

IEEE Standard for Utility Industry Metering Communication Protocol Application Layer (End Device Data Tables)

Sponsored by the

IEEE Standards Coordinating Committee 31 on
Automatic Meter Reading and Energy Management

IEEE
3 Park Avenue
New York, NY 10016-5997
USA

IEEE Std 1377™-2012
(Revision of
IEEE Std 1377-1997)

10 August 2012

IEEE Standard for Utility Industry Metering Communication Protocol Application Layer (End Device Data Tables)

Sponsor
**IEEE Standards Coordinating Committee 31 on
Automatic Meter Reading and Energy Management**

Approved 14 May 2012

IEEE-SA Standards Board

Abstract: Common structures are provided in this standard for encoding data in communication between End Devices (meters, home appliances, IEEE 1703 Nodes) and Utility enterprise collection and control systems using binary codes and Extensible Markup Language (XML) content. The Advanced Metering Infrastructure (AMI) and SmartGrid requirements are addressed as identified by the Office of Electricity Delivery and Energy Reliability of the U.S. Department of Energy and by the Smart Metering Initiative of the Ontario Ministry of Energy (Canada) and of Measurement Canada. Sets of tables are exposed that are grouped together into sections that pertain to a particular feature-set and related function such as Time-of-use, Load Profile, Security, Power Quality, and more. Each standard Table Set (Data Model) can be expanded or restricted by the Manufacturer of the IEEE 1377 Device or home appliance using XML/TDL descriptive registered syntax (XML-based Table Definition Language) and enterprise data-value management using EDL (Exchange Data Language) in a manner that is machine readable. Published jointly with NEMA and Measurement Canada, Tables are provided in support of Gas, Water, and Electric sensors and related appliances. Tables are also provided for network configuration and management by referencing its companion standard IEEE Std 1703™-2012. IEEE Std 1377-2012 is co-published as ANSI C12.19 and MC12.19.

Keywords: ANSI C12.19, End Device, IEEE 1377, Utility Tables

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Copyright © 2012 by The Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 10 August 2012. Printed in the United States of America.

IEEE is a registered trademark in the U.S. Patent & Trademark Office, owned by The Institute of Electrical and Electronics Engineers, Incorporated.

PDF: ISBN 978-0-7381-7274-3 **STD97249**
Print: ISBN 978-0-7381-7380-1 **STDPD97249**

IEEE prohibits discrimination, harassment, and bullying. For more information, visit
<http://www.ieee.org/web/aboutus/whatis/policies/p9-26.html>.

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

Notice and Disclaimer of Liability Concerning the Use of IEEE Documents: IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

Use of an IEEE Standard is wholly voluntary. IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon any IEEE Standard document.

IEEE does not warrant or represent the accuracy or content of the material contained in its standards, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained in its standards is free from patent infringement. IEEE Standards documents are supplied "AS IS."

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE standard is subjected to review at least every ten years. When a document is more than ten years old and has not undergone a revision process, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE standard.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

Translations: The IEEE consensus development process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE should be considered the approved IEEE standard.

Official Statements: A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position of IEEE.

Comments on Standards: Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE. However, IEEE does not provide consulting information or advice pertaining to IEEE Standards documents. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important to ensure that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to comments or questions except in those cases where the matter has previously been addressed. Any person who would like to participate in evaluating comments or revisions to an IEEE standard is welcome to join the relevant IEEE working group at <http://standards.ieee.org/develop/wg/>.

Comments on standards should be submitted to the following address:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854-4141
USA

Photocopies: Authorization to photocopy portions of any individual standard for internal or personal use is granted by The Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Notice to users

Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not imply compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Copyrights

This document is copyrighted by the IEEE. It is made available for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making this document available for use and adoption by public authorities and private users, the IEEE does not waive any rights in copyright to this document.

Updating of IEEE documents

Users of IEEE Standards documents should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect. In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE-SA Website at <http://standards.ieee.org/index.html> or contact the IEEE at the address listed previously. For more information about the IEEE Standards Association or the IEEE standards development process, visit IEEE-SA Website at <http://standards.ieee.org/index.html>.

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: <http://standards.ieee.org/findstds/errata/index.html>. Users are encouraged to check this URL for errata periodically.

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE-SA Website at <http://standards.ieee.org/about/sasb/patcom/patents.html>. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or nondiscriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this IEEE standard was originally completed, the End Device/Telemetry Interface Unit Subcommittee Working Group had the following membership:

Avygdor Moise, *Chair*
Kostas Tolios, *Vice Chair*
Michael Anderson, *Vice Chair*
Terry L. Penn, *Editor*

Edward J. Berozet
Brent Cain
David Haynes
Brad Johnson

Herman Millican
Lawrence Kotewa
Vuong Nguyen
Dan Nordell

Aaron Snyder
Richard Tucker
Michel Veillette
Virginia Zinkowski

We also wish to thank the following members of ANSI C12 SC17 WG2 for their significant contributions to the making of this standard:

Paul Aubin
Shawn Bagley
Norbert Balko
Ron Begley
William Beverly
Gregory Bray
William Buckley
Martin Burns
Kah-Fae Chan
Thomas Chen
Stuart Garland
Raymond Gaudreault

Ken Gilmer
Greg Gomez
Don Grundhauser
Mark Iacoviello
Janice Jennings
Jean Joly
Alex Lunycz
Keith Martin
Peter Martin
William Mazza
Robert McMichael
John Newbury
Jack Pazdon

Marc Purc
Bin Qiu
Wesley Ray
William Rush
Chris Schafer
Brian Seal
Kendall Smith
Richard Stetler
David Taylor
John Taylor
David Wiseman
Ted York

The following members of the individual balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Ali Al Awazi
Michael Anderson
Chris Bagge
Hugh Barrass
H. Stephen Berger
William Byrd
Brent Cain
Keith Chow
Sourav Dutta
Herbert Falk
Sergiu Goma
Randall Groves
David Halasz
David Haynes
Werner Hoelzl

Liang Ji
Piotr Karocki
Stanley Klein
Lawrence Kotewa
Jim Kulchisky
Paul Lambert
Richard Lancaster
Greg Luri
Faramarz Maghsoodlou
Anthony Mazy
Gary Michel
Avygdor Moise
Jose Morales
Michael S. Newman
Satoshi Obara

Chris Osterloh
Amit Patel
Terry L. Penn
Charles Petras
Michael Rush
Bartien Sayogo
Gil Shultz
James Smith
Jerry Smith
Aaron Snyder
James St. Clair
David Tepen
Richard Tucker
Scott Weikel
Paul Work

