPv4 SSCS is established, needs to register its IP address with the base node. To do so, it will use the already established connection.
- c) Whenever the IPv4 layer needs to deliver an IPv4 packet to a new destination IP address, the following two steps are to be done (in this example, the destination IP address is 192.168.0.101).
  - 1) As the IPv4 destination address is new, the IPv4 SSCS needs to request the EUI-48 associated to that IPv4 address. To do so, a lookup request message is sent to the base node.

Upon the reception of the EUI-48, a new connection (type = TYPE\_CL\_IPv4\_UNICAST) is established so that all IP packets to be exchanged between 192.168.0.100 and 192.168.0.101 will use that connection.

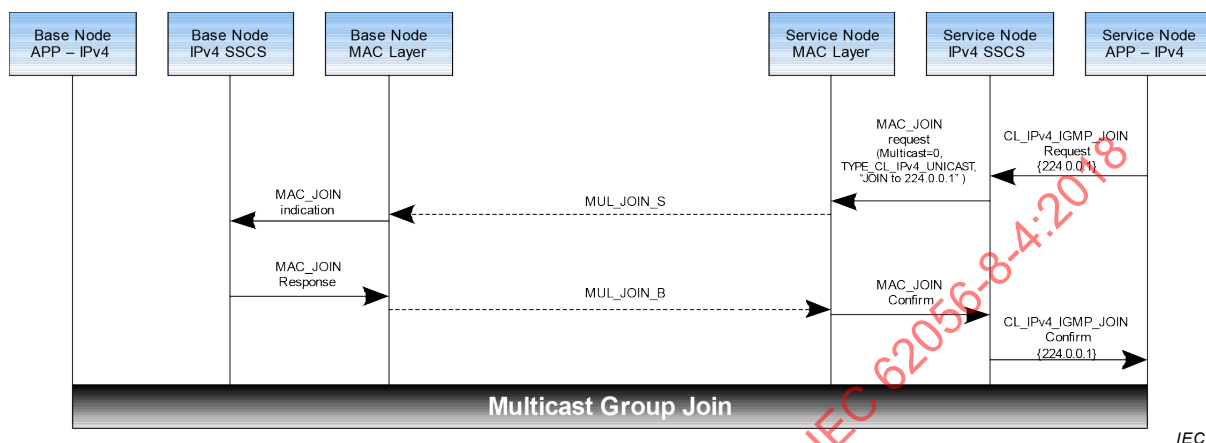


IEC

Figure A.1 – MSC of IPv4 SSCS services

## A.2 Joining a multicast group

Figure A.2 illustrates how a service node joins a multicast group. As mentioned before, the main difference between multicast and broadcast is related to the messages exchanged. For broadcast, the MAC layer will immediately issue a MAC\_JOIN.confirm primitive since it does not need to perform any end-to-end operation. For multicast, the MAC\_JOIN.confirm is only sent once the signalling between the service node and base node is complete.



IEC

Figure A.2 – Joining MSC IPv4 profile

## A.3 PRIME encoding examples

### ACSE APDUs and xDLMS APDUs carried by MAC frames using the IEC 61334-4-32 SSCS

In these examples, the following communication sequence is shown, when the DLMS/COSEM PRIME profile is used with the IEC 61334-4-32 SSCS.

- the initiator establishes an AA;
- it reads the time attribute of the Clock object;
- it reads load profile to show both PRIME segmentation and DLMS block transfer;
- the initiator releases AA.

The traces have been taken from a protocol analyser. DLMS/COSEM payload is highlighted in blue.

Open association on the Logical device LsapDest=0x01 and Management Client LsapSrc=0x01: MAC frame carrying an AARQ APDU

Hex:

```

0000  00 40 29 05 00 00 60 18 3c 87 07 00 90 01 01 60
0010  34 a1 09 06 07 60 85 74 05 08 01 01 8a 02 07 80
0020  8b 07 60 85 74 05 08 02 01 ac 08 80 06 31 32 33
0030  34 35 36 be 10 04 0e 01 00 00 00 06 5f 1f 04 00
0040  00 30 1d ff ff 63 b0 fb a5
  
```

PRIME

00.. .... = PRIME PDU Unused: 2bits. Must be always 00

..00 .... = PRIME PDU Header Type: Generic MAC header format GPDU

(0)

Generic MAC Header

.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0

```
.1... .... = PRIME Generic MAC HDR.DO field Downlink: 1
..00 0000 = PRIME Generic MAC HDR.LEVEL field, Header level: 0
PRIME Generic MAC checksum: 41
```

## GPDU Header

```
000. .... = PRIME PDU General Header reserved: 0
...0 .... = PRIME PDU General Header NAD: 0
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
0110 0000 0001 10.. = PRIME PDU General Header LNID: 6150
.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0011 1100 = PRIME PDU General Header length: 60
```

## GPDU ARQ

```
1... .... = PRIME PDU ARQ M: 1
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0111 = PRIME PDU ARQ PKTID: 7
0... .... = PRIME PDU ARQ M: 0
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0111 = PRIME PDU ARQ ACKID: 7
```

## GPDU Payload

```
PRIME PDU CRC: 1672543141 (63 b0 fb a5)
```

## PRIME 432

```
00.. .... = PRIME SAR Type: 0
..00 0000 = PRIME SAR Nseg: 0
1... .... = PRIME 432 one bit: 1
.00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
```

```
PRIME 432 data: 6034a1090607608574050801018a0207808b07608574...
```

## -- 432 Data explanation:

```
60 34 // AARQ
a1 09 06 07 60 85 74 05 08 01 01 // application-context-name
8a 02 07 80 // acse-requirements
8b 07 60 85 74 05 08 02 01 // mechanism-name
ac 08 80 06 31 32 33 34 35 36 // calling-authentication-value
be 10 04 0e 01 00 00 06 5f 1f 04 00 00 30 1d ff ff
// user-information xDLMS InititateRequest
```

Response: MAC frame carrying an AARE APDU

## Hex:

```
0000 00 00 ee 15 00 00 e0 08 31 bd 03 00 90 01 01 61
0010 29 a1 09 06 07 60 85 74 05 08 01 01 a2 03 02 01
0020 00 a3 05 a1 03 02 01 00 be 10 04 0e 08 00 06 5f
0030 1f 04 00 00 10 1d 00 f8 00 07 92 0f a2 d7
```

## PRIME

```
00.. .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0
.0.. .... = PRIME Generic MAC Downlink: 0
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 238
```

## GPDU Header

```
000. .... = PRIME PDU General Header reserved: 0
...1 .... = PRIME PDU General Header NAD: 1
.... 01.. = PRIME PDU General Header Prio: 1
```

```

.....0. = PRIME PDU General Header control: 0
.....1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.....0. = PRIME PDU General Header padding: 0
.....0 0011 0001 = PRIME PDU General Header length: 49

GPDU ARQ
1... .. = PRIME PDU ARQ M: 1
.0.. .. = PRIME PDU ARQ FLUSH: 0
..11 1101 = PRIME PDU ARQ PKTID: 61
0... .. = PRIME PDU ARQ M: 0
.0.. .. = PRIME PDU ARQ FLUSH: 0
..00 0011 = PRIME PDU ARQ ACKID: 3

GPDU Payload
PRIME PDU CRC: 2450498263 (92 0f a2 d7) at the end of frame
PRIME 432
00.. .. = PRIME SAR Type: 0
..00 0000 = PRIME SAR Nseg: 0
1... .. = PRIME 432 one bit: 1
.00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
PRIME 432 data: 6129a109060760857405080101a203020100a305a10302...

-- 432 Data explanation:

61 29 // AARE APDU
a1 09 06 07 60 85 74 05 08 01 01 // application-context-name
a2 03 02 01 00 // result
a3 05 a1 03 02 01 00 // result-source-diagnostic
be 10 04 0e 08 00 06 5f 1f 04 00 00 10 1d 00 f8 00 07
// user-information xDLMS-InitiateResponse

Read date and time, get-request

0000 00 40 29 05 00 00 e0 08 13 83 3e 00 90 01 01 c0
0010 01 c1 00 08 00 00 01 00 00 ff 02 00 7f 80 a6 4d

PRIME
00.. .. = PRIME PDU Unused: 0
..00 .. = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.....0000 0... .. = PRIME Generic MAC Header Reserved: 0
..1... .. = PRIME Generic MAC Downlink: 1
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 41

GPDU Header
000. .... = PRIME PDU General Header reserved: 0
...0 .... = PRIME PDU General Header NAD: 0
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.....1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.....0. = PRIME PDU General Header padding: 0
.....0 0001 0011 = PRIME PDU General Header length: 19

GPDU ARQ
1... .. = PRIME PDU ARQ M: 1
.0.. .. = PRIME PDU ARQ FLUSH: 0
..00 0011 = PRIME PDU ARQ PKTID: 3

```

```

0... .. = PRIME PDU ARQ M: 0
.0... .. = PRIME PDU ARQ FLUSH: 0
..11 1110 = PRIME PDU ARQ ACKID: 62
GPDU Payload
PRIME PDU CRC: 2139137613 (7f 80 a6 4d) at the end of frame
PRIME 432
00... .. = PRIME SAR Type: 0
..00 0000 = PRIME SAR Nseg: 0
1... .. = PRIME 432 one bit: 1
.00... .. = PRIME 432 Command: 0
...1 ... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
PRIME 432 data: c001c100080000010000ff0200

-- 432 Data explanation:

c0 01 // get-request, get-request-normal
c1 // invoke-id-and-priority
00 08 // class-id
00 00 01 00 00 ff // instance-id
02 // attribute-id
00

Read date and time, get-response

Hex:
0000 00 00 ee 15 00 00 e0 08 18 be 04 00 90 01 01 c4
0010 01 c1 00 09 0c 07 db 03 02 03 0a 34 08 ff 80 00
0020 04 23 1c aa 32

PRIME
00... .. = PRIME PDU Unused: 0
..00 ... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .. = PRIME Generic MAC Header Reserved: 0
.0... .. = PRIME Generic MAC Downlink: 0
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 238
GPDU Header
000... .. = PRIME PDU General Header reserved: 0
...1 ... = PRIME PDU General Header NAD: 1
... 01... = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0001 1000 = PRIME PDU General Header length: 24
GPDU ARQ
1... .. = PRIME PDU ARQ M: 1
.0... .. = PRIME PDU ARQ FLUSH: 0
..11 1110 = PRIME PDU ARQ PKTID: 62
0... .. = PRIME PDU ARQ M: 0
.0... .. = PRIME PDU ARQ FLUSH: 0
..00 0100 = PRIME PDU ARQ ACKID: 4
GPDU Payload
PRIME PDU CRC: 589081138 (23 1c aa 32) at the end of frame
PRIME 432
00... .. = PRIME SAR Type: 0
..00 0000 = PRIME SAR Nseg: 0

```

```

1... .. = PRIME 432 one bit: 1
.00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
PRIME 432 data: c401c100090c07db0302030a3408ff800004

```

-- 432 Data explanation:

```

c4 01 // get-response, get-response-normal
c1 // invoke-id-and-priority
00 09 0c // result, data, octet-string of 12
07db0302030a3408ff800004 // value

```

Read load profile, get-request with access selector:

Hex:

```

0000 00 40 29 05 00 00 e0 08 46 84 3f 00 90 01 01 c0
0010 01 c1 00 07 01 00 63 01 00 ff 02 01 01 02 04 02
0020 04 12 00 08 09 06 00 00 01 00 00 ff 0f 02 12 00
0030 00 09 0c 07 db 03 01 ff 10 00 00 ff 80 00 00 09
0040 0c 07 db 03 01 ff 17 00 00 ff 80 00 00 01 00 6b
0050 71 d8 42

```

PRIME

```

00.. .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .. = PRIME Generic MAC Header Reserved: 0
.1.. .... = PRIME Generic MAC Downlink: 1
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 41

```

GPDU Header

```

000. .... = PRIME PDU General Header reserved: 0
...0 .... = PRIME PDU General Header NAD: 0
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ..1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.... ..0. = PRIME PDU General Header padding: 0
.... .... 0100 0110 = PRIME PDU General Header length: 70

```

GPDU ARQ

```

1... .. = PRIME PDU ARQ M: 1
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0100 = PRIME PDU ARQ PKTID: 4
0... .. = PRIME PDU ARQ M: 0
.0.. .... = PRIME PDU ARQ FLUSH: 0
..11 1111 = PRIME PDU ARQ ACKID: 63

```

GPDU Payload

PRIME PDU CRC: 1802623042 (6b 71 d8 42)

PRIME 432

```

00.. .... = PRIME SAR Type: 0
..00 0000 = PRIME SAR Nseg: 0
1... .. = PRIME 432 one bit: 1
.00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1

```

PRIME 432 Source L\_SAP: 1

PRIME 432 data: c001c100070100630100ff020101020402041200080906...

-- 432 Data explanation:

```

c0 01          // get-request, get-request-normal
c1            // invoke-id-and-priority
00 07          // class-id
01 00 63 01 00 ff // instance-id
02            // attribute-id
01 01          // access-selector
02 04          // structure of 4
02 04          // structure of 4
12 00 08       // long-unsigned
09 06 00 00 01 00 00 ff // octet-string
0f 02          // integer
12 00 00       // long unsigned
09 0c 07 db 03 01 ff 10 00 00 ff 80 00 00 // octet-string
09 0c 07 db 03 01 ff 17 00 00 ff 80 00 00 // octet-string
01 00          // array of 0

```

Read load profile, get-response (first SAR segment):

Hex:

```

0000 00 00 ee 15 00 00 e0 08 4a ff 05 02 90 01 01 c4
0010 02 c1 00 00 00 00 01 00 81 c4 01 08 02 08 09 0c
0020 07 db 03 01 02 10 00 00 ff 80 00 04 11 00 06 00
0030 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00
0040 00 00 06 00 00 00 00 06 00 00 00 02 08 09 0c
0050 07 db 03 e1 3e b9 b6

```

PRIME

00.. .... = PRIME PDU Unused: 0

..00 .... = PRIME PDU Header Type: GPDU (0)

Generic MAC Header

.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0

.0... .... = PRIME Generic MAC Downlink: 0

..00 0000 = PRIME Generic MAC Header level: 0

PRIME Generic MAC checksum: 238

GPDU Header

000. .... = PRIME PDU General Header reserved: 0

...1 .... = PRIME PDU General Header NAD: 1

...01.. = PRIME PDU General Header Prio: 1

.... ..0. = PRIME PDU General Header control: 0

.... ...1 0000 0000 = PRIME PDU General Header LCID: 256

PRIME PDU General Header SID: 0

1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338

.... ..0. = PRIME PDU General Header padding: 0

.... ...0 0100 1010 = PRIME PDU General Header length: 74

GPDU ARQ

1... .... = PRIME PDU ARQ M: 1

.1.. .... = PRIME PDU ARQ FLUSH: 1

..11 1111 = PRIME PDU ARQ PKTID: 63

0... .... = PRIME PDU ARQ M: 0

.0... .... = PRIME PDU ARQ FLUSH: 0

..00 0101 = PRIME PDU ARQ ACKID: 5

GPDU Payload

PRIME PDU CRC: 3778984374 (e1 3e b9 b6)

PRIME 432

00.. .... = PRIME SAR Type: 0

..00 0010 = PRIME SAR Nseg: 2

1... .... = PRIME 432 one bit: 1

```
.00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
PRIME 432 data: c402c100000000010081c401080208090c07db03010210...
```

-- 432 Data explanation (raw-data value continues in following segments):

```
c4 02          // get-response, get-response-with-data-block
c1            // invoke-id-and-priority
00            // last-block (FALSE)
00 00 00 01    // block number (1)
00 81 c4       // result, raw-data
01 08 02 08 09 0c 07 db 03 01 02 10 00 00 ff 80 // value
00 04 11 00 06 00 00 00 00 06 00 00 00 00 06 00 // value
00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 // value
00 00 02 08 09 0c 07 db 03 //value
```

Read load profile, get-response (second SAR segment):

Hex:

```
0000 00 00 ee 15 00 00 e0 08 4a c0 05 40 01 02 11 00
0010 00 ff 80 00 04 11 00 06 00 00 00 00 06 00 00 00
0020 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00
0030 06 00 00 00 00 02 08 09 0c 07 db 03 01 02 12 00
0040 00 ff 80 00 04 11 00 06 00 00 00 00 06 00 00 00
0050 00 06 00 a0 4e 93 4d
```

PRIME

```
00.. .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0
.0.. .... = PRIME Generic MAC Downlink: 0
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 238
```

GPDU Header

```
000. .... = PRIME PDU General Header reserved: 0
...1 .... = PRIME PDU General Header NAD: 1
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ..1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0100 1010 = PRIME PDU General Header length: 74
```

GPDU ARQ

```
1... .... = PRIME PDU ARQ M: 1
.1.. .... = PRIME PDU ARQ FLUSH: 1
..00 0000 = PRIME PDU ARQ PKTID: 0
0... .... = PRIME PDU ARQ M: 0
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0101 = PRIME PDU ARQ ACKID: 5
```

GPDU Payload

PRIME PDU CRC: 2689504077 (a0 4e 93 4d)

PRIME 432

```
01.. .... = PRIME SAR Type: 1
..00 0000 = PRIME SAR Nseg: 0
PRIME 432 data: 0102110000ff8000041100060000000006000000000600...
```

-- 432 Data continues from previous segment.

Read load profile, get-response (last SAR segment):

Hex:

```
0000 00 00 ee 15 00 00 e0 08 47 81 05 81 00 00 00 06
0010 00 00 00 00 06 00 00 00 00 06 00 00 00 00 02 08
0020 09 0c 07 db 03 01 02 13 00 00 ff 80 00 04 11 00
0030 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06
0040 00 00 00 00 06 00 00 00 00 06 00 00 00 00 02 08
0050 7f 85 52 2d
```

PRIME

```
00.. .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0
.0.. .... = PRIME Generic MAC Downlink: 0
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 238
```

GPDU Header

```
000. .... = PRIME PDU General Header reserved: 0
...1 .... = PRIME PDU General Header MAD: 1
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0100 0111 = PRIME PDU General Header length: 71
```

GPDU ARQ

```
1... .... = PRIME PDU ARQ M: 1
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0001 = PRIME PDU ARQ PKTID: 1
0... .... = PRIME PDU ARQ M: 0
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0101 = PRIME PDU ARQ ACKID: 5
```

GPDU Payload

PRIME PDU CRC: 2139443757 (7f 85 52 2d)

PRIME 432

```
10.. .... = PRIME SAR Type: 2
..00 0001 = PRIME SAR Nseg: 1
PRIME 432 data: 0000000600000000006000000000600000000208090c07...
```

-- 432 Data continues from previous segment.

Read load profile, get-next-block:

Hex:

```
0000 00 40 29 05 00 00 e0 08 0d 85 02 00 90 01 01 c0
0010 02 c1 00 00 00 01 ce 1f 62 ab
```

PRIME

```
00.. .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0
.1.. .... = PRIME Generic MAC Downlink: 1
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 41
```

GPDU Header

```

000. .... = PRIME PDU General Header reserved: 0
...0 .... = PRIME PDU General Header NAD: 0
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0000 1101 = PRIME PDU General Header length: 13

GPDU ARQ
1... .... = PRIME PDU ARQ M: 1
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0101 = PRIME PDU ARQ PKTID: 5
0... .... = PRIME PDU ARQ M: 0
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0010 = PRIME PDU ARQ ACKID: 2

GPDU Payload
PRIME PDU CRC: 3458163371 (ce 1f 62 ab)
PRIME 432
00.. .... = PRIME SAR Type: 0
..00 0000 = PRIME SAR Nseg: 0
1... .... = PRIME 432 one bit: 1
.00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
PRIME 432 data: c002c100000001

-- 432 Data explanation:

c0 02          // get-request, get-request-for-next-data-block
c1             // invoke-id-and-priority
00000001       // block-number

```

Read load profile, get-response (block 2, first SAR segment):

```

Hex:
0000 00 00 ee 15 00 00 e0 08 48 c2 06 02 90 01 01 c4
0010 02 c1 01 00 00 00 02 00 81 be 09 0c 07 db 03 01
0020 02 14 00 00 ff 80 00 04 11 00 06 00 00 00 00 06
0030 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00
0040 00 00 00 06 00 00 00 00 02 08 09 0c 07 db 03 01
0050 02 84 28 b4 9a

```

```

PRIME
00.. .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0
.0.. .... = PRIME Generic MAC Downlink: 0
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 238

GPDU Header
000. .... = PRIME PDU General Header reserved: 0
...1 .... = PRIME PDU General Header NAD: 1
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338

```

```

.....0. = PRIME PDU General Header padding: 0
.....0 0100 1000 = PRIME PDU General Header length: 72
GPDU ARQ
1... .. = PRIME PDU ARQ M: 1
.1... .. = PRIME PDU ARQ FLUSH: 1
..00 0010 = PRIME PDU ARQ PKTID: 2
0... .. = PRIME PDU ARQ M: 0
.0... .. = PRIME PDU ARQ FLUSH: 0
..00 0110 = PRIME PDU ARQ ACKID: 6
GPDU Payload
PRIME PDU CRC: 2217260186 (84 28 b4 9a)
PRIME 432
00.. .. = PRIME SAR Type: 0
..00 0010 = PRIME SAR Nseg: 2
1... .. = PRIME 432 one bit: 1
.00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
PRIME 432 data: c402c101000000020081be090c07db030102140000ff80...

-- 432 Data explanation (raw-data value continues in following
segments):

c4 02 // get-response, get response-with-data-block
c1 // invoke-id-and-priority
01 // last-block (TRUE)
00 00 00 02 // block number (2)
00 81 be // result, raw-data
09 0c 07 db 03 01 02 14 00 00 ff 80 00 04 11 00 // value
06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 // value
00 00 00 00 06 00 00 00 00 06 00 00 00 00 02 08 // value
09 0c 07 db 03 01 02 // value

```

Read load profile, get-response (block 2, second SAR segment):

```

Hex:
0000 00 00 ee 15 00 00 e0 08 48 c3 06 40 15 00 00 ff
0010 80 00 04 11 00 06 00 00 00 00 06 00 00 00 00 06
0020 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00
0030 00 00 00 02 08 09 0c 07 db 03 01 02 16 00 00 ff
0040 80 00 04 11 00 06 00 00 00 00 06 00 00 00 00 06
0050 00 68 7f ac 47

```

PRIME

```

00.. .. = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .. = PRIME Generic MAC Header Reserved: 0
.0.. .... = PRIME Generic MAC Downlink: 0
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 238
GPDU Header
000. .... = PRIME PDU General Header reserved: 0
...1 .... = PRIME PDU General Header NAD: 1
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338

```

```

.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0100 1000 = PRIME PDU General Header length: 72
GPDU ARQ
1... .... = PRIME PDU ARQ M: 1
.1... .... = PRIME PDU ARQ FLUSH: 1
..00 0011 = PRIME PDU ARQ PKTID: 3
0... .... = PRIME PDU ARQ M: 0
.0... .... = PRIME PDU ARQ FLUSH: 0
..00 0110 = PRIME PDU ARQ ACKID: 6
GPDU Payload
PRIME PDU CRC: 1753197639 (68 7f ac 47)
PRIME 432
01... .... = PRIME SAR Type: 1
..00 0000 = PRIME SAR Nseg: 0
PRIME 432 data: 150000ff8000041100060000000006000000000600000000...
-- 432 Data continues from previous segment.

```

Read load profile, get-response (block 2, last SAR segment):

```

Hex:
0000 00 00 ee 15 00 00 e0 08 45 84 06 81 00 00 00 06
0010 00 00 00 00 06 00 00 00 00 06 00 00 00 00 02 08
0020 09 0c 07 db 03 01 02 17 00 00 ff 80 00 04 11 00
0030 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06
0040 00 00 00 00 06 00 00 00 00 06 00 00 00 00 e6 1c
0050 51 26

```

```

PRIME
00... .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0
.0... .... = PRIME Generic MAC Downlink: 0
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 238
GPDU Header
000. .... = PRIME PDU General Header reserved: 0
...1 .... = PRIME PDU General Header NAD: 1
.... 01... = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0100 0101 = PRIME PDU General Header length: 69
GPDU ARQ
1... .... = PRIME PDU ARQ M: 1
.0... .... = PRIME PDU ARQ FLUSH: 0
..00 0100 = PRIME PDU ARQ PKTID: 4
0... .... = PRIME PDU ARQ M: 0
.0... .... = PRIME PDU ARQ FLUSH: 0
..00 0110 = PRIME PDU ARQ ACKID: 6
GPDU Payload
PRIME PDU CRC: 3860615462 (e6 1c51 26)
PRIME 432
10... .... = PRIME SAR Type: 2
..00 0001 = PRIME SAR Nseg: 1
PRIME 432 data: 000000060000000006000000000600000000208090c07...
-- 432 Data continues from previous segment.

```

Release request, MAC frame carrying RLRQ APDU:

Hex:

```
0000 00 40 29 05 00 00 e0 08 08 86 05 00 90 01 01 62
0010 00 2e ef e9 a7
```

PRIME

```
00.. .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0
.1.. .... = PRIME Generic MAC Downlink: 1
..00 0000 = PRIME Generic MAC Header level: 0
PRIME Generic MAC checksum: 41
```

GPDU Header

```
000. .... = PRIME PDU General Header reserved: 0
...0 .... = PRIME PDU General Header NAD: 0
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0000 1000 = PRIME PDU General Header length: 8
```

GPDU ARQ

```
1... .... = PRIME PDU ARQ M: 1
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0110 = PRIME PDU ARQ PKTID: 6
0... .... = PRIME PDU ARQ M: 0
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0101 = PRIME PDU ARQ ACKID: 5
```

GPDU Payload

PRIME PDU CRC: 787474855 (2e ef e9 a7)

PRIME 432

```
00.. .... = PRIME SAR Type: 0
..00 0000 = PRIME SAR Nseg: 0
1... .... = PRIME 432 one bit: 1
..00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
PRIME 432 data: 6200
```

-- 432 Data explanation:

```
62 00 // release-request
```

Release response, MAC frame carrying RLRE APDU:

Hex:

```
0000 00 00 ee 15 00 00 e0 08 08 85 07 00 90 01 01 63
0010 00 a0 9d 21 92
```

PRIME

```
00.. .... = PRIME PDU Unused: 0
..00 .... = PRIME PDU Header Type: GPDU (0)
Generic MAC Header
.... 0000 0... .... = PRIME Generic MAC Header Reserved: 0
.0.. .... = PRIME Generic MAC Downlink: 0
..00 0000 = PRIME Generic MAC Header level: 0
```

```

PRIME Generic MAC checksum: 238
GPDU Header
000. .... = PRIME PDU General Header reserved: 0
...1 .... = PRIME PDU General Header NAD: 1
.... 01.. = PRIME PDU General Header Prio: 1
.... ..0. = PRIME PDU General Header control: 0
.... ...1 0000 0000 = PRIME PDU General Header LCID: 256
PRIME PDU General Header SID: 0
1110 0000 0000 10.. = PRIME PDU General Header LNID: 14338
.... ..0. = PRIME PDU General Header padding: 0
.... ...0 0000 1000 = PRIME PDU General Header length: 8
GPDU ARQ
1... .... = PRIME PDU ARQ M: 1
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0101 = PRIME PDU ARQ PKTID: 5
0... .... = PRIME PDU ARQ M: 0
.0.. .... = PRIME PDU ARQ FLUSH: 0
..00 0111 = PRIME PDU ARQ ACKID: 7
GPDU Payload
PRIME PDU CRC: 2694652306 (a0 9d 21 92)
PRIME 432
00.. .... = PRIME SAR Type: 0
..00 0000 = PRIME SAR Nseg: 0
1... .... = PRIME 432 one bit: 1
.00. .... = PRIME 432 Command: 0
...1 .... = PRIME 432 Command/Response: 1
.... 0000 = PRIME 432 Qualifier: 0
PRIME 432 Destination L_SAP: 1
PRIME 432 Source L_SAP: 1
PRIME 432 data: 6300

-- 432 Data explanation:
63 00 // release-response

```

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## **Annex B** (normative)

### **New COSEM interfaces classes and OBIS codes**

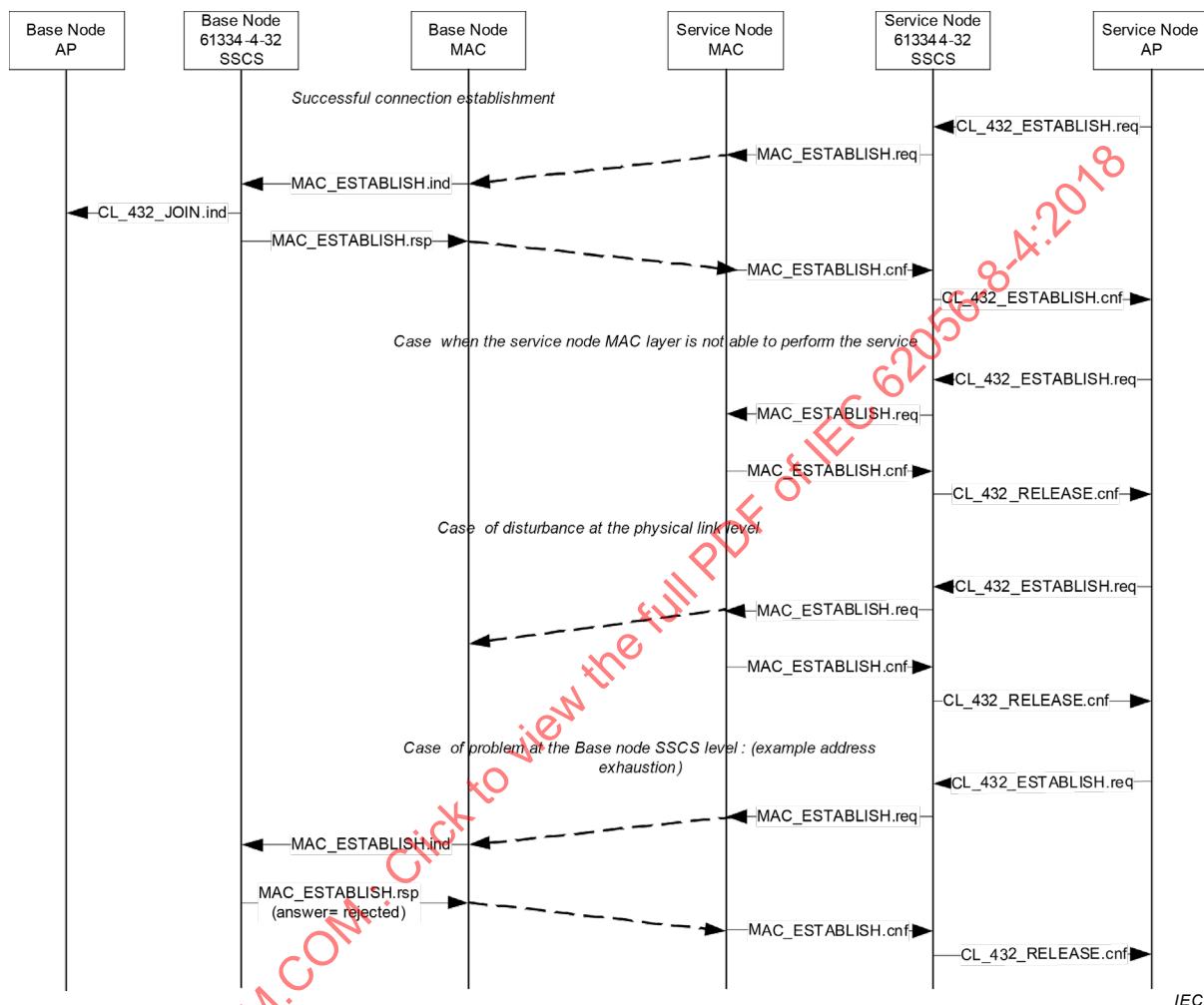
No new interface classes for configuration or new OBIS codes are required. All the configuration objects and OBIS codes are already defined in IEC 62056-6-2.

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## Annex C (informative)

### IEC 61334-4-32 profile: Error cases during connection establishment

Figure C.1 shows error cases that may occur during connection establishment.



IEC

Figure C.1 – Error cases during connection establishment

## Annex D (normative)

### Convergence layer constants

Table D.1 provides the defined TYPE values for the Convergence layers.

**Table D.1 – TYPE value assignment**

| TYPE name            | Value |
|----------------------|-------|
| TYPE_CL_IPv4_AR      | 1     |
| TYPE_CL_IPv4_UNICAST | 2     |
| TYPE_CL_432          | 3     |
| TYPE_CL_MGMT         | 4     |
| TYPE_CL_IPv6_AR      | 5     |
| TYPE_CL_IPv6_UNICAST | 6     |

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## Bibliography

EN 50065-1, *Signalling on low-voltage electrical installations in the frequency range 3 kHz to 148.5 kHz – Part 1: General requirements, frequency bands and electromagnetic disturbances*

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

IEEE 802.15.4:2006, *IEEE Standard for Information technology – Telecommunication and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 15.4: Wireless Medium Access (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Network (WPANs)*

---

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## SOMMAIRE

|                                                                                                |     |
|------------------------------------------------------------------------------------------------|-----|
| AVANT-PROPOS .....                                                                             | 83  |
| INTRODUCTION.....                                                                              | 85  |
| 1 Domaine d'application .....                                                                  | 86  |
| 2 Références normatives .....                                                                  | 86  |
| 3 Termes, définitions et termes abrégés .....                                                  | 88  |
| 3.1 Termes et définitions .....                                                                | 88  |
| 3.2 Termes abrégés .....                                                                       | 88  |
| 4 Environnements de communication ciblés.....                                                  | 90  |
| 5 Utilisation des couches de communication pour ce profil .....                                | 91  |
| 5.1 Informations relatives à l'utilisation de la norme spécifiant les couches inférieures..... | 91  |
| 5.2 Structure des profils de communication.....                                                | 92  |
| 5.2.1 Présentation .....                                                                       | 92  |
| 5.2.2 Profil IEC 61334-4-32 .....                                                              | 93  |
| 5.2.3 Profil TCP-UDP/IPv4 .....                                                                | 93  |
| 5.2.4 Profil TCP-UDP/IPv6 .....                                                                | 93  |
| 5.3 Couches de protocole inférieures et utilisation .....                                      | 93  |
| 5.3.1 Généralités .....                                                                        | 93  |
| 5.3.2 Couche physique .....                                                                    | 93  |
| 5.3.3 Couche MAC .....                                                                         | 94  |
| 5.4 Mise en correspondance de services et couches de convergence .....                         | 96  |
| 5.4.1 Présentation .....                                                                       | 96  |
| 5.4.2 SSCS IEC 61334-4-32 LLC .....                                                            | 96  |
| 5.4.3 SSCS IPv4 .....                                                                          | 98  |
| 5.4.4 SSCS IPv6 .....                                                                          | 100 |
| 5.5 Gestion des enregistrements et des connexions .....                                        | 101 |
| 5.5.1 Présentation .....                                                                       | 101 |
| 5.5.2 Profil IEC 61334-4-32 .....                                                              | 101 |
| 5.5.3 Profil TCP-UDP/IPv4 .....                                                                | 105 |
| 5.5.4 Profil TCP-UDP/IPv6 .....                                                                | 119 |
| 6 Identification et plan d'adressage .....                                                     | 135 |
| 6.1 Adressage de profil IEC 61334-4-32 .....                                                   | 135 |
| 6.1.1 Présentation .....                                                                       | 135 |
| 6.1.2 Adresse MAC .....                                                                        | 135 |
| 6.1.3 Adresses IEC 61334-4-32 SSCS.....                                                        | 136 |
| 6.1.4 Adresses LLC .....                                                                       | 136 |
| 6.2 Adressage de profil TCP-UDP/IPv4.....                                                      | 136 |
| 6.3 Adressage de profil TCP-UDP/IPv6.....                                                      | 137 |
| 7 Considérations particulières relatives aux services de couche application .....              | 137 |
| 7.1 Présentation .....                                                                         | 137 |
| 7.2 Établissement et libération d'une association d'applications (AA): Services ACSE.....      | 137 |
| 7.2.1 Établissement de l'AA: Profil IEC 61334-4-32 .....                                       | 137 |
| 7.2.2 Établissement de l'AA: Profil basé sur IP .....                                          | 138 |
| 7.2.3 Libération d'associations d'applications .....                                           | 138 |
| 7.3 Services xDLMS .....                                                                       | 139 |
| 7.4 Mécanismes de sécurité.....                                                                | 139 |

|                                                                                              |                                                                                                |     |
|----------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|-----|
| 7.4.1                                                                                        | Sécurité DLMS/COSEM .....                                                                      | 139 |
| 7.4.2                                                                                        | Sécurité des couches inférieures .....                                                         | 139 |
| 7.5                                                                                          | Transfert de longs messages d'application.....                                                 | 139 |
| 7.6                                                                                          | Considérations relatives à l'accès au support, à la bande passante et à la temporisation ..... | 139 |
| 7.7                                                                                          | Autres considérations .....                                                                    | 139 |
| 8                                                                                            | Configuration et gestion des communications .....                                              | 139 |
| 9                                                                                            | Processus d'application COSEM.....                                                             | 139 |
| 10                                                                                           | Considérations supplémentaires pour l'utilisation de ce profil .....                           | 140 |
| Annexe A (informative)                                                                       | Exemples .....                                                                                 | 141 |
| A.1                                                                                          | Échange de données entre deux homologues de communication IP.....                              | 141 |
| A.2                                                                                          | Intégration d'un groupe multicast .....                                                        | 143 |
| A.3                                                                                          | Exemples de codage PRIME.....                                                                  | 143 |
| Annexe B (normative)                                                                         | Nouvelles classes d'interface COSEM et nouveaux codes OBIS.....                                | 156 |
| Annexe C (informative)                                                                       | Profil IEC 61334-4-32: Cas d'erreur lors de l'établissement de connexion.....                  | 157 |
| Annexe D (normative)                                                                         | Constantes de couche de convergence .....                                                      | 158 |
| Bibliographie.....                                                                           |                                                                                                | 159 |
| Figure 1 – Architecture de communication .....                                               |                                                                                                | 91  |
| Figure 2 – Architectures du profil de communication OFDM PLC PRIME.....                      |                                                                                                | 92  |
| Figure 3 – Services SSCS IEC 61334-4-32 .....                                                |                                                                                                | 97  |
| Figure 4 – MSC pour les services de données dans le cas du référencement de nom logique..... |                                                                                                | 98  |
| Figure 5 – SSCS IEC 61334-4-32 .....                                                         |                                                                                                | 101 |
| Figure 6 – MSC pour les services SSCS IEC 61334-4-32.....                                    |                                                                                                | 105 |
| Figure 7 – Service SSCS IPv4 .....                                                           |                                                                                                | 107 |
| Figure A.1 – MSC des services SSCS IPv4.....                                                 |                                                                                                | 142 |
| Figure A.2 – Intégration d'un profil MSC IPv4.....                                           |                                                                                                | 143 |
| Figure C.1 – Cas d'erreur lors de l'établissement de connexion .....                         |                                                                                                | 157 |
| Tableau 1 – Valeurs du paramètre Result pour les services SSCS.....                          |                                                                                                | 104 |
| Tableau 2 – Format de message AR_REGISTER_S.....                                             |                                                                                                | 115 |
| Tableau 3 – Format de message AR_REGISTER_B.....                                             |                                                                                                | 115 |
| Tableau 4 – Format de message AR_UNREGISTER_S.....                                           |                                                                                                | 115 |
| Tableau 5 – Format de message AR_MCAST_REG_S.....                                            |                                                                                                | 115 |
| Tableau 6 – Format de message AR_MCAST_REG_B.....                                            |                                                                                                | 116 |
| Tableau 7 – Format de message AR_MCAST_UNREG_S.....                                          |                                                                                                | 116 |
| Tableau 8 – Format de message AR_MCAST_UNREG_B.....                                          |                                                                                                | 116 |
| Tableau 9 – Format de message AR_LOOKUP_S.....                                               |                                                                                                | 117 |
| Tableau 10 – Format de message AR_LOOKUP_B.....                                              |                                                                                                | 117 |
| Tableau 11 – Format de paquet IPv4 sans compression d'en-tête négocié.....                   |                                                                                                | 117 |
| Tableau 12 – Format de paquet IPv4 avec compression d'en-tête VJ.....                        |                                                                                                | 118 |
| Tableau 13 – Données de connexion envoyées par l'initiateur .....                            |                                                                                                | 118 |
| Tableau 14 – Données de connexion envoyées par le répondeur .....                            |                                                                                                | 119 |

|                                                                                                       |     |
|-------------------------------------------------------------------------------------------------------|-----|
| Tableau 15 – Entrée du tableau de la SSCS IPv6 .....                                                  | 123 |
| Tableau 16 – Mise en correspondance de la précédente IPv6 avec la priorité PRIME<br>MAC .....         | 124 |
| Tableau 17 – Format de message AR_REGISTERv6_S .....                                                  | 130 |
| Tableau 18 – Format de message AR_REGISTERv6_B .....                                                  | 130 |
| Tableau 19 – Format de message AR_UNREGISTERv6_S .....                                                | 131 |
| Tableau 20 – Format de message AR_UNREGISTERv6_B .....                                                | 131 |
| Tableau 21 – Format de message AR_LOOKUPv6_S .....                                                    | 131 |
| Tableau 22 – Format de message AR_LOOKUPv6_B .....                                                    | 132 |
| Tableau 23 – Format de message AR_MCAST_REGv6_S .....                                                 | 132 |
| Tableau 24 – Format de message AR_MCAST_REGv6_B .....                                                 | 132 |
| Tableau 25 – Format de message AR_MCAST_UNREGv6_B .....                                               | 133 |
| Tableau 26 – Format de paquet IPv6 sans compression d'en-tête négociée.....                           | 133 |
| Tableau 27 – Format de paquet UDP/IPv6 avec compression d'en-tête<br>LOWPAN_IPHC et LOWPAN_NHC .....  | 133 |
| Tableau 28 – Format de paquet IPv6 avec compression d'en-tête LOWPAN_IPHC<br>négociée.....            | 134 |
| Tableau 29 – Données de signalisation de connexion IPv6 envoyées par l'initiateur .....               | 134 |
| Tableau 30 – Données de signalisation de connexion IPv6 envoyées par le répondeur.....                | 135 |
| Tableau 31 – Valeurs de points d'accès de service client.....                                         | 136 |
| Tableau 32 – Valeurs de points d'accès de service serveur .....                                       | 136 |
| Tableau 33 – Associations d'applications et échange de données dans le profil<br>IEC 61334-4-32 ..... | 137 |
| Tableau D.1 – Attribution de valeur TYPE.....                                                         | 158 |