When the IEEE-SA Standards Board approved this standard on 14 May 2012, it had the following membership:

Richard H. Hulett, *Chair*
John Kulick, *Vice Chair*
Robert Grow, *Past Chair*

Satish Aggarwal
Masayuki Ariyoshi
Peter Balma
William Bartley
Ted Burse
Clint Chaplin
Wael Diab
Jean-Philippe Faure

Alexander Gelman
Paul Houzé
Jim Hughes
Young Kyun Kim
Joseph L. Koepfinger*
David J. Law
Thomas Lee
Hung Ling

Oleg Logvinov
Ted Olsen
Gary Robinson
Jon Walter Rosdahl
Mike Seavey
Yatin Trivedi
Phil Winston
Yu Yuan

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Richard DeBlasio, *DOE Representative*
Michael Janezic, *NIST Representative*

Catherine Berger
IEEE Standards Program Manager, Document Development

Matthew J. Ceglia
IEEE Client Services Manager, Professional Services

Introduction

This introduction is not part of IEEE Std 1377-2012, IEEE Standard for Utility Industry Metering Communication Protocol Application Layer (End Device Data Tables).

The body of this standard was developed jointly with ANSI C12.19 and MC12.19. The joint agreement calls for the standards and regulatory organizations IEEE, ANSI, and MC to maintain the body of this standard in step and congruent as they publish versions and revisions of the standard. A number of editorial errors and error of omission were discovered since the publication of ANSI C12.19-2008, just before the acceptance of IEEE Std 1377-2012. These errors are listed in Annex N, “(normative) Listing of editorial errors and errors of omission in ANSI C12.19-2008”. All reported errors were identified in the body of this standard and **highlighted** to indicate that the text was in error and was corrected. Errors of omission were corrected in the body of this document and similarly **highlighted**. Other minor editorial changes have been made to this standard, but they have not been highlighted.

The second version of this standard was developed under the auspices of the secretariat of ANSI ASC 12 SC17 WG2. For that reason, the document contains references to ANSI standards, which in a number of them are cross published by the IEEE. These standard references can be interchangeably replaced with the corresponding IEEE standards (or MC reference numbers) as follows:

ANSI Standard	IEEE Standard	MC Reference
ANSI C12.18	IEEE Std 1701 TM	MC12.18
ANSI C12.19	IEEE Std 1377 TM	MC12.19
ANSI C12.21	IEEE Std 1702 TM	MC12.21
ANSI C12.22	IEEE Std 1703 TM	MC12.22

All communications to Measurement Canada (MC) in regard to MC referenced Standard should be directed to:

Measurement Canada
Standards Building #4
151 Tunney’s Pasture Driveway
Ottawa, ON K1A 0C9 Canada

IEEE Std 1377-2012 provides a common data structure for use in transferring data to and from utility End Devices, typically meters. It has been approved after considerable cooperative effort among utilities, meter manufacturers, automated meter reading service companies, ANSI, Measurement Canada (for Industry Canada), NEMA, IEEE, Utilimetrics, and other interested parties.

The standard data structure is defined as sets of tables. The tables are grouped together into sections called decades. Each Decade pertains to a particular feature-set and related function such as Time-of-use, Load Profile, and so on. Table data is transferred from or to the End Device by reading from or writing to a particular table or portion of a table. Specific actions may be requested to be executed by the End Device through Procedure invocations.

The second release of the standard represents a philosophical departure from the first release in 1997. This release of the standard is intended to accommodate the concept of an advanced metering infrastructure such as that identified

by the Office of Electricity Delivery and Energy Reliability of the U.S. Department of Energy; the Smart Metering Initiative of the Ontario Ministry of Energy (Canada); and the stated requirements of Measurement Canada for the approval of a metering device for use in Canada.

This standard covers a broader range of functionality relative to its previous version; however, it does not follow that implementations of this broader range of functionality need to be large or complex. Implementers are encouraged to choose an appropriate subset that is suitable for their needs. Therefore, it is very unlikely for any one End Device to embed all tables or even the majority of the tables described herein. Implementers are encouraged to deploy their desired functionalities using complete and consistent suites of standard tables from standard decades to the largest extent practical.

Readers who are acquainted with IEEE Std 1377-1997 will discover that the new version contains many changes. The changes may be categorized as follows:

- Additions of new features or new functionality through the introduction of new tables, decades, or syntax.
- Extensions or improvements to preexisting functionality in support of contemporary and anticipated industry needs.
- Corrections of errors and provision of clarifications that address known deficiencies and documentation of the accepted or anticipated industry practice.
- Introduction of XML-based table description language (TDL/EDL) and documentation of services supporting the needs expressed by initiatives such as Smart Metering, Advanced Metering Infrastructure, and the like.
- Removal of features that were found undesirable.
- Provision of guidelines for implementers of this standard and developers of future versions of the standard on backward compatibility and planned phase-out of obsolete features.

Some of the many new features introduced in this revision of the standard include:

- Addition of new data types in support of high-precision sub-second time stamps, such as HTIME_DATE and HTIME.
- Creation of new syntax for the aggregation of transmitted characters into strings, STRING.
- Creation of new syntax for the aggregation of transmitted Octets into opaque blobs, BINARY.
- Addition of new Decade 9, Telephone tables. This Decade imports and supersedes the tables and syntax defined in ANSI C12.21/MC12.21/IEEE 1702.
- Creation of new Decade 11, Load Control and Pricing Tables. This Decade provides for demand-side management capabilities, including load control, demand response, prepayment, and direct customer billing functions.
- Creation of Decade 14, Extended User-defined Tables. This Decade provides capability for transmitting and receiving a contiguous stream of element and subelement data. The data are referenced methodically to the legitimate and formal IEEE 1377 data element.
- Creation of Decade 15, Quality-of-service Tables. This Decade provides control, capture, and logging of high-precision Quality-of-service performance indicators, such as power quality, and the inclusion of detailed waveforms, power spectra, and related information.
- Creation of Decade 16, One-way Devices Tables. This Decade provides control and management tables in support of one-way (talk-only) End Devices.

Some extensions or improvements to preexisting functionality include:

- Table 0, General Configuration Table, retains its original form and it is backward compatible with IEEE Std 1377-1997. However, a few notable control elements were modified:
 - 1) Extended CHAR_FORMAT to support UTF-8.
 - 2) Replaced MANUFACTURER element with DEVICE_CLASS element to facilitate a more dynamic End Device data model recognition.
 - 3) Extended TM_FORMAT in support of high-precision, sub-second, time stamps.
- Changed all data source selections in all tables (e.g., Table 22) from UINT8 to SOURCE_SELECT_RCD. This form is backward and forward compatible with IEEE Std 1377-1997 when MODEL_SELECT is 0.
- Table 3, End Device Mode Status Table, incorporates the changes proposed in ANSI C12.21/MC12.21/IEEE 1702.
- Table 7, Procedure Initiate Table, and Table 8, Procedure Response Table, were updated and extended to provide procedures in support of new functionality that is needed by:
 - 1) Decade 7, History and Event Log Tables.
 - 2) Decade 9, Telephone Control Tables.
 - 3) Decade 11, Load Control and Pricing Tables.
 - 4) Decade 12, Network Control Tables.
 - 5) Decade 15 Quality-of-service Tables.
- Table 12, Unit of Measure Entry table, was expanded with new Units of Measure (UOMs).
- Table 17, Transformer Loss Compensation, was added to Decade 1.
- Extended the event log tables to provide a new capability to manage and detect program changes through the life of the End Device.
- Extended the event log tables to provide a new capability to manage, authenticate, and verify the integrity of data read from an End Device service point by any receiving client application that may reside across any communication system.
- Extended the table type that can be transmitted to include up to 2040 Extended-user-defined tables via TABLE_IDA_BFLD. Also introduced a new Decade 14, Extended User-defined Tables, to manage the collation of table elements, down to the bit level, into extended-user-defined tables.

Corrections of errors and clarifications include:

- Section 7.0, Compliance, was updated to reduce variations in the implementation of the standard.
- Section 8.0, Table Transportation Issues, was updated to include the “Errata to ANSI Standard C12.19-1997, Utility Industry Data Tables” published in 2001. It also was updated to harmonize this standard with ANSI C12.18/MC12.18/IEEE 1701, ANSI C12.21/MC12.21/IEEE 1702, and ANSI C12.22/MC12.22/IEEE 1703.
- Table 7, Procedure Initiate Table, and Table 8, Procedure Response Table, indices calculations were completely rewritten to eliminate a fatal error in the production of unique indices for procedure parameters when using index/count access methods.

Removal of features that were considered undesirable:

- Section 5.0, Syntax, no longer supports two-dimensional arrays. Appropriate corrections were applied wherever this syntax was used. This does not impact the offset/count data access methods.

The reader is encouraged to review the descriptive text as it brings significant clarifications and examples of use where appropriate.

Contents

1	Overview	1
1.1	Scope	1
1.2	Purpose	1
2	Normative references	2
3	Definitions	4
4	General	13
4.1	Standard Tables	13
4.2	Manufacturer Tables	15
4.3	Packed Record, Bit Field, and Element properties	17
4.4	Extended User-defined Tables properties	17
5	Syntax	18
5.1	Descriptive syntax	18
6	Special data types	19
6.1	Character set selection	19
6.2	Noninteger formats	19
6.3	Date and time formats	21
6.4	Common table or procedure identifier formats	29
7	Compliance and compatibility	33
7.1	Compliance	33
7.2	Backward and forward compatibility	34
8	Table transportation issues	35
8.1	Minimum services and parameters	35
8.2	Pending Event description	40
8.3	List management description	44
9	Tables	46
9.1	Decade 0: General Configuration Tables	46
9.2	Decade 1: Data Source Tables	98
9.3	Decade 2: Register Tables	126
9.4	Decade 3: Local Display Tables	141
9.5	Decade 4: Security Tables	150
9.6	Decade 5: Time and Time-of-Use Tables	160
9.7	Decade 6: Load Profile Tables	181
9.8	Decade 7: History Log and Event Log Tables	210
9.9	Decade 8: User-defined Tables	238
9.10	Decade 9: Telephone Control Tables	252
9.11	Decade 10: Unassigned	273
9.12	Decade 11: Load Control and Pricing Tables	274
9.13	Decade 12: Reserved	295
9.14	Decade 13: Reserved	295
9.15	Decade 14: Extended User-defined Tables	296
9.16	Decade 15: Quality-of-service	308
9.17	Decade 16: One-way Devices	357

Annex A	(informative) Reserved Device Classes for meter equipment manufacturers implementing ANSI C12.19-1997 devices	372
Annex B	(normative) History and event log codes	373
Annex C	(normative) Default Sets for Decade Tables	383
Annex D	(normative) Indices for partial table read/write access	389
Annex E	(informative) Event Logger implementation	391
Annex F	(informative) Transformer losses compensation.....	394
Annex G	(normative) Document-form descriptive syntax	397
Annex H	(informative) Date-time elements conversion algorithm (TM_FORMAT=3 and TM_FORMAT=4).....	433
Annex I	(normative) XML file format of TDL and EDL files	435
Annex J	(normative) Universal Identifier	511
Annex K	(informative) Algorithms for the conversion of Table Element values to engineering units.....	512
Annex L	(informative) Registering or updating DEVICE CLASS OID.....	522
Annex M	(informative) Bibliography	540
Annex N	(normative) Listing of editorial errors and errors of omission in ANSI C12.19-2008.....	541

IEEE Standard for Utility Industry Metering Communication Protocol Application Layer (End Device Data Tables)

IMPORTANT NOTICE: IEEE Standards documents are not intended to ensure safety, health, or environmental protection, or ensure against interference with or from other devices or networks. Implementers of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.

This IEEE document is made available for use subject to important notices and legal disclaimers. These notices and disclaimers appear in all publications containing this document and may be found under the heading “Important Notice” or “Important Notices and Disclaimers Concerning IEEE Documents.” They can also be obtained on request from IEEE or viewed at <http://standards.ieee.org/TPR/disclaimers.html>.

1 Overview

1.1 Scope

This standard defines a Table structure for utility application data to be passed between an End Device and any other device. It neither defines device design criteria nor specifies the language or protocol used to transport that data. The Tables defined in this standard represent a data structure that shall be used to transport the data, not necessarily the data storage format used inside the End Device.

1.2 Purpose

The Utility Industry has a need for a standard that provides an interoperable “plug-and-play” environment for field metering devices. The purpose of this standard is to define the framework and data structures for transporting Utility End Device data to and from End Devices and for use by enterprise systems.

This standard is intended to accommodate the concept of an advanced metering infrastructure such as that identified by the Office of Electricity Delivery and Energy Reliability of the U.S. Department of Energy; the Smart Metering Initiative of the Ontario Ministry of Energy (Canada); and the stated requirements of Measurement Canada for the approval of a metering device for use in Canada.