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

**ÉCHANGE DES DONNÉES DE COMPTAGE DE L'ÉLECTRICITÉ –  
LA SUITE DLMS/COSEM –****Partie 8-4: Profils de communication pour réseaux de voisinage OFDM  
PLC PRIME à bande étroite**

## AVANT-PROPOS

- 1) La Commission Électrotechnique Internationale (IEC) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de l'IEC). L'IEC a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. À cet effet, l'IEC – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de l'IEC"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'IEC, participent également aux travaux. L'IEC collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de l'IEC concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de l'IEC intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de l'IEC se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de l'IEC. Tous les efforts raisonnables sont entrepris afin que l'IEC s'assure de l'exactitude du contenu technique de ses publications; l'IEC ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
- 4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de l'IEC s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de l'IEC dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de l'IEC et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) L'IEC elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de l'IEC. L'IEC n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
- 6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
- 7) Aucune responsabilité ne doit être imputée à l'IEC, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de l'IEC, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de l'IEC ou de toute autre Publication de l'IEC, ou au crédit qui lui est accordé.
- 8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de l'IEC peuvent faire l'objet de droits de brevet. L'IEC ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Commission Électrotechnique Internationale (IEC) attire l'attention sur le fait qu'il est déclaré que la conformité avec les dispositions de la présente norme internationale peut impliquer l'utilisation d'un service de maintenance concernant la pile de protocoles sur laquelle repose la présente norme IEC 62056-8-4.

L'IEC ne prend pas position quant à la preuve, à la validité et à la portée de ce service de maintenance.

Le fournisseur du service de maintenance a donné l'assurance à l'IEC qu'il consent à fournir les services aux demandeurs du monde entier selon des termes et conditions raisonnables et non discriminatoires. À ce propos, la déclaration du fournisseur du service de maintenance est enregistrée à l'IEC. Des informations peuvent être demandées à:

PRIME Alliance  
2-12, Avenue de la Renaissance  
1000, Bruxelles/(BE)

www.prime-alliance.org.com

La Norme internationale IEC 62056-8-4 a été établie par le comité d'études 13 de l'IEC: Comptage et pilotage de l'énergie électrique.

Le texte de cette norme est issu des documents suivants:

| CDV         | Rapport de vote |
|-------------|-----------------|
| 13/1749/CDV | 13/1763/RVC     |

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Une liste de toutes les parties de la série IEC 62056, publiées sous le titre général *Échange des données de comptage de l'électricité – La suite DLMS/COSEM*, peut être consultée sur le site web de l'IEC.

Cette publication a été rédigée selon les Directives ISO/IEC, Partie 2.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de l'IEC sous "<http://webstore.iec.ch>", dans les données relatives à la publication recherchée. À cette date, la publication sera

- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

**IMPORTANT – Le logo "colour inside" qui se trouve sur la page de couverture de cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.**

## INTRODUCTION

Comme cela a été défini dans l'IEC 62056-1-0, la suite IEC 62056 DLMS/COSEM fournit des normes spécifiques de profil de communication pour les supports de communication destinés au comptage intelligent.

Ces normes de profil de communication spécifient la manière dont le modèle de données COSEM et la couche application DLMS/COSEM peuvent être utilisés sur les couches de protocole inférieures spécifiques au support de communication.

Les normes de profil de communication sont liées aux normes de communication faisant partie intégrante de la suite IEC 62056 DLMS/COSEM ou à une autre norme de communication ouverte.

La présente Norme internationale spécifie les profils de communication utilisant la Recommandation UIT-T G.9904:2012 *Émetteurs-récepteurs OFDM à bande étroite utilisant les courants porteurs en ligne – PRIME*. Elle s'applique aux dispositifs installés sur le réseau de voisinage.

Elle suit les règles définies dans l'IEC 62056-5-3:2017, Annexe A, et dans l'IEC 62056-1-0 ainsi que les recommandations de l'IEC TS 62056-1-1 quant à sa structure.

Le profil de communication spécifié dans le présent document s'appuie sur les résultats du projet européen OPEN Meter, Topic Energy 2008.7.1.1, N° 226369 ([www.openmeter.com](http://www.openmeter.com)) et a été élaboré par le groupe de travail PRIME Alliance Technical Working Group ([www.prime-alliance.org](http://www.prime-alliance.org)).

IECNORM.COM : Click to view the full PDF of IEC 62056-8-4:2018

## ÉCHANGE DES DONNÉES DE COMPTAGE DE L'ÉLECTRICITÉ – LA SUITE DLMS/COSEM –

### Partie 8-4: Profils de communication pour réseaux de voisinage OFDM PLC PRIME à bande étroite

#### 1 Domaine d'application

La présente partie de l'IEC 62056 spécifie les profils de communication DLMS/COSEM pour les réseaux de voisinage OFDM utilisant les courants porteurs en ligne PRIME utilisant la modulation conformément à la Recommandation UIT-T G.9904:2012.

Trois profils de communication sont spécifiés:

- un profil utilisant la couche LLC IEC 61334-4-32;
- un profil utilisant TCP-UDP/IPv4;
- un profil utilisant TCP-UDP/IPv6.

#### 2 Références normatives

Les documents suivants cités dans le texte constituent, pour tout ou partie de leur contenu, des exigences du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

IEC 61334-4-32:1996, *Automatisation de la distribution à l'aide de systèmes de communication à courants porteurs – Partie 4: Protocoles de communication de données – Section 32: Couche liaison de données – Contrôle de liaison logique (LLC)*

IEC 61334-4-511:2000, *Automatisation de la distribution à l'aide de systèmes de communication à courants porteurs – Partie 4-511: Protocoles de communication de données – Administration de systèmes – Protocole CIASE*

IEC 62056-1-0, *Échange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 1-0: Cadre de normalisation du comptage intelligent*

IEC TS 62056-1-1, *Electricity metering data exchange – The DLMS/COSEM suite – Part 1-1: Template for DLMS/COSEM communication profile standards* (disponible en anglais seulement)

IEC 62056-4-7:2015, *Échange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 4-7: Couche transport DLMS/COSEM pour réseaux IP*

IEC 62056-5-3:2017, *Echange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 5-3: Couche application DLMS/COSEM*

IEC 62056-6-1, *Echange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 6-1: Système d'identification des objets (OBIS)*

IEC 62056-6-2:2017, *Echange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 6-2: Classes d'interfaces COSEM*

IEC 62056-9-7:2013, *Échange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 9-7: Profil de communication pour réseaux TCP-UDP/IP*

Recommandation UIT-T G.9904:2012, *Série G: Systèmes et supports de transmission, systèmes et réseaux numériques Réseaux d'accès – Réseaux dans les locaux. Émetteurs-récepteurs OFDM à bande étroite utilisant les courants porteurs en ligne – PRIME*

STD0005 – Internet Protocol

Auteur: J. Postel

Date: Septembre 1981

Également: RFC0791, RFC0792, RFC0919, RFC0922, RFC0950, RFC1112

Disponible à l'adresse: <http://www.ietf.org/rfc/rfc0791.txt>

STD0006 – User Datagram Protocol

Auteur: J. Postel

Date: jeudi 28 août 1980

Également: RFC 768

Disponible à l'adresse: <http://www.ietf.org/rfc/rfc0768.txt>

STD0007 – Transmission Control Protocol

Auteur: J. Postel

Date: Septembre 1981

Disponible à l'adresse: <http://www.ietf.org/rfc/rfc0793.txt>

RFC 1144, Compressing TCP/IP Headers for Low Speed serial Link

Auteur: V. Jacobson

Date: Février 1990

Disponible à l'adresse <https://tools.ietf.org/rfc/rfc1144.txt>

RFC 2460, Internet Protocol, Version 6 (IPv6) Specification

Auteurs: S. Deering, Cisco, R. Hinden Nokia

Date: Décembre 1998

Disponible à l'adresse: <http://www.ietf.org/rfc/rfc2460.txt>

RFC 2464, Transmission of IPv6 Packets over Ethernet Networks

Auteur: M. Crawford Fermilab.

Date: Décembre 1998

Disponible à l'adresse: <http://www.ietf.org/rfc/rfc2464.txt>

RFC 3315, Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

Auteurs: R. Droms, E J. Bound, B. Volz, T. Lemon, C. Perkins, M. Carney

Date: Juillet 2003

Disponible à l'adresse: [www.ietf.org/rfc/rfc3315.txt](http://www.ietf.org/rfc/rfc3315.txt)

RFC 4291, IP Version 6 Addressing Architecture

Auteurs: R. Hinden Nokia, S. Deering Cisco Systems

Date: Février 2006

Disponible à l'adresse: <http://www.ietf.org/rfc/rfc4291.txt>

RFC 4862, IPv6 Stateless Address Configuration

Auteurs: S. Thomson, Cisco, T. Narten IBM, T. Jinmei, Toshiba

Date: Septembre 2007.

Disponible à l'adresse: [www.ietf.org/rfc/rfc4862.txt](http://www.ietf.org/rfc/rfc4862.txt)

RFC 6282, Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks

Auteurs: J. Hui, Ed. Arch Rock Corporation P. Thubert Cisco

Date: Septembre 2011.

Disponible à l'adresse: <http://www.ietf.org/rfc/rfc6282.txt>

### 3 Termes, définitions et termes abrégés

#### 3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

L'ISO et l'IEC tiennent à jour des bases de données terminologiques destinées à être utilisées en normalisation, consultables aux adresses suivantes:

- IEC Electropedia: disponible à l'adresse <http://www.electropedia.org/>
- ISO Online browsing platform: disponible à l'adresse <http://www.iso.org/obp>

NOTE Voir également la Recommandation UIT-T G.9904:2012.

##### 3.1.1

##### **nœud de base**

nœud principal qui commande et gère les ressources d'un sous-réseau

##### 3.1.2

##### **enregistrement**

processus par lequel un nœud de service est accepté comme membre du sous-réseau, auquel un LNID est attribué

##### 3.1.3

##### **nœud de service**

nœud d'un sous-réseau qui n'est pas un nœud de base

##### 3.1.4

##### **annulation d'enregistrement**

processus par lequel un nœud de service quitte un sous-réseau

#### 3.2 Termes abrégés

| Abréviation | Signification                                                                      |
|-------------|------------------------------------------------------------------------------------|
| AA          | Association d'Applications                                                         |
| AARE        | Application Association Response (réponse d'association d'applications)            |
| AARQ        | Application Association Request (demande d'association d'applications)             |
| ACSE        | Application Control Service Element (élément de service de contrôle d'application) |
| AL          | Application layer (Couche application)                                             |
| AP          | Application Process (processus d'application)                                      |
| APDU        | Application Protocol Data Unit (Unité de données de protocole d'application)       |
| ARQ         | Automatic Repeat Request (requête de répétition automatique)                       |
| CL          | Convergence Layer (couche de convergence)                                          |
| .cnf        | Confirm service primitive (primitive de service de confirmation)                   |

| <b>Abréviation</b> | <b>Signification</b>                                                                                           |
|--------------------|----------------------------------------------------------------------------------------------------------------|
| COSEM              | Companion Specification for Energy Metering (spécification d'accompagnement pour le comptage de l'énergie)     |
| CPCS               | Common Part Convergence Sublayer (partie commune de la Sous-couche de convergence)                             |
| CSMA/CA            | Carrier Sense Multiple Access – Collision Avoidance                                                            |
| D8PSK              | Differential Eight Phase Shift Keying (modulation par déplacement de phase différentielle à huit états)        |
| DBPSK              | Differential Binary Phase Shift Keying (Modulation différentielle par déplacement de phase bivalente)          |
| DGW                | Default GateWay (passerelle par défaut)                                                                        |
| DHCP               | Dynamic Host Configuration Protocol (protocole de configuration d'hôte dynamique)                              |
| DLMS               | Device Language Message Specification (spécification de message de langage de dispositif)                      |
| DQPSK              | Differential Quaternary Phase Shift Keying (Modulation différentielle par déplacement de phase quadrivalente)  |
| EUI-48             | 48-bit Extended Unique Identifier (identifiant unique étendu 48 bits)                                          |
| FU                 | Firmware Upgrade (mise à niveau de micrologiciel)                                                              |
| FW                 | Firmware (micrologiciel)                                                                                       |
| IANA               | Internet Assigned Numbers Authority (Organisation chargée de la gestion de l'espace d'adressage IP d'Internet) |
| IGMP               | Internet Group Management Protocol (protocole de gestion de groupe Internet)                                   |
| .ind               | Indication service primitive (primitive de service d'indication)                                               |
| IP                 | Internet Protocol (Protocole Internet)                                                                         |
| IPv4               | Internet Protocol, version 4 (Protocole Internet, version 4)                                                   |
| IPv6               | Internet Protocol, version 6 (Protocole Internet version 6)                                                    |
| LCID               | Local Connection Identifier (identifiant de connexion local)                                                   |
| LD                 | Logical Device (dispositif logique)                                                                            |
| LLC                | Logical Link Control (commande de liaison logique) (sous-couche)                                               |
| LNID               | Local Node Identifier (identifiant de nœud local)                                                              |
| MAC                | Medium Access Control (contrôle d'accès au support), entité de sous-couche MAC                                 |
| MLME               | MAC Layer Management Entity (entité de gestion de couche MAC)                                                  |
| MPDU               | MAC Protocol Data Unit (unité de données de protocole MAC)                                                     |
| NAT                | Network Address Translation (traduction d'adresse de réseau)                                                   |
| NHC                | Next Header Compression (compression d'en-tête suivant)                                                        |
| NL                 | Noise Level (niveau de bruit)                                                                                  |
| OBIS               | OBject Identification System (système d'identification d'objet)                                                |
| OFDM               | Orthogonal Frequency Division Multiplexing (multiplexage par répartition orthogonale de la fréquence)          |
| OSI                | Open Systems Interconnection (interconnexion de systèmes ouverts)                                              |

| Abréviation | Signification                                                                            |
|-------------|------------------------------------------------------------------------------------------|
| PHY         | Physical Layer entity (entité de couche physique)                                        |
| PLC         | Power Line Communication (courant porteur en ligne)                                      |
| PIB         | PLC Information Base (base d'informations PLC)                                           |
| PLME        | Physical Layer Management Entity (entité de gestion de couche physique)                  |
| PPDU        | PHY Protocol Data Unit (unité de données de protocole de couche Physique)                |
| .req        | Request service primitive (primitive de service de requête)                              |
| RFC         | Request For Comment (demande de commentaires)                                            |
| .rsp        | Response service primitive (primitive de service de réponse)                             |
| SDU         | Service Data Unit (unité de données de service)                                          |
| SID         | Switch Identifier (identifiant de commutation)                                           |
| SNA         | Subnetwork Address (adresse de sous-réseau)                                              |
| SNR         | Signal-to-Noise Ratio (rapport signal sur bruit)                                         |
| SSCS        | Service Specific Convergence Sublayer (sous-couche de convergence spécifique au service) |
| TCP         | Transmission Control Protocol (Protocole de contrôle de transmission)                    |
| TOS         | Type Of Service (type de service)                                                        |
| UDP         | User Datagram Protocol (Protocole de datagramme utilisateur)                             |
| xDLMS_ASE   | extended DLMS Application Service Element (élément de service d'application DLMS étendu) |
| ZCT         | Zero Crossing Time (heure de passage à zéro)                                             |

#### 4 Environnements de communication ciblés

Les profils de communication DLMS/COSEM pour réseaux de voisinage OFDM PLC PRIME à bande étroite sont destinés à l'échange de données éloignées sur des réseaux de voisinage (NN) entre les points d'accès aux réseaux de voisinage (NNAP) et les points d'accès au réseau local (LNAPs) ou les dispositifs terminaux grâce à la technologie OFDM PLC sur le réseau de distribution d'électricité basse tension comme support de communication. L'architecture fonctionnelle de référence est représentée à la Figure 1.

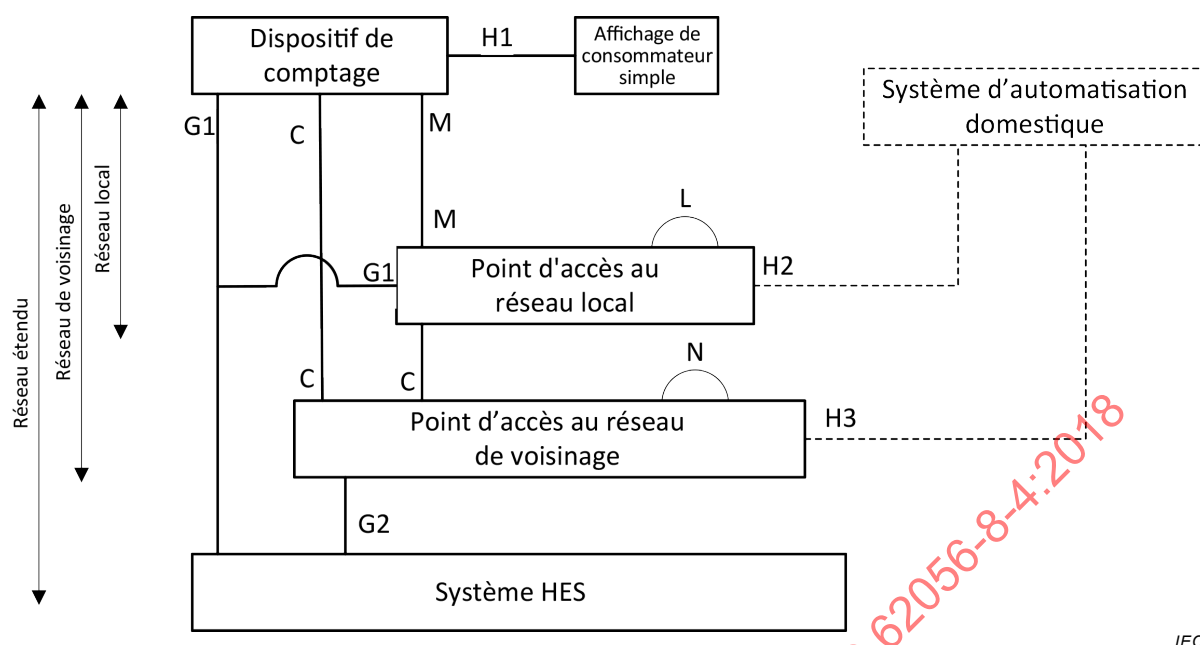


**Figure 1 – Architecture de communication**

Les dispositifs terminaux (en général les compteurs d'électricité) comportent des fonctions d'application et des fonctions de communication. Ils peuvent être reliés directement au NNAP par l'intermédiaire de l'interface C ou à un LNAP par l'intermédiaire d'une interface M, le LNAP étant relié au NNAP par l'intermédiaire de l'interface C. La fonction LNAP peut cohabiter avec les fonctions de comptage.

Un NNAP est composé de fonctions de passerelle et peut comporter des fonctions de concentrateur. En amont, il est relié à l'HES (Metering Head End System – système centralisé de gestion et de télérelevé) à l'aide de supports et protocoles de communication adaptés.

Les dispositifs terminaux et les LNAP peuvent communiquer avec différents NNAP, mais avec un seul NNAP à la fois. Du point de vue de la communication par PLC, le NNAP joue le rôle de nœud de base, tandis que les dispositifs terminaux et les LNAP jouent le rôle de nœuds de service.

Les NNAP, et de façon similaire les LNAP, peuvent communiquer entre eux, mais cela ne fait pas partie du domaine d'application du présent document, qui ne couvre que l'interface C.

Si le NNAP possède des fonctions de concentrateur, il agit comme un client DLMS/COSEM. Si le NNAP possède uniquement des fonctions de passerelle, le HES joue le rôle d'un client DLMS/COSEM. Les dispositifs terminaux ou les LNAP agissent comme des serveurs DLMS/COSEM.

Une architecture mixte est également possible, c'est-à-dire que le HES et le NNAP peuvent agir comme un client.

## 5 Utilisation des couches de communication pour ce profil

### 5.1 Informations relatives à l'utilisation de la norme spécifiant les couches inférieures

La Recommandation UIT-T G.9904:2012 définit les couches PHY et MAC pour la communication par courant porteur utilisant OFDM. En haut de la couche MAC, elle définit également un certain nombre de couches de convergence, dont l'objet est de mettre en

correspondance les couches de protocole inférieures et les couches de protocole supérieures.

Le présent document utilise l'ensemble des couches PHY et MAC. De plus, il définit (en 5.5) l'utilisation des couches de convergence respectives.

## 5.2 Structure des profils de communication

### 5.2.1 Présentation

Les piles de protocoles proposées utilisent les couches OSI suivantes (voir la Figure 2).

- la couche application DLMS/COSEM spécifiée dans l'IEC 62056-5-3 couvrant les fonctionnalités Application, Présentation et Session;
- la sous-couche LLC spécifiée dans l'IEC 61334-4-32, utilisée avec le profil DLMS/COSEM 61334-4-32 sur les réseaux PRIME;
- la couche transport DLMS/COSEM pour les réseaux IP spécifiée dans l'IEC 62056-4-7:2015, utilisée avec les profils DLMS/COSEM TCP-UDP/IPv4 et TCP-UDP/IPv6 sur les réseaux PRIME;
- la couche PRIME MAC, la CPCS et la SSCS correspondante, selon le profil choisi (IEC 61334-4-32, TCP-UDP/IPv4 ou TCP-UDP/IPv6);
- la couche physique PRIME.

Dans le cadre de ce modèle de référence, trois profils différents peuvent être identifiés, tous utilisant les couches PRIME PHY et MAC comme couches inférieures et CPCS d'une part, et la couche application DLMS/COSEM spécifiée dans l'IEC 62056-5-3 et le modèle d'objet COSEM spécifié dans l'IEC 62056-6-1 et l'IEC 62056-6-2 d'autre part. Les couches inférieures (autrement dit PHY et MAC) s'appuient sur les principes de l'IEEE 802.15.4.

NOTE Les classes d'interface COSEM pour la configuration et la gestion de l'échange de données sur le réseau OFDM PLC PRIME à bande étroite sont spécifiées dans l'IEC 62056-6-2.

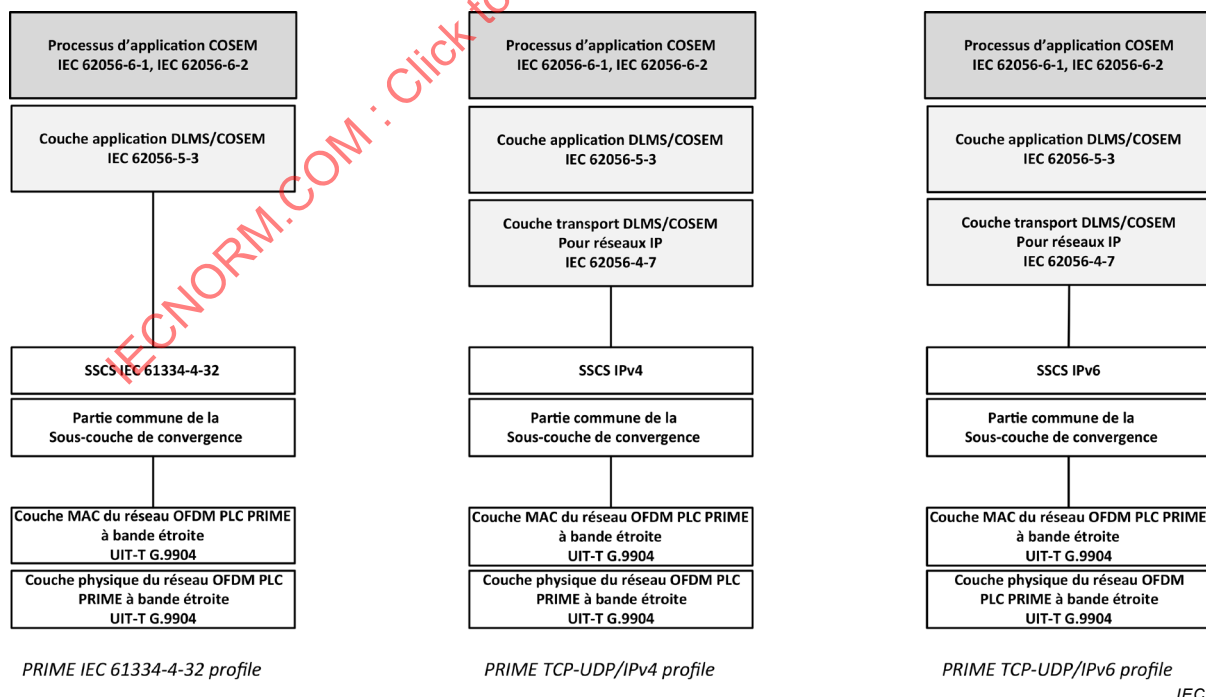


Figure 2 – Architectures du profil de communication OFDM PLC PRIME

### 5.2.2 Profil IEC 61334-4-32

Le profil IEC 61334-4-32 utilise la SSCS PRIME IEC 61334-4-32 assurant l'adaptation nécessaire entre les couches PRIME PHY et MAC et la couche application DLMS/COSEM.

### 5.2.3 Profil TCP-UDP/IPv4

Le profil TCP-UDP/IPv4 utilise la SSCS PRIME IPv4 assurant l'adaptation nécessaire entre les couches PRIME PHY et MAC et la couche IPv4 et prenant en charge la couche transport et la couche application DLMS/COSEM.

### 5.2.4 Profil TCP-UDP/IPv6

Le profil TCP-UDP/IPv6 utilise la SSCS PRIME IPv6 assurant l'adaptation nécessaire entre les couches PRIME PHY et MAC et la couche IPv6 et prenant en charge la couche transport et la couche application DLMS/COSEM.

## 5.3 Couches de protocole inférieures et utilisation

### 5.3.1 Généralités

Les trois profils spécifiés dans le présent document partagent les mêmes couches PHY et MAC.

### 5.3.2 Couche physique

#### 5.3.2.1 Généralités

Cette couche assure l'interface entre le matériel et le support de transmission physique (le réseau de distribution d'électricité). Elle transmet et reçoit des MPDU entre des nœuds voisins.

#### 5.3.2.2 Services de plan de données PRIME PHY

Les services PHY DATA sont générés/utilisés par l'entité de couche MAC à chaque fois que des données (des PPDU) doivent être transmises à une ou plusieurs entités MAC homologues ou reçues de celles-ci à l'aide des procédures de transmission PHY. Voir la Recommandation UIT-T G.9904:2012, 7.10.2.

#### 5.3.2.3 Services de plan de contrôle PRIME PHY

Les services de plan de contrôle PRIME PHY sont utilisés pour permettre à la couche MAC de contrôler la couche physique. Voir la Recommandation UIT-T G.9904:2012, 7.10.3. Ces concepts sont les suivants:

- PHY\_AGC: permet à l'entité de couche MAC de définir ou d'extraire le mode de gain automatique (Automatic Gain Mode) de la couche PHY;
- PHY\_TIMER: permet à l'entité de couche MAC d'extraire l'heure à laquelle la transmission doit commencer;
- PHY\_CD: permet à l'entité de couche MAC de rechercher le signal de détection de porteuse, afin de déterminer si le support physique est disponible;
- PHY\_NL: permet à l'entité de couche MAC d'extraire la valeur plancher du niveau de bruit de fond sur la ligne d'énergie;
- PHY\_SNR: permet à l'entité de couche MAC d'extraire la valeur du rapport signal sur bruit, afin de déterminer le degré de robustesse approprié nécessaire à l'échange des données;
- PHY\_ZCT: permet à l'entité de couche MAC d'extraire l'heure de passage à zéro du réseau et la durée entre la dernière transmission ou réception et le passage à zéro du réseau.

#### 5.3.2.4 Services de plan de gestion PRIME PHY

Les services de plan de gestion PRIME PHY sont utilisés pour permettre à la couche MAC de gérer la couche physique. Voir la Recommandation UIT-T G.9904:2012, 7.10.4. Ces concepts sont les suivants:

- **PLME\_RESET:** permet à l'entité de couche MAC de demander à la couche PHY de rétablir son état fonctionnel présent. Par suite de cette primitive, il convient que la couche PHY réinitialise tous les états internes et efface de toutes les mémoires tampons les données reçues ou transmises placées dans la file d'attente;
- **PLME\_SLEEP:** permet à l'entité de couche MAC de demander à la couche PHY de suspendre ses activités en cours, y compris toutes les fonctions de réception. Il convient que la couche PHY mette fin à toutes les transmissions en attente avant de passer à l'état de veille;
- **PLME\_RESUME:** permet à l'entité de couche MAC de demander à la couche PHY de reprendre ses activités suspendues. Par suite de cette primitive, il convient que la couche PHY lance ses fonctions normales de transmission et de réception;
- **PLME\_TESTMODE:** permet à l'entité de couche MAC de placer la couche PHY dans des modes fonctionnels sans défaut. Le mode fonctionnel spécifique parmi les différents modes possibles est fourni en tant que paramètre d'entrée. Après la réception de cette primitive, il convient que la couche PHY mette fin à toutes les transmissions en attente dans sa mémoire tampon avant d'entrer dans le mode d'essai demandé;
- **PLME\_GET:** permet à l'entité de couche MAC de demander des informations relatives à un attribut donné de la base d'informations PRIME.

#### 5.3.3 Couche MAC

##### 5.3.3.1 Présentation – principales fonctionnalités et fonctions

Un sous-réseau peut être considéré comme une structure arborescente contenant deux types de nœuds: le nœud de base et les nœuds de service. Le nœud de base est à la racine de l'arborescence et fait office de nœud maître assurant la connectivité du réseau. Il n'y a qu'un seul et unique nœud de base dans un sous-réseau. Tous les autres nœuds de sous-réseau sont des nœuds de service. Les nœuds de service sont des feuilles ou des points de branchement (commutateurs) de la structure arborescente.

Au départ, le nœud de base est le sous-réseau lui-même, et il convient que tous les autres nœuds suivent un processus d'enregistrement pour intégrer le sous-réseau.

Les nœuds de service commencent à l'état "Déconnecté" et tentent de trouver un nœud de base ou un commutateur pour s'enregistrer auprès du sous-réseau. Ensuite, ils deviennent des feuilles de l'arborescence.

L'état des nœuds de service peut passer de manière dynamique des fonctions de "Terminal" aux fonctions de "Commutateur", et inversement. Ces passages se produisent en fonction de certains événements prédéfinis sur le réseau. Les nœuds de service à l'état "Commutateur" deviennent des points de branchement de l'arborescence, capables de commuter les données de leurs voisins pour propager la connectivité.

Les trois états fonctionnels des nœuds de service sont les suivants:

- **Déconnecté:** tous les nœuds sont dans cet état au départ ou après un redémarrage. Dans cet état, un nœud n'est pas en mesure de participer au réseau. Un nœud de service dans cet état a pour principale fonction de rechercher un réseau opérationnel à proximité et de tenter de s'enregistrer auprès de lui;
- **Terminal:** dans cet état, un nœud de service fait partie intégrante du réseau. Il est en mesure de communiquer son trafic en établissant des connexions, mais pas de commuter le trafic d'un autre nœud; et enfin:

- **Commutateur:** dans cet état, un nœud de service est en mesure d'exécuter toutes les fonctions de Terminal. De plus, il est en mesure de transférer les données entre les dispositifs présents dans le sous-réseau.

Les événements et les processus associés qui déclenchent le passage d'un état fonctionnel à l'autre sont l'enregistrement, l'annulation d'enregistrement, la promotion et la rétrogradation.

Les autres fonctions de la couche MAC sont les suivantes:

- résolution et diffusion d'adresse, et l'adressage multicast;
- mise en œuvre de l'algorithme CSMA/CA;
- promotion des nœuds de service de l'état "Terminal" à l'état "Commutateur" ou rétrogradation de l'état "Commutateur" à l'état "Terminal";
- établissement de connexions directes d'un nœud de service à un autre;
- agrégation de paquets;
- fonctions de sécurité (gestion des clés de chiffrement et de sécurité, par exemple);
- gestion de la robustesse de la couche PHY afin de choisir le meilleur schéma de modulation pour une situation donnée;
- mécanisme ARQ.

#### 5.3.3.2 Services utilisés par le nœud de base et les nœuds de service

La connexion MAC et les services de données utilisés par les nœuds de base et les nœuds de service sont les suivants (voir la Recommandation UIT-T G.9904:2012, 8.5.2 et 8.5.4):

- MAC\_ESTABLISH: utilisé pour gérer l'établissement de connexion au niveau de la couche MAC;
- MAC\_RELEASE: utilisé pour libérer une connexion au niveau de la couche MAC;
- MAC\_JOIN: utilisé pour rejoindre une connexion de diffusion ou une connexion multicast et permettre la réception des paquets;
- MAC\_LEAVE: utilisé pour quitter une connexion de diffusion ou une connexion multicast;
- MAC\_DATA: utilisé pour envoyer et recevoir des données unicast, multicast ou de diffusion.

#### 5.3.3.3 Services de gestion

Les services de gestion MAC (voir la Recommandation UIT-T G.9904:2012, 8.5.5) sont les suivants:

- MLME\_REGISTER: utilisé pour les enregistrements et pour annoncer quand ils ont été réalisés;
- MLME\_UNREGISTER: utilisé pour annuler les enregistrements et pour annoncer quand a eu lieu l'annulation;
- MLME\_PROMOTE: utilisé pour promouvoir les nœuds de service de l'état "Terminal" à l'état "Commutateur" et pour annoncer quand a eu lieu la promotion;
- MLME\_DEMOTE: utilisé pour rétrograder les nœuds de service de l'état "Commutateur" à l'état "Terminal" et pour annoncer quand a eu lieu la rétrogradation;
- MLME\_RESET: utilisé pour réinitialiser la couche MAC à un état "correct" connu. Après la réinitialisation, la couche MAC est à l'état fonctionnel déconnecté;
- MLME\_GET: utilisé pour extraire des valeurs individuelles de la couche MAC (des statistiques, par exemple);
- MLME\_LIST\_GET: utilisé pour extraire une liste de valeurs de la couche MAC;
- MLME\_SET: utilisé pour définir les valeurs de configuration dans la couche MAC.

## 5.4 Mise en correspondance de services et couches de convergence

### 5.4.1 Présentation

NOTE 1 La description suivante s'appuie sur la Recommandation UIT-T G.9904:2012, 9.1.

La couche de convergence est divisée en deux sous-couches:

La partie commune de la Sous-couche de convergence (CPCS) offre un ensemble de services génériques. L'utilisation des services CPCS est facultative en ce sens qu'une SSCS configure dans sa pile de protocoles les services exigés à partir de la CPCS et ignore les services qui ne sont pas exigés. Dans la présente spécification, la CPCS offre les services suivants aux différentes SSCS: segmentation et réassemblage.

La sous-couche de convergence spécifique au service (SSCS) contient des services spécifiques à un profil de communication donné. Quatre SSCS sont spécifiées pour connecter la couche MAC à la couche supérieure:

- la SSCS IEC 61334-4-32, voir 5.4.2;
- la SSCS IPv4, voir 5.4.3;
- la SSCS IPv6, voir 5.4.4;
- la sous-couche de convergence nulle.

NOTE 2 La sous-couche de convergence nulle ne relève pas du domaine d'application du présent document. Les informations peuvent être consultées à l'Article 9 de la Recommandation UIT-T G.9904:2012.

### 5.4.2 SSCS IEC 61334-4-32 LLC

#### 5.4.2.1 Généralités

La sous-couche de convergence IEC 61334-4-32 LLC (SSCS) offre des fonctions de convergence pour le profil de communication utilisant les services IEC 61334-4-32 LLC. Les mises en œuvre conformes à cette SSCS doivent offrir tous les services LLC spécifiés dans le présent document. De plus, la SSCS PRIME IEC 61334-4-32 doit offrir les services qui aident à la mise en correspondance du protocole LLC sans connexion avec la nature orientée connexion de PRIME MAC, ce qui signifie qu'avant de pouvoir échanger des données, une connexion doit être établie. Voir la Recommandation UIT-T G.9904:2012, 9.5.2.

Caractéristiques principales:

- un nœud de service peut uniquement échanger des données avec le nœud de base et pas avec d'autres nœuds de service;
- chaque session IEC 61334-4-32 SSCS établit une connexion PRIME MAC dédiée pour l'échange de données unicast avec le nœud de base;
- la session SSCS du nœud de service est chargée d'initier cette connexion au nœud de base. La SSCS du nœud de base ne peut pas initier de connexion au nœud de service;
- chaque SSCS IEC 61334-4-32 écoute une connexion MAC de diffusion PRIME dédiée au transfert de données de diffusion IEC 61334-4-32 entre le nœud de base et les nœuds de service. Cette connexion de diffusion est utilisée lorsque le processus d'application du nœud de base utilisant le profil de communication IEC 61334-4-32 formule une demande de transmission avec l'adresse de destination utilisée pour la diffusion ou lorsque les fonctions SAP de diffusion sont utilisées. Si un nœud de service contient plusieurs sessions SSCS, une connexion MAC de diffusion PRIME est partagée par toutes les sessions SSCS.

Les services SSCS IEC 61334-4-32 sont représentés à la Figure 3 et spécifiés en 5.4.2.2.

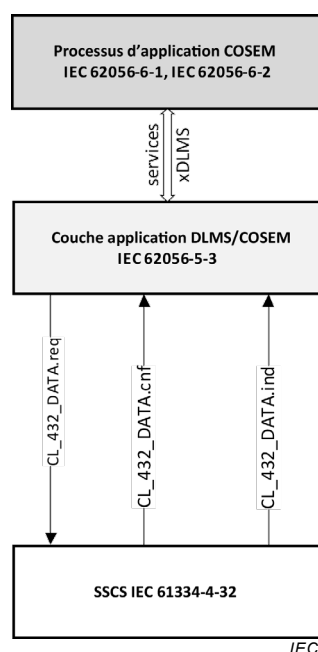


**Figure 3 – Services SSCS IEC 61334-4-32**

#### 5.4.2.2 Services CL\_432\_DATA

##### 5.4.2.2.1 Généralités

L'IEC 61334-4-32 spécifie les services de données de couche LLC suivants:

- Services DL\_Data;
- Services DL\_Broadcast;
- Services DL\_Reply; et
- Services DL\_Update\_Reply.