This standard is to provide a uniform, structured, and adaptive data model, such that Utility End Devices and ancillary devices (e.g., home appliances and communication technology) can operate in a “plug-and-play” and multisource enterprise **Advanced Metering Infrastructure (AMI)** environment.

This standard extends the definitions provided by IEEE Std 1377-1997 to include provisions for enterprise-level asset management, data management, and uniform data exchange capability, through the use of common and managed **Extensible Markup Language (XML)/Table Definition Language (TDL) and XML/Exchange Data Language (EDL)** End Device Class models.

2 Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

AEICGL-2010, Smart Grid/AEIC AMI Interoperability Standard Guidelines for ANSI C12.19 / IEEE 1377 / MC12.19 End Device Communications and Supporting Enterprise Devices, Network and Related Accessories, The Association of Edison Illuminating Companies (AEIC), v2.0, 2010.¹

ANSI C12.10-2011, American National Standard for Physical Aspects of Watthour Meters—Safety Standard.²

ANSI C12.19, American National Standard for Utility Industry End Device Data Tables.

ANSI C12.19-1997, American National Standard for Utility Industry End Device Data Tables.

ANSI C12.19-2008, American National Standard for Utility Industry End Device Data Tables.

ANSI X9.31-1998, Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry (rDSA).

AS 4140-1995, Australian Standard, Metering and Utility Information Exchange—Glossary of Terms.³

ECMA 94-1996, 8-Bit Single-Byte Coded Graphic Character Sets—Latin Alphabets No. 1 to No. 4-2nd Edition.⁴

FERC-727-728-2008, Federal Energy Regulatory Commission, Survey on Demand Response, Time-Based Rate Programs/Tariffs and Advanced Metering Infrastructure Glossary, FERC-727 and FERC-728, OMB Control Nos. 1902-0214 & 1902-0213.⁵

IEEE Std 519™-1993, IEEE Recommended Practices and Requirements for Harmonic Control in Electrical Power Systems.^{6, 7}

IEEE Std 754™-2008, IEEE Standard for Binary Floating-Point Arithmetic.

IEEE Std 1159™-1995, IEEE Recommended Practice for Monitoring Electric Power Quality.

IEEE Std 1377™-1998, IEEE Standard for Utility Industry End Device Data Tables.

IEEE Std C57.123™-2010, IEEE Guide for Transformer Loss Measurement.

IEC 62053-23-2003, Electricity Metering Equipment (a.c.)—Particular Requirements—Part 23: Static Meters for Reactive Energy (classes 2 and 3).⁸

¹ AEIC publications are available from The Association of Edison Illuminating Companies (<http://www.aeic.org/>). This document is available from http://www.aeic.org/meter_service/AEICSmartGridStandardv2-11-19-10.pdf.

² ANSI publications are available from the American National Standards Institute (<http://www.ansi.org/>).

³ AS publications are available from Standards Australia (<http://www.standards.org.au/Pages/default.aspx>).

⁴ ECMA publications are available from the European Computer Manufacturers Association (<http://www.ecma-international.org/>).

⁵ FERC publications are available from the Federal Energy Regulatory Commission (<http://www.ferc.gov/>).

⁶ IEEE publications are available from The Institute of Electrical and Electronics Engineers (<http://standards.ieee.org/>).

⁷ The IEEE standards or products referred to in this clause are trademarks of The Institute of Electrical and Electronics Engineers, Inc.

⁸ IEC publications are available from the International Electrotechnical Commission (<http://www.iec.ch/>). IEC publications are also available in the United States from the American National Standards Institute (<http://www.ansi.org/>).

IEC/TR 61000-2-1-1990, Electromagnetic Compatibility (EMC)—Part 2: Environment—Section 1: Description of the Environment—Electromagnetic Environment for Low-frequency Conducted Disturbances and Signalling in Public Power Supply Systems.

ISO 8859-1-1998, Information Technology—8-bit Single-byte Coded Graphic Character Sets—Part 1: Latin Alphabet No 1.⁹

ISO/IEC 646-1991, Information Technology—ISO 7-bit Coded Character Set for Information Interchange.¹⁰

ISO/IEC 7498-1-1994, Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model.

ISO/IEC 10118-1-2000, Information Technology—Security Techniques—Hash-functions—Part 1: General.

ISO/IEC 10118-2-2000, Information Technology—Security Techniques—Hash-functions—Part 2: Hash-Functions Using an n-bit Block Cipher.

ISO/IEC 10646-2011, Information Technology—Universal Multiple-Octet Coded Character Set (UCS).

MC S-EG-02-2010, Measurement Canada Specifications for Approval of Physical Sealing Provisions for Electricity and Gas Meters, S-EG-02 (rev. 1).¹¹

XHTML-2002, XHTML 1.0 The Extensible HyperText Markup Language (Second Edition), W3C Recommendation 26, 2000, revised 2002.¹²

XML-2006, Extensible Markup Language (XML) 1.1 (Second Edition), W3C Recommendation 16.¹³

XMLSchema-2004, XML Schema Part 1: Structures (Second Edition), W3C Recommendation 28.¹⁴

XMLSig-2008, XML Signature Syntax and Processing (Second Edition), W3C Recommendation 10.¹⁵

⁹ ISO publications are available from the ISO Central Secretariat (<http://www.iso.org/>). ISO publications are also available in the United States from the American National Standards Institute (<http://www.ansi.org/>).

¹⁰ ISO/IEC publications are available from the ISO Central Secretariat (<http://www.iso.org/>). ISO publications are also available in the United States from the American National Standards Institute (<http://www.ansi.org/>).

¹¹ MC publications are available from Measurement Canada (<http://www.ic.gc.ca/eic/site/mc-mc.nsf/eng/home>).

¹² World Wide Web (W3) Consortium publications are available from the World Wide Web Consortium (<http://www.w3.org/>). This document is available at <http://www.w3.org/TR/xhtml1>.

¹³ This document is available at <http://www.w3.org/TR/xml11>.

¹⁴ This document is available at <http://www.w3.org/TR/xmlschema-1>.

¹⁵ This document is available at <http://www.w3.org/TR/xmlsig-core>.

3 Definitions

For the purposes of this document, the following terms and definitions apply. The *IEEE Standards Dictionary Online* [B10] should be consulted for terms not defined in this clause.^{16, 17}

The definitions used in this standard have been derived from the sources cited in Clause 2, “Normative references”, with the exception of those introduced in this standard revision, labeled “[New]”.