Dans le profil DLMS/COSEM PRIME IEC 61334-4-32, seuls les services DL\_Data sont utilisés. Ils sont mis en correspondance avec les services CL\_432\_DATA. Tous les autres services sont également mis en correspondance avec les services CL\_432\_DATA (voir ci-dessous).

##### 5.4.2.2.2 Services CL\_432\_DATA

Pour les services DL\_Data, voir l'IEC 61334-4-32. Dans le présent document, les services DL\_Data sont mis en correspondance avec les services CL\_432\_DATA.

Les services xDLMS LN sont présentés en exemple ici. PRIME n'impose aucune limite quant à l'utilisation de l'un des services xDLMS.

La Figure 4 présente un exemple d'utilisation des services CL\_432\_Data.

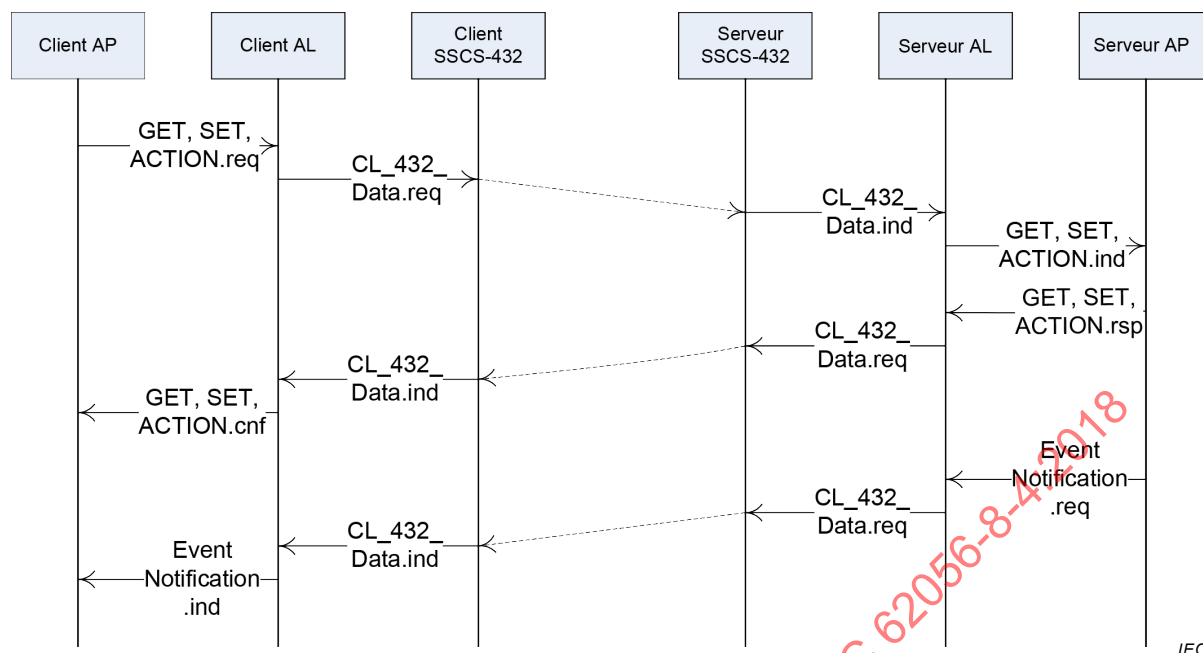


**Figure 4 – MSC pour les services de données dans le cas du référencement de nom logique**

#### 5.4.2.2.3 Services DL\_Broadcast

Le service DL\_Broadcast.request spécifié dans l'IEC 61334-4-32 n'est pas utilisé dans ce profil. La primitive CL\_432\_DATA.request est utilisée à sa place avec une adresse de diffusion comme adresse de destination.

#### 5.4.2.2.4 Services DL\_Reply et DL\_Update\_Reply

Dans l'IEC 61334-4-32, les services DL\_Reply et Update\_Reply sont utilisés pour la gestion de service non sollicité. Dans le profil DLMS/COSEM PRIME IEC 61334-4-32, ces services sont remplacés par les services CL\_432\_DATA au niveau de la SSCS.

### 5.4.3 SSCS IPv4

#### 5.4.3.1 Services de données

##### 5.4.3.1.1 Généralités

Les primitives suivantes sont utilisées pour l'envoi et la réception de paquets IPv4.

##### 5.4.3.1.2 CL\_IPv4\_DATA.request

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.5.2.

##### Fonction

Cette primitive est transmise de la couche IPv4 à la SSCS IPv4. Elle contient un paquet IPv4 à envoyer.

##### Paramètres de service

La sémantique de cette primitive est la suivante:

CL\_IPv4\_DATA.request(IPv4\_PDU)

L'IPv4\_PDU est le paquet IPv4 à envoyer.

#### *Utilisation*

Cette primitive est utilisée lorsqu'un paquet IPv4 doit être transmis à la couche IPv4 homologue.

#### **5.4.3.1.3 CL\_IPv4\_DATA.confirm**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.5.3.

#### *Fonction*

Cette primitive est transmise de la SSCS IPv4 à la couche IPv4. Elle contient une indication d'état et un paquet IPv4 qui vient d'être envoyé.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_DATA.confirm(IPv4\_PDU, Result)

L'IPv4\_PDU est le paquet IPv4 qui a été envoyé.

L'état est décrit dans le champ "Result". La valeur Result indique si le paquet a été envoyé ou si une erreur s'est produite. Sa valeur est celle du Tableau 1.

#### *Utilisation*

Cette primitive est générée en local pour indiquer au demandeur le résultat de la demande de transfert de données.

#### **5.4.3.1.4 CL\_IPv4\_DATA.indicate**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.5.4.

#### *Fonction*

Cette primitive est transmise de la SSCS IPv4 à la couche IPv4. Elle contient un paquet IPv4 qui vient d'être reçu.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_DATA.indicate(IPv4\_PDU)

L'IPv4\_PDU est le paquet IPv4 qui a été reçu.

#### *Utilisation*

Dès l'arrivée d'un nouveau paquet IPv4 de la part de l'homologue, la primitive est générée.

#### **5.4.4 SS CS IPv6**

##### **5.4.4.1 Services de données**

###### **5.4.4.1.1 Généralités**

Les primitives suivantes sont utilisées pour l'envoi et la réception de paquets IPv6.

###### **5.4.4.1.2 CL\_IPv6\_DATA.request**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.6.9.5.2.

###### *Fonction*

Cette primitive est transmise de la couche IPv6 à la SS CS IPv6. Elle contient un paquet IPv6 à envoyer.

###### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_DATA.request(IPv6\_PDU)

L'IPv6\_PDU est le paquet IPv6 à envoyer.

###### *Utilisation*

Cette primitive est utilisée lorsqu'un paquet IPv6 doit être transmis à la couche IPv6 homologue.

###### **5.4.4.1.3 CL\_IPv6\_DATA.confirm**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.6.9.5.3.

###### *Fonction*

Cette primitive est transmise de la SS CS IPv6 à la couche IPv6. Elle contient une indication d'état et un paquet IPv6 qui vient d'être envoyé.

###### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_DATA.confirm(IPv6\_PDU, Result)

L'IPv6\_PDU est le paquet IPv6 qui a été envoyé.

La valeur Result indique si le paquet a été envoyé ou si une erreur s'est produite. Sa valeur est celle du Tableau 1.

###### *Utilisation*

Cette primitive est générée en local pour indiquer au demandeur le résultat de la demande de transfert de données.

###### **5.4.4.1.4 CL\_IPv6\_DATA.indicate**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.6.9.5.4.

### Fonction

Cette primitive est transmise de la SSCS IPv6 à la couche IPv6. Elle contient un paquet IPv6 qui vient d'être reçu.

### Paramètres de service

La sémantique de cette primitive est la suivante:

CL\_IPv6\_DATA.indicate(IPv6\_PDU)

L'IPv6\_PDU est le paquet IPv6 qui a été reçu.

### Utilisation

Dès l'arrivée d'un nouveau paquet IPv6 de la part de l'homologue, la primitive est générée.

## 5.5 Gestion des enregistrements et des connexions

### 5.5.1 Présentation

Le présent paragraphe donne les éléments pour la gestion des enregistrements de dispositif et des connexions. Toutes les SSCS PRIME sont orientées connexion et exigent l'établissement d'une connexion pour pouvoir échanger des données.

### 5.5.2 Profil IEC 61334-4-32

#### 5.5.2.1 Services de gestion des connexions

##### 5.5.2.1.1 Présentation

Les services suivants sont utilisés pour la gestion des connexions SSCS IEC 61334-4-32 (voir la Figure 5).

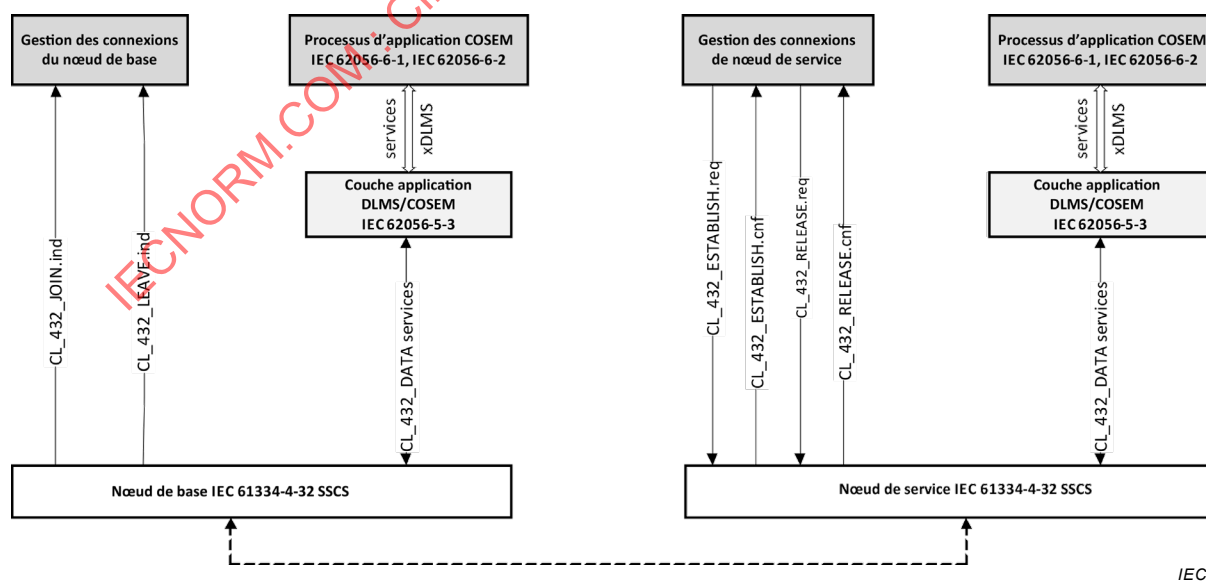


Figure 5 – SSCS IEC 61334-4-32

##### 5.5.2.1.2 CL\_432\_ESTABLISH.request

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.5.6.1.1.

### *Fonction*

Ce service a pour objet de demander une ouverture de connexion avec le nœud de base.

### *Paramètres de service*

La sémantique de la primitive est la suivante:

```
CL_432_ESTABLISH.request
(
    Device_Identifier
)
```

Device\_Identifier est le numéro de série du nœud de service.

### *Utilisation*

Cette primitive de service est utilisée par le nœud de service IEC 61334-4-32 Connection Manager pour demander l'établissement d'une connexion avec le nœud de base.

#### **5.5.2.1.3 CL\_432\_ESTABLISH.confirm**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.5.6.1.2.

### *Fonction*

Ce service a pour objet d'informer le nœud de service IEC 61334-4-32 Connection Manager du résultat de la CL\_432\_ESTABLISH.request précédente.

### *Paramètres de service*

La sémantique de la primitive est la suivante:

```
CL_432_ESTABLISH.confirm
(
    Device_Identifier,
    Destination_Address,
    Base_Address
)
```

Device\_Identifier est le numéro de série du nœud de service. Sa valeur est la même que celle de la primitive CL\_432\_ESTABLISH.req.

Destination\_Address est l'adresse attribuée par le nœud de base au nœud de service concerné lors du processus d'établissement de la connexion. Sa portée est le sous-réseau géré par le nœud de base. Lors du processus d'enregistrement, le nœud de base attribue une adresse de destination de sous-réseau unique aux nœuds de service.

Base\_Address est l'adresse attribuée au nœud de base.

### *Utilisation*

Dès la réception de la primitive de confirmation d'établissement de la connexion, la session SSCS IEC 61334-4-32 confirme au nœud de service IEC 61334-4-32 Connection Manager que la session SSCS IEC 61334-4-32 a été ouverte et indique la Destination\_Address attribuée à la session SSCS IEC 61334-4-32 du nœud de service, ainsi que l'adresse du nœud de base.

Le nœud de base SSCS IEC 61334-4-32 est chargé d'attribuer ces adresses de manière dynamique et d'associer l'identifiant du dispositif de session SSCS IEC 61334-4-32 du nœud de service à la Destination\_Address attribuée (voir l'IEC 61334-4-511).

Le nœud de service ouvre également une connexion de diffusion MAC si aucune autre session SSCS IEC 61334-4-32 n'a été ouverte en tant que telle. Cette connexion permet de recevoir les paquets de diffusion envoyés par le nœud de base IEC 61334-4-32 SSCS à toutes les sessions SSCS IEC 61334-4-32 du nœud de service.

Si CL\_432\_ESTABLISH.request échoue, une primitive CL\_432\_Release.confirm est renvoyée. La raison de l'échec est indiquée dans le résultat.

#### **5.5.2.1.4 CL\_432\_JOIN.indicate**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.5.6.3.2.

##### *Fonction*

Ce service a pour objet d'informer le nœud de base IEC 61334-4-32 Connection Manager de l'établissement d'une nouvelle connexion.

##### *Paramètres de service*

La sémantique du service est la suivante:

```
CL_432_JOIN.indicate
(
    Device_Identifier,
    Destination_Address
)
```

Device\_Identifier est le numéro de série du nœud de service qui a rejoint le sous-réseau PRIME PLC.

Destination\_Address est l'adresse de destination fournie par la SSCS du nœud de base au nœud de service qui a demandé l'établissement de la connexion.

Voir également 6.1.

##### *Utilisation*

La SSCS IEC 61334-4-32 du nœud de base utilise ce service pour informer l'IEC 61334-4-32 Connection Manager que l'adresse de destination concernée est attribuée au nœud de service identifié par Device\_Identifier.

#### **5.5.2.1.5 CL\_432\_RELEASE.request**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.5.6.1.3.

##### *Fonction*

Cette primitive de service est transmise du nœud de service IEC 61334-4-32 Connection Manager à la SSCS IEC 61334-4-32. Elle est utilisée pour fermer la SSCS et libérer toutes les ressources qu'elle peut contenir.

##### *Paramètres de service*

La sémantique du service est la suivante:

```
CL_432_RELEASE.request
(
    Destination_Address
)
```

Destination\_Address est l'adresse de destination fournie par la SSCS du nœud de base au nœud de service lors de l'établissement de la connexion.

#### Utilisation

Cette primitive est utilisée par le nœud de service IEC 61334-4-32 Connection Manager pour demander à la SSCS IEC 61334-4-32 de fermer la connexion.

#### 5.5.2.1.6 CL\_432\_RELEASE.confirm

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.5.6.1.4.

#### Fonction

Cette primitive de service est transmise par la SSCS IEC 61334-4-32 du nœud de service à l'IEC 61334-4-32 Connection Manager pour l'informer que la SSCS a été fermée. Cela peut être dû à une précédente CL\_432\_RELEASE.request ou à une erreur, ce qui a forcé la fermeture de la SSCS.

#### Paramètres de service

La sémantique de cette primitive est la suivante:

```
CL_432_RELEASE.confirm
(
    Destination_Address,
    Result
)
```

Destination\_Address identifie la connexion qui a été fermée. Les valeurs possibles du paramètre Result sont indiquées dans le Tableau 1.

#### Utilisation

Cette primitive est utilisée par la SSCS IEC 61334-4-32 du nœud de service pour informer l'IEC 61334-4-32 Connection Manager du résultat de la demande de libération.

**Tableau 1 – Valeurs du paramètre Result pour les services SSCS**

| Result             | Description                                                             |
|--------------------|-------------------------------------------------------------------------|
| Succès = 0         | Le service SSCS a été exécuté avec succès.                              |
| Rejet = 1          | Le service SSCS n'a pas abouti car il a été rejeté par le nœud de base. |
| Expiration = 2     | Le traitement du service SSCS a expiré.                                 |
| Non enregistré = 6 | Le nœud de service n'est pas encore enregistré dans un sous-réseau.     |

#### 5.5.2.1.7 CL\_432\_LEAVE.indicate

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.5.6.3.3.

#### Fonction

Cette primitive de service est utilisée par la SSCS du nœud de base pour informer l'IEC 61334-4-32 Connection Manager que le nœud de service identifié par la Destination\_Address a quitté le sous-réseau.

#### Paramètres de service

La sémantique de cette primitive est la suivante:

```
CL_432_LEAVE.indicate
(
    Destination_Address
)
```

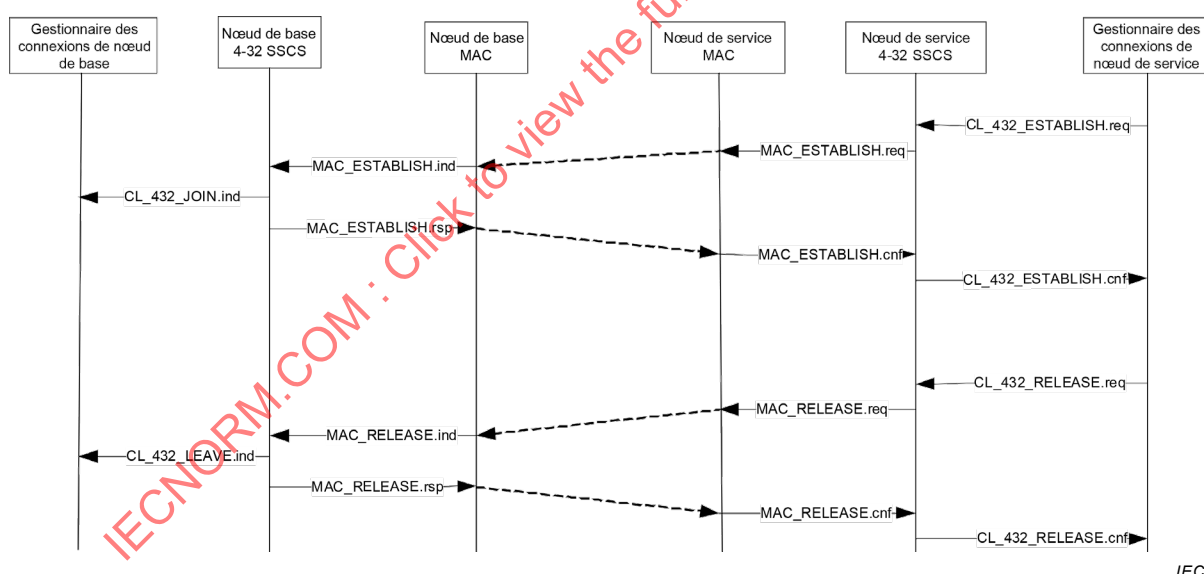
Destination\_Address est l'adresse du nœud de service qui quitte le sous-réseau.

#### Utilisation

La SSCS IEC 61334-4-32 du nœud de base utilise ce service pour informer l'IEC 61334-4-32 Connection Manager que le nœud de service identifié par la Destination\_Address a quitté le sous-réseau.

### 5.5.2.2 Récapitulatif des services de gestion des connexions

Le MSC des primitives de service échangées entre l'IEC 61334-4-32 Connection Manager et la SSCS IEC 61334-4-32 est représenté à la Figure 6. Des cas d'erreur sont présentés à l'Annexe C.