ACTUAL (ACT): The programmed functional capabilities of an End Device. [ANSI C12.19-1997]

Address: Those inputs whose states select a particular cell or group of cells. [*IEEE Standards Dictionary*]

Advanced Metering Infrastructure (AMI): Defined as the communications hardware and software and associated system and data management software that creates a network between advanced meters and utility business systems and which allows collection and distribution of information to customers and other parties such as competitive retail providers, in addition to providing it to the utility itself. [FERC-727-728-2008]

Alias: An **alternative** name of a Table, Procedure, Element, or a derived type. An alias name can be used as an alternative when making references to Tables, Procedures, Elements, or derived types. Example: Table 0 of ANSI C12.19-1997 contains the Element **MANUFACTURER**, whereas this standard renamed it to **DEVICE_CLASS**, and defined it to be an alias to **MANUFACTURER** for backward compatibility. All references to **MANUFACTURER** and **DEVICE_CLASS** refer to the same Element.

ARRAY: In text form, “Array” or “array,” a grouping of repetitions of the same data type, used to describe a single Element. [New]

BIT FIELD (BFLD): In text form, “Bit Field” or “bit field” is a collection of data subelements, each spanning one or more of bits, that may not be aligned on an Octet boundary nor span an integral multiple of Octets. Multiple occurrences of these subelements are grouped (packed) together so that the entire collection of bits ends on an Octet boundary. For purposes of transmission, the entire group of bits is treated as an Atomic Element having the type of unsigned integer. [New]

Block: A collection of interval-based Register values spanning one or more channels. [New]

Channel: A single path for the collection of interval-based Register values over a period of time. [New]

Clock: A device that generates periodic signals used for synchronization. A device that measures and indicates time. A register whose content changes at regular intervals in such a way as to measure time. [ANSI C12.19-1997]

Count: The number of Octets selected when using the Offset/Octet-count partial Table access method. [ANSI C12.19-1997]

Cumulative Demand: Summation of Maximum Demands during the preceding periods is indicated during the period after the meter has had a demand reset performed and before it has been reset again. Note that the maximum demand for any one demand interval is equal or proportional to the difference between the accumulated readings before and after reset. [*IEEE Standards Dictionary*]

Customer: The purchaser/user of a product or service supplied by a service provider or utility. [AS 4140-1995]

¹⁶ The *IEEE Standards Dictionary Online* subscription is available at http://www.ieee.org/portal/innovate/products/standard/standards_dictionary.html.

¹⁷ The numbers in brackets correspond to those of the bibliography in Annex M.

Data Encryption: The changing of the form of a data stream such that only the intended recipient can read or alter the information and detect unauthorized messages. [ANSI C12.19-1997]

Decade: A functional grouping of Tables by application into groups of ten. The Tables are numbered “X0” through “X9”, with “X” representing the Decade number. [ANSI C12.19-1997]

Default Set: Element-values provided when FLC Tables or FLC+1 Tables of any Decade are not used by the End Device. This standard provides Default Set Element-values for all FLC and FLC+1 Table-elements. End Device vendors may provide Default Set Element-values for any Manufacturer-defined Decades FLC and FLC+1 using EDL. [New]

Demand: The consumption (e.g., energy and volume) recorded during the integration time period divided by the integration time period. [ANSI C12.19-1997]

Demand, Block: An average value (e.g., Watthours/hours) occurring over a Demand period specified by the End Device. This value may be used by the End Device for maximum or minimum registration computation. [ANSI C12.19-1997]

Demand, Maximum: The highest Demand measured over a selected period of time. Also referred to as “Peak Demand.” [ANSI C12.19-1997]

Demand, Minimum: The lowest Demand measured over a selected period of time. [ANSI C12.19-1997]

Demand, Present: Demand occurring during the present Demand interval or subinterval (e.g., Watts or VA). [ANSI C12.19-1997]

Demand, Sliding Window (Rolling-Interval): The Block Demand calculated over an integration period that includes subintervals of previous Demand calculations. [ANSI C12.19-1997]

Deprecated: Outdated or superseded. Deprecated syntax. Tables or Elements that are defined in the standard or in the Manufacturer reference documentation of the End Device (TDL) and are clearly marked as deprecated in the document.

DIMENSION (DIM): The maximum functional capability designed into an End Device. (See the discussion in 4.1.1, “Standard Tables grouping”, on the Function Limiting Control (FLC) Table.) [ANSI C12.19-1997]

Document Type Definition (DTD): The XML document type declaration contains or points to markup declarations that provide a grammar for a class of documents. This grammar is known as a document type definition or DTD. The document type declaration can point to an external subset (a special kind of external entity) containing markup declarations, can contain the markup declarations directly in an internal subset, or can do both. The DTD for a document consists of both subsets taken together. [Reprinted from W3C Recommendation, *Extensible Markup Language (XML) 1.1 (Second Edition)*¹⁸ © 2006 W3C, XML-2006]

Element: The union of all Atomic Elements that share the same index prefix. An Element can be any type, a set, an array, or a selection from an array. [ANSI C12.19-1997]

Element, Atomic: A restricted subset of an Element that is the smallest component of an Element that can be transmitted as an integral number of Octets without loss of its meaning or interpretation during transmission, in accordance with Octet ordering and bit packing. [New]

Element, Final: An innermost expression of an Atomic Element or a Sub-element. This expression can be used to retrieve a single value. Only Final Elements can be used in calculations. Decade names, Table names, Procedure

¹⁸ This document is available at <http://www.w3.org/TR/xml11>.

names, and constants are also Final Elements, when used in a calculation. The value returned by Decade, Table, and Procedure names is their corresponding number. The value returned by a constant is the constant's assigned value. [New]

Element, Volatile: A Table Element whose values may change asynchronously by means other than reading or writing Tables. Two consecutive reads of the same Table Element may yield different values due to real-time changes to the End Device Register, sensed values, state, or status. [New]

Element-count: The number of Elements selected when using the Index/Element-count partial Table access method. The Elements selected are derived from the serialization of all Elements, starting with the one identified by the index, sequentially, which share part or all of the selection index prefix. [ANSI C12.19-1997]

End Device: The closest device to the sensor or control point within a metering application communication system **that** is compliant with the Utility Industry End Device Data Tables. [ANSI C12.19-1997]

End Device Agent: The entity responsible for managing the End Device operations and programs. It is capable of affecting change to the End Device program. It is also responsible for keeping a record of all End Device programming parameters, constants, change history, and event logger records. The End Device Agent is also responsible for maintaining and demonstrating continuity between metrological data records and End Device programs, exclusively through the use of Utility Industry End Device Data Tables. [ANSI C12.19-1997]