IEC

Figure 6 – MSC pour les services SSCS IEC 61334-4-32

### 5.5.3 Profil TCP-UDP/IPv4

#### 5.5.3.1 Présentation

##### 5.5.3.1.1 Architecture générale

Le profil de communication basé sur TCP-UDP/IPv4 s'inscrit totalement dans le profil de communication DLMS/COSEM pour TCP-UDP/IP, tel que spécifié dans l'IEC 62056-9-7:2013. Se référer à cette norme pour plus d'informations. Le présent paragraphe donne des informations relatives à la liaison des couches TCP-UDP/IPv4 à la SSCS IPv4 uniquement.

La SSCS IPv4 fournit une méthode efficace de transport des paquets IPv4 sur le réseau PRIME.

Même si IPv4 est un protocole sans connexion, la SSCS IPv4 est orientée connexion, ce qui signifie qu'une connexion est établie entre les nœuds de service source et de destination pour le transfert des paquets IP. Cette connexion est maintenue lors du transfert du trafic et peut être interrompue après une période d'inactivité. La SSCS présente deux types de connexions:

- pour la résolution d'adresse: il existe une connexion au nœud de base pour la résolution d'adresse;
- pour le transfert de données IPv4, il existe une connexion par nœud de destination. La destination peut être le nœud de base, un autre nœud de service du même sous-réseau ou une destination à l'extérieur du sous-réseau.

Les en-têtes TCP/IPv4 peuvent éventuellement être compressés. La compression est négociée dans le cadre de la phase d'établissement de la connexion.

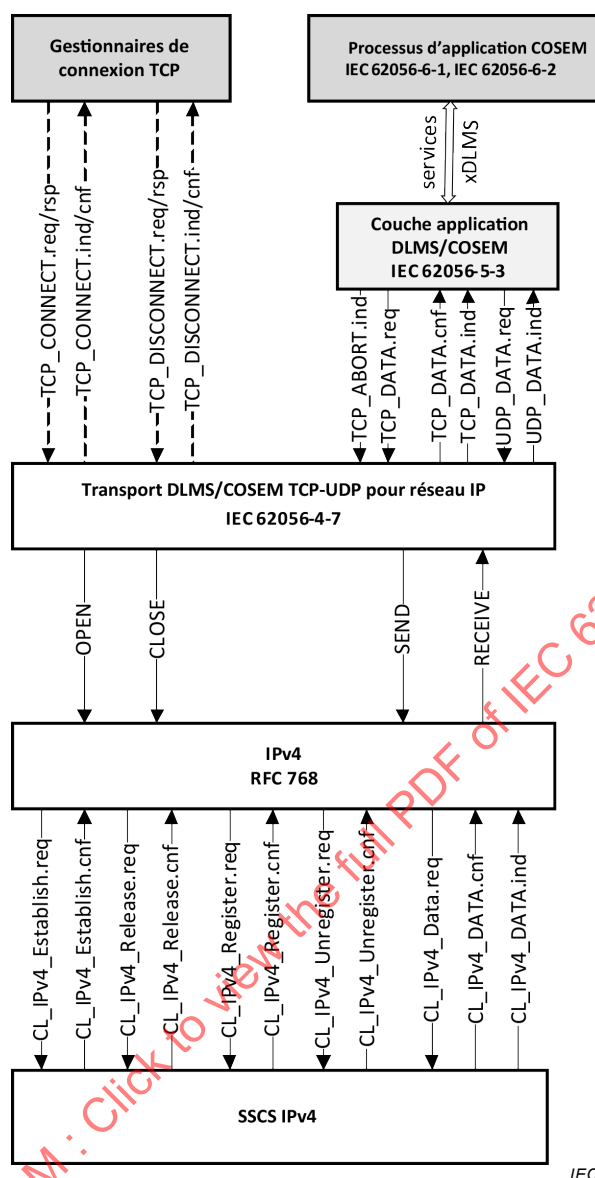
La diffusion des paquets IPv4 est prise en charge à l'aide du mécanisme de diffusion MAC.

La multidiffusion des paquets IPv4 est prise en charge à l'aide du mécanisme de multidiffusion MAC.

Les services de segmentation et de réassemblage sont assurés par la CPCS.

Pour plus d'informations, voir la Recommandation UIT-T G.9904:2012.

L'architecture générale est représentée à la Figure 7.



IEC

NOTE Des détails relatifs au nœud qui émet les primitives sont donnés dans la description du service. Voir 5.5.3.2 et 5.5.3.5.3.

**Figure 7 – Service SSCS IPv4**

Dans cette figure, seules les couches supérieures sont présentées. Les couches inférieures et leurs relations sont décrites dans la Recommandation UIT-T G.9904:2012.

#### 5.5.3.1.2 Gestionnaire de connexions TCP

Le gestionnaire de connexions TCP est spécifié dans l'IEC 62056-9-7:2013.

#### 5.5.3.1.3 TCP-UDP/IPv4

Les couches IPv4, TCP et UDP mettent respectivement en œuvre les services IPv4, TCP et UDP normalisés spécifiés dans STD0005, STD0006 et STD0007. La manière dont la couche IPv4 est liée à la SSCS IPv4 SSCS fait l'objet des paragraphes suivants.

#### 5.5.3.1.4 Passerelle de sous-réseau

Le nœud de base agit en général comme une passerelle vers le sous-réseau PRIME assurant la connectivité IP aux nœuds de service. Ses principales fonctions sont les suivantes:

- elle peut échanger des paquets IPv4 avec les nœuds de service et peut également transférer les paquets d'un nœud de service à un autre;
- elle peut exécuter les fonctions NAT;
- elle peut exécuter les fonctions DHCP;

Il est noté que les adresses IPv4 peuvent être préalablement configurées de manière statique ou dynamique. Si l'adresse IPv4 est configurée de manière dynamique, un serveur DHCP est utilisé. Si DHCP est utilisé, il est exécuté à l'aide des services de diffusion/multidiffusion. Ensuite, l'adresse IPv4 est enregistrée à l'aide du service CL\_IPv4\_REGISTER. La description du processus DHCP ou NAT ne relève pas du domaine d'application du présent document.

- elle procède à la résolution d'adresse IPv4 en adresse EUI-48, en maintenant une base de données des adresses EUI-48 et IPv4 de chaque nœud de service enregistré. Les nœuds de service (à condition qu'ils soient enregistrés avec le nœud de base) peuvent alors interroger le nœud de base pour résoudre une adresse IPv4 en adresse EUI-48;
- elle vérifie dans chaque paquet l'adresse IPv4 afin de déterminer s'il convient que le paquet soit traité par un nœud ou envoyé à une destination à l'extérieur du sous-réseau.

Afin de conserver la simplicité de la mise en œuvre, une seule route est prise en charge par adresse IPv4 locale.

### 5.5.3.2 Ouverture et fermeture de la SSCS IPv4

#### 5.5.3.2.1 Généralités

Les services suivants sont utilisés pour ouvrir et fermer la SSCS IPv4. À un moment donné, il ne peut exister qu'une seule connexion SSCS, la SSCS ne doit donc être ouverte qu'une seule fois. La couche IPv4 peut fermer la SSCS lorsque l'interface IPv4 est réduite. La SSCS se ferme également lorsque la connexion MAC sous-jacente au nœud de base est perdue.

Pour le MSC des services SSCS IPv4, voir l'Annexe A.

#### 5.5.3.2.2 CL\_IPv4\_ESTABLISH.request

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.2.2.

##### *Fonction*

Cette primitive de service est émise par la couche IPv4 du nœud de service à la SSCS IPv4. Cela est assuré à chaque fois que la couche IPv4 relance l'interface.

##### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_ESTABLISH.request()

##### *Utilisation*

Dès la réception de cette primitive de service, la SSCS IPv4 du nœud de service forme la connexion de résolution d'adresse pour le nœud de base et rejoint le groupe de diffusion utilisé pour la réception/transmission des paquets de diffusion. Voir Figure A.1.

### 5.5.3.2.3 CL\_IPv4\_ESTABLISH.confirm

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.2.3.

#### *Fonction*

Cette primitive de service est transmise de la SSCS IPv4 du nœud de service à la couche IPv4. En cas de succès, elle signale que la SSCS IPv4 est prête à accepter les paquets IPv4 à envoyer aux homologues.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_ESTABLISH.confirm()

#### *Utilisation*

Cette primitive de service est utilisée pour fournir le résultat du service CL\_IPv4\_ESTABLISH.request. Dès que la SSCS IPv4 a établi toutes les connexions nécessaires et est prête à transmettre et recevoir des paquets IPv4, cette primitive est transmise à la couche IPv4.

Si la SSCS IPv4 rencontre une erreur au moment de l'ouverture, elle répond par une primitive CL\_IPv4\_RELEASE.confirm au lieu d'une primitive CL\_IPv4\_ESTABLISH.confirm.

### 5.5.3.2.4 CL\_IPv4\_RELEASE.request

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.2.4.

#### *Fonction*

Cette primitive de service est utilisée par la couche IPv4 du nœud de service à la SSCS IPv4 pour demander la fermeture de la SSCS IPv4.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_RELEASE.request()

#### *Utilisation*

Cette primitive est utilisée par le nœud de service lorsque la SSCS doit être fermée pour une raison quelconque. La SSCS IPv4 ferme toutes les connexions. Toutes les ressources sont libérées et plus aucun autre paquet IPv4 ne peut être reçu ou envoyé. Lorsque la SSCS IPv4 a libéré toutes ses connexions et ressources, elle renvoie une primitive de service CL\_IPv4\_RELEASE.confirm.

### 5.5.3.2.5 CL\_IPv4\_RELEASE.confirm

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.2.5.

#### *Fonction*

Cette primitive de service est utilisée par la SSCS IPv4 du nœud de service pour indiquer à la couche IPv4 que la SSCS IPv4 a été fermée. Cela peut être le résultat d'une primitive CL\_IPv4\_RELEASE.request ou d'une primitive CL\_IPv4\_ESTABLISH.request qui n'a pas

abouti, ou s'expliquer par le fait qu'une couche MAC indique que la connexion de résolution d'adresse a été perdue (c'est-à-dire que cela peut se produire après une erreur de communication, comme des messages corrompus, trop de retransmissions ARQ, etc.) ou par le fait que le nœud de service lui-même n'est plus enregistré, ce qui signifie que la connectivité IP a disparu.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_RELEASE.confirm(Result)

Les valeurs possibles du paramètre Result sont indiquées dans le Tableau 1.

#### *Utilisation*

La SSCS IPv4 utilise ce service pour informer la couche IPv4 que le nœud de service a quitté le sous-réseau au niveau IP.

### **5.5.3.3 Gestion d'adresses monodiffusion**

#### **5.5.3.3.1 Généralités**

Les services définis ici sont utilisés pour la gestion d'adresses monodiffusion, c'est-à-dire pour l'enregistrement et l'annulation d'enregistrement des adresses IPv4 monodiffusion avec la SSCS IPv4.

Si aucune adresse IPv4 unicast n'est enregistrée avec la SSCS IPv4, cette dernière peut seulement envoyer et recevoir des paquets de diffusion et de multidiffusion. Toutefois, cela est suffisant pour que l'opération BOOTP/DHCP permette au dispositif d'obtenir une adresse IPv4. Lorsqu'une adresse IPv4 a été enregistrée, la couche IPv4 peut transmettre les paquets unicast dont l'adresse source correspond à l'une de ses adresses IPv4 enregistrées.

#### **5.5.3.3.2 CL\_IPv4\_REGISTER.request**

NOTE 1 Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.3.2.

#### *Fonction*

Cette primitive est transmise par la couche IPv4 du nœud de service à la SSCS IPv4 pour demander l'enregistrement d'une adresse IPv4.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_REGISTER.req(  
IPv4\_Address,  
Netmask,  
Gateway)

Où:

- IPv4\_Address est l'adresse à enregistrer;

NOTE 2 Un nœud de service peut enregistrer plusieurs adresses IPv4 monodiffusion.

- Netmask est le masque de réseau. Il est utilisé par la SSCS IPv4 pour déterminer s'il convient de livrer le paquet directement ou s'il convient d'utiliser la passerelle;

- Gateway est une adresse IPv4 de la passerelle à utiliser si l'adresse de destination est à l'extérieur du sous-réseau.

Voir également 5.5.3.1.4.

Chaque sous-réseau PRIME ne disposant que d'un seul nœud de base, l'adresse de passerelle est en général l'adresse IPv4 du nœud de base, mais ce n'est pas obligatoire.

NOTE 3 L'adresse IPv4, le masque de réseau et l'adresse de passerelle sont fournis par la couche IP et sont configurés à l'intérieur du dispositif à l'aide d'un objet "IPv4 setup". Voir IEC 62056-6-2.

#### *Utilisation*

À partir du côté du nœud de service, cet enregistrement est remonté au nœud de base.

Si le nœud de service a correctement enregistré son adresse IPv4 avec le nœud de base, une primitive CL\_IPv4\_REGISTER.cnf est utilisée. Si l'enregistrement n'a pas abouti, la primitive CL\_IPv4\_RELEASE.cnf est utilisée.

Côté nœud de base, l'enregistrement n'est que local et est utilisé pour indiquer à la SSCS IPv4 les paquets qui doivent être transmis à la couche supérieure.

#### **5.5.3.3.3 CL\_IPv4\_REGISTER.confirm**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.3.3.

#### *Fonction*

Cette primitive de service est transmise par la SSCS IPv4 du nœud de service à la couche IPv4 pour indiquer que l'enregistrement de l'adresse IPv4 a abouti.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_REGISTER.cnf(IPv4\_Address)

IPv4\_Address est l'adresse dont l'enregistrement a abouti.

#### *Utilisation*

La primitive de service CL\_IPv4\_REGISTER.cnf est utilisée par la SSCS IPv4 du nœud de service à l'issue de l'enregistrement. La couche IPv4 peut alors utiliser cette adresse pour envoyer des paquets IPv4.

#### **5.5.3.3.4 CL\_IPv4\_UNREGISTER.request**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.3.4.

#### *Fonction*

Cette primitive de service est transmise par la couche IPv4 du nœud de service à la SSCS IPv4 pour demander l'annulation d'enregistrement d'une adresse IPv4.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_UNREGISTER.req(IPv4\_Address)

IPv4\_Address est l'adresse dont l'enregistrement est à annuler.

#### *Utilisation*

Dès que l'enregistrement de l'adresse IPv4 a été annulé avec le nœud de base, une primitive CL\_IPv4\_UNREGISTER.confirm est utilisée. Si CL\_IPv4\_UNREGISTER.request échoue, une primitive CL\_IPv4\_RELEASE.confirm est renvoyée. La raison de l'échec est indiquée dans le résultat.

Plusieurs adresses IPv4 pouvant être enregistrées par le nœud de service, seul l'enregistrement de l'adresse concernée est annulé par cette requête. Les adresses dont l'enregistrement n'a pas été annulé sont toujours valables, jusqu'à ce qu'elles le soient ou que la SSCS soit fermée.

#### **5.5.3.3.5 CL\_IPv4\_UNREGISTER.confirm**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012 9.4.8.3.5.

#### *Fonction*

Cette primitive de service est transmise par la SSCS IPv4 du nœud de service à la couche IPv4 pour indiquer que l'annulation de l'enregistrement de l'adresse IPv4 a abouti.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante.

CL\_IPv4\_UNREGISTER.confirm(IPv4\_Address)

IPv4\_Address est l'adresse IPv4 unicast dont l'enregistrement a été annulé.

#### *Utilisation*

Dès que l'annulation de l'enregistrement a abouti, la couche IPv4 ne peut pas envoyer de paquets IPv4 à l'aide de cette adresse source.

#### **5.5.3.4 Gestion de groupes multicast**

##### **5.5.3.4.1 Généralités**

Le présent paragraphe décrit les primitives de service utilisées pour gérer les groupes multicast.

NOTE Noter que les primitives CL\_432\_JOIN et CL\_IPv4\_IGMP\_JOIN sont de nature différente. Le premier groupe informe le nœud de base des nouveaux nœuds de service qui ont rejoint le réseau. Le deuxième groupe gère l'abonnement à un certain groupe de multicast IP. De même, noter que le nom de cet ensemble de primitives contient l'IGMP, ce qui permet de faire clairement la distinction.

##### **5.5.3.4.2 CL\_IPv4\_IGMP\_Join.req**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.4.2.

#### *Fonction*

Cette primitive de service est transmise de la couche IPv4 du nœud de service ou du nœud de base à la SSCS IPv4. Elle contient l'adresse d'un groupe multicast IPv4 à rejoindre. Le nœud de base et le nœud de service peuvent ensuite utiliser les services de multidiffusion.

### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_IGMP\_JOIN.req(IPv4\_Address)

IPv4\_Address est le groupe multicast IPv4 à rejoindre.

### *Utilisation*

Lorsque la SSCS IPv4 reçoit cette primitive, elle assure que les paquets IPv4 envoyés à ce groupe sont des paquets multicast dans le sous-réseau et de recevoir les paquets à l'aide de cette adresse à transmettre à la couche IPv4. Si la SSCS peut rejoindre le groupe, elle doit utiliser la primitive CL\_IPv4\_IGMP\_JOIN.confirm pour indiquer le succès. Sinon, elle doit utiliser la primitive de service CL\_IPv4\_IGMP\_LEAVE.confirm.

#### **5.5.3.4.3 CL\_IPv4\_IGMP\_JOIN.confirm**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.4.3.

### *Fonction*

Cette primitive de service est transmise de la SSCS IPv4 du nœud de service ou du nœud de base à la couche IPv4.

### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_IGMP\_JOIN.confirm(IPv4\_Address)

IPv4\_Address est le groupe multicast IPv4 qui a été rejoint.

### *Utilisation*

La SSCS commence à transférer les paquets multicast IPv4 pour le groupe multicast donné.

#### **5.5.3.4.4 CL\_IPv4\_IGMP\_LEAVE.request**

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.4.4.

### *Fonction*

Cette primitive de service est transmise de la couche IPv4 du nœud de service ou du nœud de base à la SSCS IPv4. Elle contient l'adresse multicast IPv4 à quitter.

### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_IGMP\_LEAVE.req(IPv4\_Address)

IPv4\_Address est le groupe multicast IPv4 à quitter.

### *Utilisation*

La SSCS arrête de transférer les paquets multicast IPv4 de ce groupe et peut quitter le groupe multicast MAC.

#### 5.5.3.4.5 CL\_IPv4\_IGMP\_LEAVE.confirm

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.4.8.4.5.

##### *Fonction*

Cette primitive de service est transmise de la SSCS IPv4 du nœud de service ou du nœud de base à la couche IPv4 pour indiquer le résultat de CL\_IPv4\_IGMP\_LEAVE.request.

##### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv4\_IGMP\_LEAVE.confirm(IPv4\_address, Result)

IPv4\_Address est l'adresse du groupe multicast IPv4 qui a été quitté. La SSCS IPv4 arrête de transférer les paquets multicast IPv4 pour le groupe multicast donné.

Pour les valeurs de Result, voir le Tableau 1.

##### *Utilisation*

Cette primitive peut être utilisée par la SSCS IPv4 du nœud de service:

- par suite de CL\_IPv4\_IGMP\_LEAVE.request, ou
- par suite d'une primitive CL\_IPv4\_IGMP\_JOIN.request en cas d'échec, ou
- par suite d'une condition d'erreur donnant lieu à la perte de la connexion multicast MAC.

#### 5.5.3.5 PDU IPv4 SSCS

##### 5.5.3.5.1 Généralités

Les paragraphes suivants spécifient les PDU IPv4 SSCS. Les services qui les transportent sont les services de sous-couche MAC présentés à la Figure 7 et à la Figure A.1.

##### 5.5.3.5.2 PDU de résolution d'adresse

###### 5.5.3.5.2.1 Généralités

Les messages suivants sont utilisés sur la connexion de résolution d'adresse entre le nœud de service et le nœud de base. Voir la Recommandation UIT-T G.9904:2012, 9.4.7.

Les valeurs AR.MSG de 6 à 7 sont réservées.

###### 5.5.3.5.2.2 AR\_REGISTER\_S

Le Tableau 2 présente le message d'enregistrement de résolution d'adresse conçu par la SSCS du nœud de service et envoyé par le nœud de service au nœud de base.

**Tableau 2 – Format de message AR\_REGISTER\_S**

| Nom       | Longueur | Description                                                       |
|-----------|----------|-------------------------------------------------------------------|
| AR.MSG    | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_REGISTER_S = 0 |
| AR.IPv4   | 32 bits  | Adresse IPv4 à enregistrer                                        |
| AR.EUI-48 | 48 bits  | Adresse EUI-48 à enregistrer                                      |

**5.5.3.5.2.3 AR\_REGISTER\_B**

Le Tableau 3 présente le message d'acquiescement de résolution d'adresse conçu par la SSCS du nœud de base et envoyé par le nœud de base au nœud de service.

**Tableau 3 – Format de message AR\_REGISTER\_B**

| Nom       | Longueur | Description                                                       |
|-----------|----------|-------------------------------------------------------------------|
| AR.MSG    | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_REGISTER_B = 1 |
| AR.IPv4   | 32 bits  | Adresse IPv4 enregistrée                                          |
| AR.EUI-48 | 48 bits  | Adresse EUI-48 enregistrée                                        |

**5.5.3.5.2.4 AR\_UNREGISTER\_S**

Le Tableau 4 présente le message d'annulation d'enregistrement de résolution d'adresse envoyé par le nœud de service au nœud de base.

**Tableau 4 – Format de message AR\_UNREGISTER\_S**

| Nom       | Longueur | Description                                                                           |
|-----------|----------|---------------------------------------------------------------------------------------|
| AR.MSG    | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_UNREGISTER_S = 2                   |
| AR.IPv4   | 32 bits  | Adresse IPv4 dont l'enregistrement est à annuler                                      |
| AR.EUI-48 | 48 bits  | Adresse EUI-48 du dispositif qui a demandé l'IPv4 dont l'enregistrement est à annuler |

**5.5.3.5.2.5 AR\_MCAST\_REG\_S**

Le Tableau 5 présente le message d'enregistrement de résolution d'adresse multicast envoyé par le nœud de service au nœud de base.

**Tableau 5 – Format de message AR\_MCAST\_REG\_S**

| Nom     | Longueur | Description                                                        |
|---------|----------|--------------------------------------------------------------------|
| AR.MSG  | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_MCAST_REG_S = 8 |
| AR.IPv4 | 32 bits  | Adresse multicast IPv4 à enregistrer                               |

**5.5.3.5.2.6 AR\_MCAST\_REG\_B**

Le Tableau 6 présente le message d'acquiescement d'enregistrement de résolution d'adresse multicast envoyé par le nœud de base au nœud de service.

Le champ AR.IPv4 est inclus dans le message AR\_MCAST\_REG\_B de sorte que le nœud de service puisse procéder à plusieurs enregistrements se chevauchant.

**Tableau 6 – Format de message AR\_MCAST\_REG\_B**

| Nom     | Longueur | Description                                                        |
|---------|----------|--------------------------------------------------------------------|
| AR.MSG  | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_MCAST_REG_B = 9 |
| AR.IPv4 | 32 bits  | Adresse multicast IPv4 enregistrée                                 |
| Réservé | 2 bits   | Réservé. Doit être codé en 0                                       |
| AR.LCID | 6 bits   | LCID attribué à cette adresse multicast IPv4                       |

#### 5.5.3.5.2.7 AR\_MCAST\_UNREG\_S

Le Tableau 7 présente le message d'annulation d'enregistrement de résolution d'adresse multicast envoyé par le nœud de service au nœud de base.

**Tableau 7 – Format de message AR\_MCAST\_UNREG\_S**

| Nom     | Longueur | Description                                                           |
|---------|----------|-----------------------------------------------------------------------|
| AR.MSG  | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_MCAST_UNREG_S = 10 |
| AR.IPv4 | 32 bits  | Adresse multicast IPv4 dont l'enregistrement est à annuler            |

#### 5.5.3.5.2.8 AR\_MCAST\_UNREG\_B

Le Tableau 8 présente le message d'acquiescement d'annulation d'enregistrement de résolution d'adresse multicast envoyé par le nœud de base au nœud de service.

**Tableau 8 – Format de message AR\_MCAST\_UNREG\_B**

| Nom     | Longueur | Description                                                           |
|---------|----------|-----------------------------------------------------------------------|
| AR.MSG  | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_MCAST_UNREG_B = 11 |
| AR.IPv4 | 32 bits  | Adresse multicast IPv4 dont l'enregistrement a été annulé             |

Le champ AR.IPv4 est inclus dans le message AR\_MCAST\_UNREG\_B de sorte que le nœud de service puisse procéder à plusieurs annulations d'enregistrement se chevauchant.

### 5.5.3.5.3 Établissement de connexions de données

#### 5.5.3.5.3.1 Présentation

Les PDU suivantes sont envoyées lors de l'établissement de connexions de données.

#### 5.5.3.5.3.2 AR\_LOOKUP\_S

Le Tableau 9 présente le message de consultation de résolution d'adresse envoyé par le nœud de service au nœud de base.

**Tableau 9 – Format de message AR\_LOOKUP\_S**

| Nom     | Longueur | Description                                                     |
|---------|----------|-----------------------------------------------------------------|
| AR.MSG  | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_LOOKUP_S = 4 |
| AR.IPv4 | 32 bits  | Adresse IPv4 à consulter                                        |

**5.5.3.5.3.3 AR\_LOOKUP\_B**

Le Tableau 10 présente le message de réponse de résolution d'adresse envoyé par le nœud de base au nœud de service.

**Tableau 10 – Format de message AR\_LOOKUP\_B**

| Nom       | Longueur | Description                                                                                                                                                          |
|-----------|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| AR.MSG    | 8 bits   | Type de message de résolution d'adresse<br>Pour AR_LOOKUP_B = 5                                                                                                      |
| AR.IPv4   | 32 bits  | Adresse IPv4 recherchée                                                                                                                                              |
| AR.EUI-48 | 48 bits  | EUI-48 pour l'adresse IPv4                                                                                                                                           |
| AR.Status | 8 bits   | État de la consultation, indiquant si l'adresse a été trouvée ou si une erreur s'est produite.<br>0 = trouvé, AR.EUI-48 valide.<br>1 = inconnu, AR.EUI-48 non défini |

La consultation peut échouer si l'adresse demandée n'a pas été enregistrée. Dans ce cas, la valeur d'AR.Status n'est pas nulle et le contenu d'AR.EUI-48 n'est pas défini. La consultation aboutit uniquement lorsque la valeur d'AR.Status est nulle. Dans ce cas, le champ EUI-48 contient l'adresse résolue.

**5.5.3.6 Format de paquet IPv4 SSCS****5.5.3.6.1 Généralités**

Les formats PDU suivants sont utilisés pour transférer des paquets IPv4 entre des nœuds de service. Deux formats sont définis. Le premier format est utilisé sans compression d'en-tête. Le second format est utilisé avec la compression d'en-tête Van Jacobsen.

**5.5.3.6.2 Format de paquet IPv4 sans compression d'en-tête**

Si aucune compression d'en-tête n'a été négociée, le paquet IP est simplement envoyé en l'état, sans compression d'en-tête (voir le Tableau 11). Pour la négociation de compression, voir 5.5.3.7.2.

**Tableau 11 – Format de paquet IPv4 sans compression d'en-tête négocié**

| Nom     | Longueur | Description |
|---------|----------|-------------|
| IPv4PKT | n octets | Paquet IPv4 |

**5.5.3.6.3 Format de paquet IPv4 avec compression d'en-tête Van Jacobsen**

Le Tableau 12 présente le format de paquet IPv4 lorsque la compression d'en-tête Van Jacobsen a été négociée. Voir la RFC 1144:1990, 3.2.

**Tableau 12 – Format de paquet IPv4 avec compression d'en-tête VJ**

| Nom       | Longueur | Description                                                                                                                                              |
|-----------|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| IPv4.Type | 2 bits   | Type de paquet compressé.<br>IPv4.Type = 0 – TYPE_IP<br>IPv4.Type = 1 – UNCOMPRESSED_TCP<br>IPv4.Type = 2 – COMPRESSED_TCP<br>IPv4.Type = 3 – TYPE_ERROR |
| IPv4.Seq  | 6 bits   | Numéro de séquence de paquet                                                                                                                             |
| IPv4.PKT  | n octets | Paquet IPv4 (avec l'en-tête compressé)                                                                                                                   |

TYPE\_ERROR de la valeur IPv4.Type n'est jamais envoyé. Il s'agit d'un type de pseudo paquet utilisé pour indiquer au programme de décompression qu'un paquet a été perdu.

### 5.5.3.7 Données de connexion

#### 5.5.3.7.1 Généralités

Les PDU suivantes sont échangées lors de la phase d'établissement de connexions de données. Si une connexion est établie entre des nœuds de service pour le transfert de paquets IPv4, les données sont également transférées dans les paquets de demande de connexion. Ces données permettent de négocier la compression et la notification de l'adresse IPv4. Voir la Recommandation UIT-T G.9904:2012, 8.6.6.1.

#### 5.5.3.7.2 Données de connexion provenant de l'initiateur

Le Tableau 13 présente les données de connexion telles qu'elles sont envoyées par l'initiateur de la connexion.

**Tableau 13 – Données de connexion envoyées par l'initiateur**

| Nom       | Longueur | Description                                                                                                                                                                      |
|-----------|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Réservé   | 6 bits   | Il convient de le coder en zéro dans cette version du protocole SSCS                                                                                                             |
| Data.HC   | 2 bits   | Compression d'en-tête<br>Data.HC = 0 – Aucune compression demandée<br>Data.HC = 1 – Compression VJ demandée<br>Data.HC = 2, 3 – Réserve aux futures versions du présent document |
| Data.IPv4 | 32 bits  | Adresse IPv4 de l'initiateur                                                                                                                                                     |

Si le dispositif accepte la connexion, il convient de copier l'adresse Data.IPv4 dans une nouvelle entrée de tableau avec la valeur Data.HC négociée.

#### 5.5.3.7.3 Données de connexion provenant du répondeur

Le Tableau 14 présente les données de connexion envoyées en réponse à la demande de connexion.

**Tableau 14 – Données de connexion envoyées par le répondeur**

| Nom     | Longueur | Description                                                                                                                                             |
|---------|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| Réservé | 6 bits   | Il convient de le coder en zéro dans cette version du protocole SSCS                                                                                    |
| Data.HC | 2 bits   | Compression d'en-tête négociée<br>Data.HC = 0 – Aucune compression d'en-tête admise<br>Data.HC = 1 – Compression VJ négociée<br>Data.HC = 2,3 – Réservé |

Un schéma de compression d'en-tête peut uniquement être utilisé s'il est pris en charge par les deux nœuds de service. Le répondeur peut uniquement attribuer la valeur 0 à Data.HC ou la même valeur que celle reçue de l'initiateur. Si la même valeur est utilisée, elle indique que le schéma de compression demandé a été négocié et qu'il sera utilisé pour la connexion. La valeur 0 attribuée à Data.HC permet au répondeur de refuser la demande pour ce schéma de compression d'en-tête ou de forcer l'utilisation d'aucune compression d'en-tête.

#### 5.5.4 Profil TCP-UDP/IPv6

##### 5.5.4.1 Présentation

###### 5.5.4.1.1 Architecture générale

L'architecture générale TCP-UDP/IPv6 est la même que celle de TCP-UDP/IPv4, conformément à l'IEC 62056-9-7:2013. Les couches inférieures sont conformes à la Recommandation UIT-T G.9904:2012. Le présent paragraphe donne des informations relatives à la liaison des couches TCP-UDP/IPv6 à la SSCS IPv6. Voir Figure 2.

La SSCS IPv6 fournit une méthode efficace de transfert des paquets IPv6 sur le réseau PRIME.

Un nœud de service peut transmettre des paquets IPv6 au nœud de base ou directement à d'autres nœuds de service.

Par défaut, le nœud de base fait office de routeur entre le sous-réseau PRIME et le réseau dorsal. Un nœud de base doit avoir au moins cette capacité de connectivité. Un autre nœud à l'intérieur du sous-réseau peut également faire office de passerelle. Le nœud de base peut également faire office de routeur NAT. Toutefois, compte de l'abondance d'adresses IPv6, cela n'est pas prévu. La manière dont le nœud de base se connecte au réseau dorsal ne relève pas du domaine d'application du présent document.

###### 5.5.4.1.2 Attribution d'adressage IPv6 monodiffusion

Les nœuds de service IPv6 (et les nœuds de base) doivent prendre en charge le protocole IPv6 normalisé (voir le RFC 2460).

Les nœuds de service IPv6 (et les nœuds de base) doivent prendre en charge l'architecture d'adressage IPv6 normalisée (voir le RFC 4291).

Les nœuds de service IPv6 (et les nœuds de base) doivent prendre en charge les adressages IPv6 monodiffusion, les adresses IPv6 de liaison locale et les adresses IPv6 multicast (voir le RFC 4291).

Les nœuds de service IPv6 (et les nœuds de base) doivent prendre en charge la configuration d'adresse automatique à l'aide de la configuration d'adresse sans état (voir le RFC 4862). Ils peuvent également prendre en charge la configuration d'adresse automatique à l'aide de la configuration d'adresse avec état (voir le RFC 3315) et prendre en charge la configuration

manuelle des adresses IPv6. Le choix du schéma de configuration d'adresse à utiliser est spécifique au déploiement.

Les nœuds de service doivent prendre en charge le client DHCPv6, lorsque les nœuds de base prennent en charge le serveur DHCPv6 (voir le RFC 3315) pour la configuration d'adresse sans état.

#### **5.5.4.1.3 Rôle du nœud de base**

Au niveau de la SSCS IPv6, le nœud de base maintient un tableau contenant toutes les adresses monodiffusion IPv6 et les adresses EUI-48 qui y sont associées. L'un des rôles du nœud de base consiste à procéder à la résolution d'adresse IPv6 en adresse EUI-48. Chaque nœud de service appartenant au sous-réseau géré par le nœud de base enregistre son adresse IPv6 et l'adresse EUI-48 avec le nœud de base. Les autres nœuds de service peuvent alors demander au nœud de base de résoudre une adresse IPv6 en adresse EUI-48. Cela exige d'établir une connexion dédiée au nœud de base pour la résolution d'adresse.

Les en-têtes UDP/IPv6 peuvent éventuellement être compressés. La compression est négociée dans le cadre de la phase d'établissement de la connexion. Le présent document décrit actuellement une technique de compression d'en-tête, utilisée pour la transmission de paquets IPv6 sur des réseaux IEEE 802.15.4 (voir le RFC 6282). Elle est également appelée LOWPAN\_IPHC.

La multidiffusion des paquets IPv6 est prise en charge à l'aide du mécanisme de multidiffusion MAC.

#### **5.5.4.2 SSCS IPv6**

##### **5.5.4.2.1 Généralités**

La SSCS IPv6 présente deux types de connexion. Pour la résolution d'adresse, il existe une connexion au nœud de base. Pour le transfert de données IPv6, il existe une connexion par nœud de destination. Le nœud de base peut faire office de passerelle IPv6 vers le monde extérieur ou vers un autre nœud dans le même sous-réseau.

##### **5.5.4.2.2 Routage dans le sous-réseau**

Le routage des paquets IPv6 relève du domaine d'application de la SSCS IPv6. En d'autres termes, la SSCS IPv6 décide s'il convient d'envoyer le paquet directement à un autre nœud de service ou de le transférer à la passerelle configurée en fonction de l'adresse de destination IPv6.

Bien qu'IPv6 soit un protocole sans connexion, la SSCS IPv6 est orientée connexion. Après avoir procédé à la résolution d'adresse, une connexion est établie entre les nœuds de service source et de destination pour le transfert des paquets IPv6. Voir 5.5.4.11.4. Cette connexion est maintenue tout au long du transfert du trafic et peut être interrompue après une période d'inactivité.

##### **5.5.4.2.3 CPCS: Segmentation et réassemblage**

La sous-couche CPCS doit toujours être présente avec la SSCS IPv6 assurant les fonctions de segmentation et de réassemblage. Ainsi, les MSDU générées par la SSCS IPv6 sont toujours inférieures aux octets CMTUSize et les messages d'application sont réputés ne pas être plus longs que CIMaxAppPktSize.

### **5.5.4.3 Configuration d'adresse IPv6**

#### **5.5.4.3.1 Généralités**

Les nœuds de service peuvent utiliser les adresses IPv6 configurées manuellement, les adresses de liaison locale, l'autoconfiguration sans état conformément au RFC 4862 ou à DHCPv6 pour obtenir des adresses IPv6. Tous les nœuds doivent prendre en charge l'adresse de liaison locale monodiffusion en plus des autres adresses configurées ci-dessous, et les adresses multicast si le nœud appartient à des groupes multicast.

#### **5.5.4.3.2 Identifiant d'interface**

Pour utiliser l'autoconfiguration d'adresse sans état et les adresses de liaison locale, il est nécessaire de définir comment l'identifiant d'interface (voir le RFC 4291) est déduit. Chaque nœud PRIME comporte un EUI-48 unique. Cet EUI-48 est converti en EUI-64 de la même manière que pour les réseaux Ethernet (voir le RFC 2464). Cet EUI-64 est ensuite utilisé comme identifiant d'interface.

#### **5.5.4.3.3 Configuration d'adresse de liaison locale IPv6**

L'adresse de liaison locale IPv6 d'une interface PRIME est formée en ajoutant l'identifiant d'interface défini ci-dessus au préfixe FE80::/64.

#### **5.5.4.3.4 Autoconfiguration d'adresse sans état**

La longueur du préfixe d'adresse IPv6 d'une interface PRIME utilisée pour l'autoconfiguration sans état (voir le RFC 4862) doit être de 64 bits. Le préfixe IPv6 est obtenu par les nœuds de service auprès du nœud de base par l'intermédiaire des messages d'annonce de routeur, qui sont régulièrement envoyés par les routeurs ou à la demande du nœud de base.

#### **5.5.4.3.5 Autoconfiguration d'adresse avec état**

En variante, les adresses IPv6 peuvent être configurées à l'aide de DHCPv6 (voir le RFC 3315). DHCPv6 peut fournir à un dispositif des adresses attribuées par un serveur DHCPv6 et d'autres informations de configuration données en option.

#### **5.5.4.3.6 Adresse multicast**

Les nœuds de service IPv6 (et les nœuds de base) doivent prendre en charge l'adressage IPv6 multicast (voir le RFC 4291:2006, 2.7).

#### **5.5.4.3.7 Résolution d'adresse**

##### **5.5.4.3.7.1 Généralités**

La couche IPv6 présente la SSCS IPv6 avec un paquet IPv6 à transférer. La SSCS IPv6 est chargée de déterminer le nœud de service auquel il convient de livrer le paquet, grâce aux adresses IPv6 présentes dans le paquet. La SSCS IPv6 doit ensuite établir une connexion à la destination s'il n'en existe pas, de manière à pouvoir transférer le paquet. Deux classes d'adresses IPv6 peuvent être utilisées, et le paragraphe ci-dessous décrit la manière de les résoudre en adresses PRIME EUI-48. Il convient de noter qu'IPv6 ne dispose pas d'adresse de diffusion. Toutefois, la diffusion est possible à l'aide de la multidiffusion vers toutes adresses des nœuds.

##### **5.5.4.3.7.2 Adresse monodiffusion**

###### **5.5.4.3.7.2.1 Généralités**

Les adresses monodiffusion IPv6 doivent être résolues en adresses EUI-48 monodiffusion PRIME. Le nœud de base gère une base de données centrale d'adresses IPv6 et d'adresses EUI-48. Les fonctions de résolution d'adresse sont réalisées en interrogeant cette base de données. Le nœud de service doit établir une connexion au service de résolution d'adresse

fonctionnant sur le nœud de base, en utilisant la valeur TYPE TYPE\_CL\_IPv6\_AR (voir la Recommandation UIT-T G.9904:2012, Tableau C.1). Il convient de ne transmettre aucune donnée dans la signalisation d'établissement de connexion.