Event: A noteworthy occurrence that has a location in time and space, and does not have duration. Event parameters include information about the state of the End Device before the Event, after the Event, the Event time, and related Event-identifying information. Event parameters can be made available for inspection directly or indirectly through a calculable derivative. [Adopted from ANSI C12.19-1997 and *The Unified Modeling Language Reference Manual* [B11]]

Event Instance: A specific occurrence of an Event. Each Event Instance is characterized by a set of Event parameters. There are four kinds of Event Instances: Call Event, Change Event, Signal Event, and Time Event. [Adopted from ANSI C12.19-1997 and *The Unified Modeling Language Reference Manual* [B11]]

Event Logger Data: The entire content of all Tables designated as Program Tables, all associated pointers, and all signatures that are expressed by or implied by Event Logger Record for a specific Event. The Event Logger Data originates initially inside the End Device, but it may extend into the End Device Agent record-keeping system. [ANSI C12.19-1997]

Event Logger Record: The pointers and signatures that are expressed by or implied by **EVENT_LOG_DATA_TBL.EVENT_LOG_RCD** and **EVENT_LOG_DATA_TBL.EVENT_ENTRY_RCD** to define the Event Logger Data for a specific Event. [New]

Event, Call: An Event Instance caused by receipt of an explicit synchronous request (e.g., a request to write a Table by a communication protocol). [Adopted from *The Unified Modeling Language Reference Manual* [B11]]

Event, Change: An Event Instance caused by change in the value of a Boolean expression (e.g., tamper detect **and** change in metering mode). [Adopted from *The Unified Modeling Language Reference Manual* [B11]]

Event, Signal: An Event Instance caused by explicitly named asynchronous communication among Objects (e.g., advancing list pointers as a result of execution of an advance list pointers procedure). [Adopted from *The Unified Modeling Language Reference Manual* [B11]]

Event, Time: An Event Instance caused by the occurrence of an absolute time or the passage of a relative amount of time (e.g., time-of-use schedule invoking a self-read). [Adopted from *The Unified Modeling Language Reference Manual* [B11]]

Exchange Data Language (EDL): For encoding, import, and export of Default Sets defined by this standard, Table Element values retrieved from an End Device, Table Element values to be stored in an End Device, Manufacturer constant Element-values, and Utility or Customer site-specific constant Element-values that need to be known by application. The Exchange Data Language Element-values are encoded using XML. [New]

Extensible Markup Language (XML): A recommendation for a lexical grammar produced by the World Wide Web Consortium (W3C). XML describes a class of data objects called XML documents and partially describes the behavior of computer programs that process them. XML is an application profile or restricted form of SGML, the Standard Generalized Markup Language [ISO 8879]. By construction, XML documents are conforming SGML documents. [Reprinted from W3C Recommendation, *Extensible Markup Language (XML) 1.1 (Second Edition)* © 2006 W3C, XML-2006]

Index: A unique sequence of numbers used to select a Table Element. An Index may be represented in dot-notation format (text transmission form) or as in binary array (binary transmission form). An Index may reference any Table Element. [ANSI C12.19-1997]

Interharmonics: Frequencies between the harmonics of the power frequency voltage and current that can be observed but are not an integer multiple of the fundamental frequency. They can appear as discrete frequencies or as a wide-band spectrum. [Adapted from IEC/TR 61000-2-1-1990 and the *IEEE Standards Dictionary*]

Load Profile: The recording, storage, and analysis of consumption data over a period of time for a particular installation. [ANSI C12.19-1997]

Meter: A device that measures and records the consumption or usage of the product/service. [ANSI C12.19-1997]

Metrological Adjustment: Any physical means or methods designed or used, whether directly or indirectly, whether intentionally or unintentionally, to alter or correct the measurement characteristics of an End Device or any of its subsystems that are used for trade measurement purposes. This includes altering or replacing Registers, components, connections, or internal working parts of the End Device. It also includes the alteration of any Metrological Parameter that would be made available from the End Device or any of its outputs. Metrological Adjustments manifest themselves in changes to Metrological Tables. [Adopted from MC S-EG-06:2011 [B7] and *Utility Industry Standard Tables User's Guide* [B8]]

Mode, Factory: A superset of Meter Shop Mode where, in addition, manufacturer limited functions and parameters are also available for modification. [New]

Mode, Meter Shop: The operating state of the End Device where revenue metering is prohibited. The entry state and the exit state of the End Device may be unrelated and untraceable. An exit from Meter Shop Mode represents a new initial state for the End Device. While operating in Meter Shop Mode, all operational safeguards such as logging, security, and accumulation may be disabled. [New]

Mode, Metering: The operating state of the End Device wherein measured or input quantities are accumulated for the purpose of revenue metering. [New]

Mode, Test: The operating state of the End Device wherein the Metering Mode state parameters are saved prior to entry and restored upon exit. [New, Adopted from ANSI C12.1-2001]

Object: A Table Object or an Element Object. [Revised]

Object, Element: An Atomic Element, an Element, a Sub-element, or a Final Element of a Table that can be described, encoded, or communicated using this standard. [New]

Object, Table: A Table and all the Elements it contains that can be described, encoded, or communicated according to this standard. When the Table referenced is a Pending Table, then the Table Object includes the Pending Header. [New]

Octet: A sequence of eight bits. [ANSI C12.19-1997]

Offset: The number of Octets, relative to the beginning of a Table, that precede the desired Element. When the Table is a Pending Table, then Offset zero (0) is the first Element at **the** beginning of the Table following the Pending Table header. [ANSI C12.19-1997]

Override: In the context of declaration of type, a relaxed variation of the Replace attribute in that Overridden types have different Offsets and Indices from those of the referenced Elements, and thus, **they** can expose additional details to **the** internal structure of Elements that were otherwise opaque. [New]

Packed Record: In text form, “Packed Record,” or any variation thereof, a group of Element types, *anyElementType*, created as a new derived data type. [New]

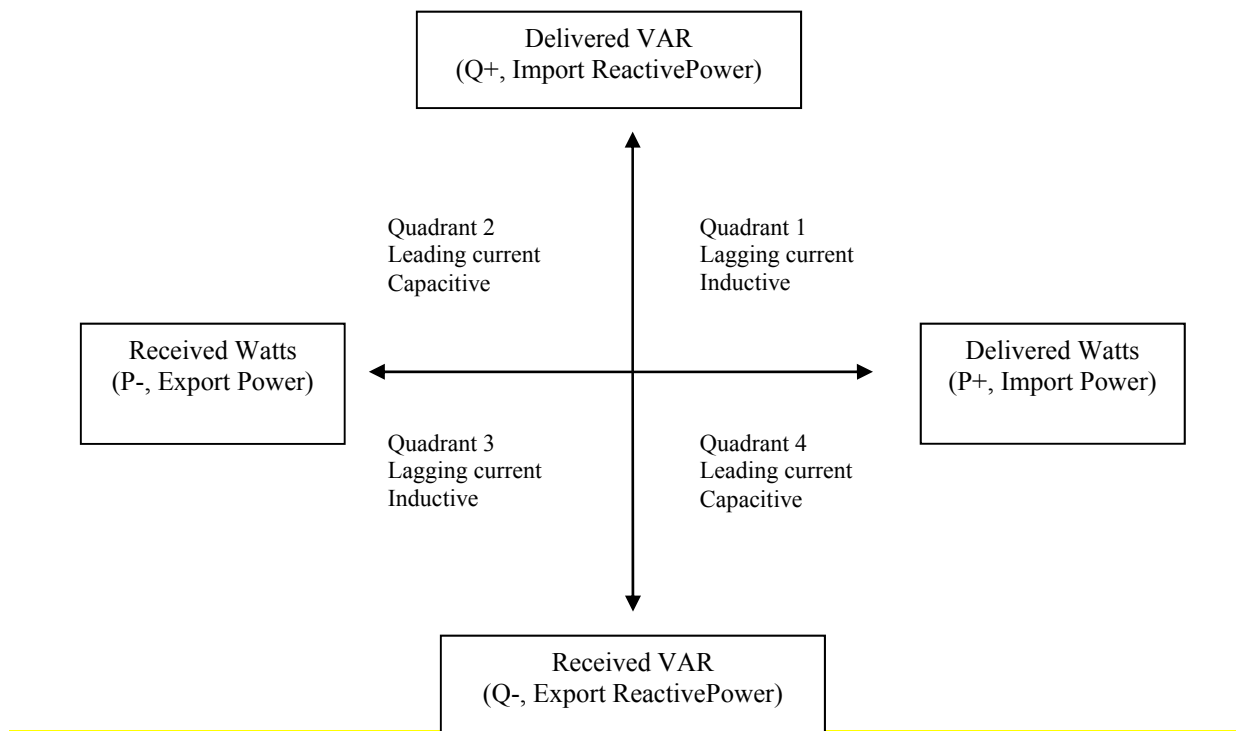
Procedure: An action that can be performed by the End Device. A Procedure is initiated by writing to Standard Table 07, and the result may be retrieved by reading Standard Table 08. [New]

Procedure, Manufacturer: A Procedure whose structure and actions are defined by an individual End Device manufacturer. [New]

Procedure, Metrological: A Procedure whose action causes a change to a Metrological Parameter or results in a Metrological Adjustment. [New]

Procedure, Standard: A Procedure whose structure and actions are defined by this standard. [New]

Quadrant: Divisions of the plane formed by the representation of active power (watts), reactive power (VAR), and vector apparent power (voltamperes) in a Cartesian coordinate system. In this standard, the convention used is that the vectors rotate counter clockwise and the quadrants are numbered as shown in the diagram below. For clarity, the diagram includes both the *Handbook* definition and the IEC terminology in parentheses. It should be noted that the ANSI terms are with respect to the utility (e.g., “delivered” means delivered by the utility) and the IEC 62053 terms are with respect to the consumer (e.g., “imported” means imported by the consumer). [Adapted from the *Handbook for Electricity Metering* [B3], IEC 62053-23-2003, and the *CRC Standard Mathematical Tables and Formulae* [B1]]



Redefine: Used in the context of declaration of type. The new type definition is constrained by the referenced type definition whereby the new definition shall have identical Element content, types, Indices, and Offsets to those found in the referenced type. The referenced type is left unaffected. This feature is useful when defining **ACT** Tables, then reusing only the Element types (not the Element names or descriptions) of the **DIM** Tables to define actual limits Tables. The local redefinition feature is subject to the following constraints:

- A type can be Redefined any number of times. Redefinition of a type that has already been previously redefined elsewhere is possible.
- New types based on Redefined types can be defined in the same **Table Definition Language (TDL)** document that defines the referenced type.
- Redefinition of a Replaced packed record is possible.
- Redefinitions are processed after Replacements and Overrides.

Register: A device for use with a meter that indicates or records units of the delivered commodity or other quantity measured. Registers in End Devices are often spoken of according to the units involved, such as accumulation (energy, volume, mass), Demand, Time-of-use (TOU), instantaneous values, or other(s) as appropriate. [New, portions adopted from ANSI C12.1-2001 and ANSI C12.19-1997]

Register, Secured: A Volatile Element whose transmitted value can be encoded by the End Device as a one-way hash in response to a Read Service request from an initiator. The purpose of the encoding is to detect whether the value transmitted was subsequently altered and to enable the receiver to verify the authenticity of the value in a manner that is independent from the transport protocol used. The Reading of a Secured Register may generate a Read Secured Register or Read Secured Table event within the End Device. [New]

Replace: The action whereby an older type definition shall be replaced by (i.e., mapped to) a new type definition. Any reference to the old definition shall resolve to the new definition. This feature enables one standard to be derived from another standard and for a manufacturer of End Devices to define (thus document) opaque fields that

are defined by the referenced standard and implemented in the End Device. The replacement feature is subject to the following constraints:

- A referenced type can be Replaced only once in any one instance of an End Device **Table Definition Language (TDL)** definition; therefore, Replacement of a type that has already been previously replaced elsewhere is not possible.
- The Replaced type cannot be defined in the same TDL document that defines the original type.
- Replaced types have identical Element Offsets and Indices as those of the referenced Elements; however, they can expose additional **details** to **the** internal structure of **Elements** that were otherwise opaque.
- A Replaced type cannot be Overridden in the same TDL document.