#### **5.5.4.3.7.2.2 Enregistrement et annulation d'enregistrement d'adresse**

Un nœud de service utilise le message AR\_REGISTERv6\_S pour enregistrer une adresse IPv6 et l'adresse EUI-48 correspondante. Le nœud de base accuse réception avec un message AR\_REGISTERv6\_B. Le nœud de service peut enregistrer plusieurs adresses IPv6 pour le même EUI-48.

Un nœud de service utilise le message AR\_UNREGISTERv6\_S pour annuler l'enregistrement d'une adresse IPv6 et de l'adresse EUI-48 correspondante. Le nœud de base accuse réception avec un message AR\_UNREGISTERv6\_B.

Si la connexion de résolution d'adresse entre le nœud de service et le nœud de base est fermée, il convient que le nœud de base supprime toutes les adresses associées à cette connexion.

#### **5.5.4.3.7.2.3 Consultation d'adresse**

Un nœud de service utilise le message AR\_LOOKUPv6\_S pour procéder à une consultation. Le message contient l'adresse IPv6 à résoudre. Le nœud de base doit répondre par un message AR\_LOOKUPv6\_B contenant un code d'erreur et, en l'absence d'erreur, l'adresse EUI-48 associée à l'adresse IPv6. Si la base de données du nœud de base comporte plusieurs entrées pour la même adresse IPv6, l'adresse EUI-48 renvoyée n'est pas identifiée.

Il convient de noter que dans le cas des adresses de liaison locale, et compte tenu du fait que l'adresse EUI-48 peut être obtenue à partir de l'adresse IPv6, la consultation peut simplement renvoyer cette valeur en l'extrayant de l'adresse IPv6.

#### **5.5.4.3.7.2.4 Adresse multicast**

Les IPv6 multicast sont mis en correspondance avec les descripteurs de connexion (ConnHandle) par la SSCS IPv6.

Pour rejoindre un groupe multicast, la SSCS IPv6 utilise la primitive MAC\_JOIN.request avec l'adresse IPv6 spécifiée dans le champ de données. Une primitive MAC\_JOIN.confirm correspondante est générée par la couche MAC à l'issue du processus. La primitive MAC\_Join.confirm contient le résultat (succès/échec) et le ConnHandle correspondant à utiliser par la SSCS IPv6. La couche MAC gère le transfert des données pour cette connexion à l'aide des LCID appropriés. Pour quitter le groupe multicast, la SSCS IPv6 au niveau du nœud de service doit utiliser la primitive MAC\_LEAVE.request(ConnHandle).

Pour envoyer un paquet multicast IPv6, la SSCS IPv6 envoie simplement le paquet au groupe à l'aide du ConnHandle attribué. Le ConnHandle est conservé si plusieurs paquets sont à envoyer. Toutefois, au bout de Tmcast\_reg secondes sans envoi de paquet multicast IPv6 au groupe, il convient que le nœud libère le ConnHandle à l'aide de la primitive MAC\_LEAVE.request. La valeur nominale de Tmcast\_reg est 10 min, mais d'autres valeurs peuvent être utilisées.

NOTE Tmcast\_reg est le délai d'annulation d'enregistrement du groupe multicast. Voir la Recommandation UIT-T G.9904:2012, 9.4.2.4.

#### **5.5.4.3.7.2.5 Retransmission des paquets de résolution d'adresse**

La connexion entre le nœud de service et le nœud de base pour la résolution d'adresse n'est pas fiable. Le service MAC ARQ n'est pas utilisé. Le nœud de service est chargé des retransmissions si le nœud de base ne répond pas dans la seconde qui suit. Il ne s'agit pas d'une erreur si le nœud de base reçoit plusieurs fois les mêmes demandes d'enregistrement

ou s'il lui est demandé de supprimer un enregistrement qui n'existe pas. Ces conditions peuvent être le résultat des retransmissions.

#### 5.5.4.4 Transfert de paquet IPv6

Pour les paquets à transférer, il est nécessaire d'établir une connexion entre les nœuds source et de destination (voir la Figure A.1). La SSCS IPv6 examine chaque paquet IP afin de déterminer l'adresse EUI-48 de destination. Si une connexion à la destination a déjà été établie, le paquet est simplement envoyé. À cet effet, la SSCS IPv6 gère un tableau pour chaque connexion dont elle fait l'objet avec les informations présentées au Tableau 15. Pour utiliser ce tableau, il est en premier lieu nécessaire de déterminer si l'adresse distante se trouve dans le sous-réseau local ou si une passerelle doit être utilisée. Pour ce faire, le masque de réseau associé à l'adresse IPv6 locale est utilisé. Si l'adresse de destination n'est pas dans le sous-réseau local, l'adresse de la passerelle est utilisée à la place de l'adresse de destination lorsque le tableau fait l'objet d'une recherche.

**Tableau 15 – Entrée du tableau de la SSCS IPv6**

| Paramètre             | Description                                            |
|-----------------------|--------------------------------------------------------|
| CL_IPv6_Con.Remote_IP | Adresse IP distante de cette connexion                 |
| CL_IPv6_Con.ConHandle | Descripteur de connexion MAC pour la connexion         |
| CL_IPv6_Con.LastUsed  | Horodatage du dernier paquet reçu/transmis             |
| CL_IPv6_Con.HC        | Schéma de compression d'en-tête en cours d'utilisation |

La SSCS IPv6 peut fermer une connexion si elle n'a pas été utilisée pendant une durée définie par la mise en œuvre. Si la connexion est fermée, l'entrée correspondante est supprimée aux deux extrémités de la connexion.

Si une connexion à la destination n'existe pas, un travail supplémentaire est nécessaire. Le dispositif de résolution d'adresse est utilisé pour déterminer l'adresse EUI-48 de l'adresse IP distante si elle est locale ou la passerelle associée à l'adresse locale si l'adresse de destination se trouve dans un autre sous-réseau. Si le nœud de base répond avec l'adresse EUI-48 du nœud de service de destination, une connexion MAC est établie au dispositif distant. La valeur TYPE de cette connexion établie est TYPE\_CL\_IPv6\_UNICAST (voir la Recommandation UIT-T G.9904:2012, Tableau C.1). Les données transmises dans le message de demande sont définies en 5.5.4.11.3. L'adresse IPv6 locale est fournie de sorte que le dispositif distant puisse ajouter la nouvelle connexion dans son cache des connexions pour envoyer les données dans le sens opposé. L'utilisation de la compression d'en-tête est également négociée dans le cadre de l'établissement de la connexion. Dès que la connexion MAC a été établie, le paquet IPv6 peut être envoyé.

#### 5.5.4.5 Segmentation et réassemblage

La SSCS IPv6 doit prendre en charge les paquets IPv6 avec une MTU de 1 500 octets. Cela exige l'utilisation du service de segmentation et de réassemblage de la partie commune de la Sous-couche de convergence.

#### 5.5.4.6 Compression

Un dispositif PRIME est réputé être en mesure d'assurer la compression/décompression d'en-tête IPv6LOWPAN\_IPHC. Il peut également être capable de procéder à une compression/décompression UDP. La compression UDP/IPv6 est donc négociée.

Aucune négociation ne peut avoir lieu pour les paquets multicast. Les nœuds peuvent uniquement utiliser les fonctions de compression obligatoires.

En fonction du type d'adresse IPv6 transportée par le paquet et des capacités négociées entre les nœuds concernés par l'échange de données, la compression d'en-tête IPv6 est réalisée.

Tous les nœuds de service et le nœud de base doivent prendre en charge la compression d'en-tête IPv6 en utilisant une compression sans état d'adresses source et de destination (voir le RFC 6282). Les adresses IPv6 source et de destination utilisant la compression avec état et la compression d'en-tête IPv6 Next sont négociables.

#### 5.5.4.7 Qualité de la mise en correspondance de service

Dans la spécification de la couche PRIME MAC, il est indiqué que le mécanisme d'accès basé sur la contention prend en charge 4 niveaux de priorité (1-4). Le niveau 1 est utilisé pour les messages de signalisation MAC, mais pas exclusivement.

Les paquets IPv6 incluent un champ de classe de trafic en en-tête pour indiquer la qualité de service (QoS) que le paquet souhaiterait recevoir. La classe de trafic peut être utilisée de la même manière qu'IPv4 ToS (voir la Recommandation UIT-T G.9904:2012, 4.6). En effet, trois bits du ToS indiquent la précedence IP. Le Tableau 16 spécifie la manière dont la précedence IP mise en correspondance avec la priorité PRIME MAC.

**Tableau 16 – Mise en correspondance de la précedence IPv6 avec la priorité PRIME MAC**

| Précedence IP                                                                                                                                                                              | Priorités MAC |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|
| 000 – Routine                                                                                                                                                                              | 4             |
| 001 – Priorité                                                                                                                                                                             | 4             |
| 010 – Immédiate                                                                                                                                                                            | 3             |
| 011 – Flash                                                                                                                                                                                | 3             |
| 100 – Précedence Flash                                                                                                                                                                     | 2             |
| 101 – Critique                                                                                                                                                                             | 2             |
| 110 – Contrôle entre réseaux                                                                                                                                                               | 1             |
| 111 – Contrôle de réseau                                                                                                                                                                   | 1             |
| NOTE Au niveau de la couche MAC la priorité définie dans le champ d'en-tête de paquet est la valeur attribuée dans le présent tableau moins 1, la plage du champ PKT.PRIO allant de 0 à 3. |               |

#### 5.5.4.8 Ouverture et fermeture de la SSCS IPv6

##### 5.5.4.8.1 Généralités

Les services suivants sont utilisés pour ouvrir et fermer la SSCS IPv6. À un moment donné, il ne peut exister qu'une seule connexion IPv6 SSCS. Par conséquent, la SSCS IPv6 ne doit être ouverte qu'une seule fois. La couche IPv6 peut fermer la SSCS IPv6 lorsque l'interface IPv6 est réduite. La SSCS IPv6 se ferme également lorsque la connexion MAC sous-jacente au nœud de base est perdue.

NOTE Pour de plus amples informations, voir la Recommandation UIT-T G.9904:2012, 9.6.9.2.

##### 5.5.4.8.2 CL\_IPv6\_ESTABLISH.request

*Fonction*

Cette primitive de service est émise par la couche IPv6 du nœud de service à la SSCS IPv6. Cela est assuré à chaque fois que la couche IPv6 relance l'interface.

*Paramètres de service*

La sémantique de cette primitive est la suivante:

```
CL_IPv6_ESTABLISH.request()
```

#### *Utilisation*

Lors de la réception de cette primitive de service, la SSCS IPv6 du nœud de service assure la connexion de résolution d'adresse au nœud de base.

#### **5.5.4.8.3 CL\_IPv6\_Establish.confirm**

##### *Fonction*

Cette primitive de service est transmise de la SSCS IPv6 du nœud de service à la couche IPv6. En cas de succès, elle signale que la SSCS IPv6 est prête à accepter les paquets IPv6 à envoyer aux homologues.

##### *Paramètres de service*

La sémantique de cette primitive est la suivante:

```
CL_IPv6_Establish.confirm()
```

#### *Utilisation*

Cette primitive de service est utilisée pour fournir le résultat du service CL\_IPv6\_Establish.request. Dès que la SSCS IPv6 a établi toutes les connexions et est prête à transmettre et recevoir des paquets IPv6, cette primitive est transmise à la couche IPv6.

Si la SSCS IPv6 rencontre une erreur au moment de l'ouverture, elle répond par une primitive CL\_IPv6\_Release.confirm au lieu d'une primitive CL\_IPv6\_Establish.confirm.

#### **5.5.4.8.4 CL\_IPv6\_Release.request**

##### *Fonction*

Cette primitive de service est utilisée par la couche IPv6 du nœud de service à la SSCS IPv6 pour demander la fermeture de la connexion IPv6 SSCS.

##### *Paramètres de service*

La sémantique de cette primitive est la suivante:

```
CL_IPv6_Release.request()
```

#### *Utilisation*

Cette primitive est utilisée par le nœud de service lorsque la connexion doit être fermée pour une raison quelconque. La SSCS IPv6 ferme toutes les connexions. Toutes les ressources sont libérées et plus aucun autre paquet IPv6 ne peut être reçu ou envoyé. Lorsque la SSCS IPv6 a libéré toutes ses connexions et ressources, elle renvoie une primitive de service CL\_IPv6\_Release.confirm.

#### **5.5.4.8.5 CL\_IPv6\_Release.confirm**

##### *Fonction*

Cette primitive de service est utilisée par la SSCS IPv6 du nœud de service pour indiquer à la couche IPv6 que la SSCS IPv6 a été fermée. Cela peut être le résultat d'une primitive CL\_IPv6\_Release.request, d'une primitive CL\_IPv6\_Establish.request qui n'a pas abouti ou de la couche MAC qui indique que la connexion de résolution d'adresse a été perdue ou que le nœud de service lui-même n'est plus enregistré.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_Release.confirm(Result)

Les valeurs possibles du paramètre Result sont indiquées dans le Tableau 1.

#### *Utilisation*

La SSCS IPv6 utilise ce service pour informer la couche IPv6 que le nœud de service a quitté le sous-réseau.

### **5.5.4.9 Gestion d'adresses monodiffusion**

#### **5.5.4.9.1 Généralités**

Les services définis ici sont utilisés pour la gestion d'adresses, c'est-à-dire pour l'enregistrement et l'annulation d'enregistrement des adresses IPv6 monodiffusion avec la SSCS IPv6.

#### **5.5.4.9.2 CL\_IPv6\_Register.request**

##### *Fonction*

Cette primitive est transmise par la couche IPv6 du nœud de service à la SSCS IPv6 pour demander l'enregistrement d'une adresse IPv6.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_REGISTER.req(  
IPv6\_Address,  
Netmask,  
Gateway)

Où:

- IPv6\_Address est l'adresse à enregistrer;

NOTE Un nœud de service peut enregistrer plusieurs adresses IPv6 monodiffusion.

- Netmask est le masque de réseau. Il est utilisé par la SSCS IPv6 pour déterminer s'il convient de livrer le paquet directement ou s'il convient d'utiliser la passerelle;
- Gateway est une adresse IPv6 de la passerelle à utiliser si l'adresse de destination est à l'extérieur du sous-réseau.

Chaque sous-réseau PRIME OFDM ne pouvant avoir qu'un seul nœud de base, ce dernier peut faire office de passerelle entre le sous-réseau PRIME et le réseau dorsal. Tous les nœuds de base doivent disposer de ces capacités de connectivité. Un autre nœud à l'intérieur du sous-réseau peut également faire office de passerelle.

#### *Utilisation*

À partir du côté du nœud de service, cet enregistrement est remonté au nœud de base.

Si le nœud de service a correctement enregistré son adresse IPv6 avec le nœud de base, une primitive CL\_IPv6\_REGISTER.cnf est utilisée. Si l'enregistrement n'a pas abouti, la primitive CL\_IPv6\_Release.cnf est utilisée.

#### **5.5.4.9.3 CL\_IPv6\_Register.confirm**

##### *Fonction*

Cette primitive de service est transmise par la SSCS IPv6 du nœud de service à la couche IPv6 pour indiquer que l'enregistrement de l'adresse IPv6 a abouti.

##### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_REGISTER.cnf(IPv6\_Address)

IPv6\_Address est l'adresse dont l'enregistrement a abouti.

##### *Utilisation*

La primitive de service CL\_IPv6\_Register.cnf est utilisée par la SSCS IPv6 du nœud de service à l'issue de l'enregistrement. La couche IPv6 peut alors utiliser cette adresse pour envoyer des paquets IPv6.

#### **5.5.4.9.4 CL\_IPv6\_UNREGISTER.request**

##### *Fonction*

Cette primitive de service est transmise par la couche IPv6 du nœud de service à la SSCS IPv6 pour demander l'annulation d'enregistrement d'une adresse IPv6.

##### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_UNREGISTER.req(IPv6\_Address)

IPv6\_Address est l'adresse dont l'enregistrement est à annuler.

##### *Utilisation*

Dès que l'enregistrement de l'adresse IPv6 a été annulé avec le nœud de base, une primitive CL\_IPv6\_UNREGISTER.confirm est utilisée. Si CL\_IPv6\_Unregister.request échoue, une primitive CL\_IPv6\_Release.confirm est renvoyée. La raison de l'échec est indiquée dans le résultat.

Plusieurs adresses IPv6 pouvant être enregistrées par le nœud de service, seul l'enregistrement de l'adresse concernée est annulé par cette requête. Les adresses dont l'enregistrement n'a pas été annulé sont toujours valables, jusqu'à ce qu'elles le soient ou qu'une primitive soit libérée.

#### 5.5.4.9.5 CL\_IPv6\_Unregister.confirm

##### *Fonction*

Cette primitive de service est transmise par la SSCS IPv6 du nœud de service à la couche IPv6 pour indiquer que l'annulation de l'enregistrement de l'adresse IPv6 a abouti.

##### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_UNREGISTER.confirm(IPv6\_Address)

IPv6\_Address est l'adresse IPv6 unicast dont l'enregistrement a été annulé.

##### *Utilisation*

Dès que l'annulation de l'enregistrement a abouti, la couche IPv6 n'envoie pas de paquets IPv6 à l'aide de cette adresse source.

#### 5.5.4.10 Gestion de groupes multicast

##### 5.5.4.10.1 Généralités

Le présent paragraphe décrit les primitives utilisées pour gérer les groupes multicast.

NOTE Noter que les primitives CL\_432\_JOIN et CL\_IPv6\_MUL\_JOIN sont de nature différente. Le premier groupe informe le nœud de base des nouveaux nœuds de service qui ont rejoint le réseau. Le deuxième groupe gère l'abonnement à un certain groupe de multicast IP. De même, noter que le nom de cet ensemble de primitives contient le MUL, ce qui permet de faire clairement la distinction.

##### 5.5.4.10.2 CL\_IPv6\_MUL\_Join.request

##### *Fonction*

Cette primitive est transmise par la couche IPv6 à la couche de convergence IPv6. Elle contient l'adresse multicast IPv6 à rejoindre.

##### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_MUL\_JOIN.request(IPv6\_Address)

IPv6\_Address est l'adresse du groupe multicast IPv6 à rejoindre.

##### *Utilisation*

Lorsque la couche de convergence reçoit cette primitive, elle assure que les paquets IP envoyés à ce groupe sont des paquets multicast dans le réseau PRIME et de recevoir les paquets à l'aide de cette adresse à transmettre à la pile IPv6. Si la SSCS IPv6 ne peut pas rejoindre le groupe, il utilise la primitive CL\_IPv6\_MUL\_LEAVE.confirm. Sinon, la primitive CL\_IPv6\_MUL\_JOIN.confirm est utilisée pour indiquer le succès.

##### 5.5.4.10.3 CL\_IPv6\_MUL\_Join.confirm

##### *Fonction*

Cette primitive est transmise par la couche de convergence IPv6 à la couche IPv6. Elle contient un statut de résultat et une adresse multicast IPv6 qui a été rejointe.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_MUL\_JOIN.confirm(IPv6\_Address)

IPv6\_Address est l'adresse du groupe multicast IPv6 qui a été rejoint.

#### *Utilisation*

La couche de convergence commence à transférer les paquets multicast IPv6 pour le groupe multicast donné.

#### **5.5.4.10.4 CL\_IPv6\_MUL\_Leave.request**

##### *Fonction*

Cette primitive est transmise par la couche IPv6 à la couche de convergence IPv6. Elle contient l'adresse multicast IPv6 à quitter.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_MUL\_LEAVE.request(IPv6\_address)

IPv6\_address est l'adresse du groupe multicast IPv6 à quitter.

#### *Utilisation*

La couche de convergence arrête de transférer les paquets multicast IPv6 de ce groupe et peut quitter le groupe multicast PRIME MAC.

#### **5.5.4.10.5 CL\_IPv6\_MUL\_Leave.confirm**

##### *Fonction*

Cette primitive est transmise par la couche de convergence IPv6 à la couche IPv6. Elle contient un statut de résultat et une adresse multicast IPv6 qui a été quittée.

#### *Paramètres de service*

La sémantique de cette primitive est la suivante:

CL\_IPv6\_MUL\_LEAVE.confirm(IPv6\_address, Result)

IPv6\_address est le groupe multicast IPv6 qui a été quitté.

#### *Utilisation*

La couche de convergence arrête de transférer les paquets multicast IPv6 pour le groupe multicast donné.