[New]

Reserved: Qualifier applied to one or more Elements or values indicating they are not available for use with this standard. The use of a Reserved Element or value shall render an application noncompliant with this standard. [New]

Seal: An item or process that has the purpose of providing marketplace integrity and confidence against End Device tampering and as such it has three main functions: (1) **to** secure an End Device so that access or changes to Metrological Adjustments and Sealable Parameters will be detectable; (2) to identify the date and organization **that** verified the End Device (this information is necessary for the administration and enforcement of legislation, for complaint or dispute investigation purposes, and for seal period and seal extension purposes); **and** (3) to act as a deterrent to the unauthorized access or changing of metrologically sensitive adjustments and Sealable Parameters. Seals may be implemented as Physical Seals and/or **as** Logical Seals. [Adopted from **MC S-EG-02-2010**]

Seal, Logical (Event Loggable): A Seal that **comprises** firmware and software mechanisms that work together to secure access to an End Device's Metrological Adjustments and Sealable Parameters. This standard provides Event Log Tables for the management of Seals. [New]

Seal, Physical: A physical mechanism that is used to secure access to an End Device's Metrological Adjustments and Sealable Parameters. [New]

Sealable Parameter: A Metrological Parameter that is secured or protected from alteration by a Seal. [New]

Season: A calendar-specified period used for activation of rate schedules. [ANSI C12.19-1997]

Secured Read: Reading of a Secured Register from an End Device that is capable of generating an End Device Secured Register event, provided that this capability was enabled in the End Device. [New]

Self-read: A copy of Elements from one Table to another Table inside of an End Device, often at periodic intervals, triggered by an Event. [New]

SET: A collection of Booleans, eight Booleans per Octet. Each Boolean member of the SET is a Final Element. [New]

Signature: Often referred to as a hash-code; an n-bit sequence that is generated using an algorithm on the content of an End Device Table or a group of End Device Tables. The resulting sequence of bits is truncated to a small number of bits (e.g., 128 bits). A good hash function will enable (1) detection of file (or table) content corruption (unauthorized change) and (2) authentication of the End Device Agent who created the file (or Table). [ANSI C12.19-1997]

Strict: In the context of declaration of type. When a type is created and the Replace, Redefine, or Override attribute is used and strictness is asserted (TRUE), then the search rule for the target type shall be strictly applied. A strict search requires that the implied or explicit target scope contains the Replaced, Redefined, or Overridden type. A nonstrict search requires that the implied or explicit target scope is where the search starts in the End Device Table Definition Language (TDL) document scope hierarchy. This search is carried out in such a manner that the first encounter of the target type within any of the parent scopes will be considered a match for the sought Replaced or Redefined type. [New]

STRING: An Atomic Array of Octets to be interpreted as characters dependent on CHAR_FORMAT. See 6.1, “Character set”. [New]

Sub-element: A subset of an Atomic Element, which is a single bit of a SET or a member of a BIT FIELD. [ANSI C12.19-1997]

Table: Functionally related utility application data Elements, grouped together into a single data structure for transport. [ANSI C12.19-1997]

Table Definition Language (TDL): Syntax used for detailing the standard’s or an End Device’s Table structure, defined types, and constraints in a machine-readable format. The Table Definition Language is encoded using Extensible Markup Language (XML). [New]

Table, Extended User-defined (EUDT): An extension on UDTs to add up to 2040 virtual Tables that can select from, or reference into, Standard Tables and Manufacturer Tables. Extended User-defined Tables shall not refer to, or select from, Extended User-defined Tables. [New]

Table, Function Limiting Control (FLC): The first Table in each Decade. This Table specifies the limits designed into an End Device with respect to Elements used within that Decade. [ANSI C12.19-1997]

Table, Manufacturer: A Table whose structure is defined by an individual End Device manufacturer. [New]

Table, Metrological: A Table that contains at least one Element that is also a Metrological Parameter. [New]

Table, Pending: Tables that have yet to be activated through conditions specified by the EVENT_CODE and trigger conditions in the EVENT_STORAGE_RCD in the Pending Event Description of the End Device. [New]

Table, Program: An End Device Table that contains setup, configuration, or general product or record-keeping information about the End Device. Programming Tables shall not contain sensory information, End Device state information, or End Device logs that are volatile and change in the normal course of operation of the End Device. [New]

Table, Secured: A Table that contains Secured Register. [New]

Table, Standard: A Table whose structure is defined by this standard. [New]

Table, User-defined (UDT): A Table whose Elements values that are derived from references (for writing) or Element values (for reading) into a temporary common area, which enables applications to make references to Tables and Table Elements that are present or implied to be present in an End Device. The Elements gathered may then be retrieved, and the Elements referenced may be written, as if the standard provided an explicit Table and descriptive syntax to express the resulting User-defined Table structure. [New]

Tariff: A published list of rate schedules and terms and conditions. [ANSI C12.19-1997]

Tier: A consumption-based or time-based rate range for the commodity. [New]

Time-Of-Use (TOU) Metering: The use of metering equipment to separately record metered or measured quantities according to a time schedule. [ANSI C12.19-1997]

Undefined: Qualifier applied to one or more Elements or values that are not defined by this standard. The use of an Undefined Element or value shall not affect compliance with this standard; however, the outcome is indeterminate. [New]

Utility: A provider of electricity, gas, water, telecommunications, or related services to a community. [ANSI C12.19-1997]

4 General

4.1 Standard Tables

Standard Tables are those whose structures are specified by this standard. They should be used for both End Device programming and reading. The Tables provide control Tables as well as data Tables for a wide variety of functions to be implemented in addition to those presently defined. This standard provides for a total of 2040 Standard Tables, although not all are defined.

4.1.1 Standard Tables grouping

The Standard Tables are grouped together in Decades, with each Decade covering a general area of functionality. The first Table of each Decade, beginning with Decade 1, is referred to as a Function Limiting Control (FLC) Table. The purpose of the FLC Table is to specify the designed DIMENSION (DIM) limits (maximum limits) for Elements used within the Decade for the End Device.

If present, the Table immediately following an FLC table is called the FLC+1 Table. The FLC+1 Table defines the ACTUAL (ACT) limits used for Elements in the current configuration of the End Device.

If the FLC+1 Table is not present, the ACT limits are identical to the designed DIM limits specified in the FLC Table.

If the FLC and FLC+1 Tables are not used in a Decade, then the limits are defined by a default set of values. The Default Sets for a particular Decade are defined within this standard in Annex C, “(normative) Default Sets for Decade Tables”. Possible combinations of FLC, FLC+1, and Decade Tables are defined in Figure 4-1.

	FLC Table	FLC+1 Table	FLC+n Table	Where ACT Values Are Found
A	Not Used	Not Used	Not Used	None - Decade not used
B	Not Used	Not Used	Used	Default set, per Standard
C	Not Used	Used	Not Used	Decade FLC+1 Table**
D	Not Used	Used	Used	Decade FLC+1 Table
E	Used	Not Used	Not Used	Decade FLC Table**
F	Used	Not Used	Used	Decade FLC Table
G	Used	Used	Not Used	Decade FLC+1 Table**
H	Used	Used	Used	Decade FLC+1 Table

** Not recommended practice.

Figure 4-1—Possible combinations of FLC, FLC+1, and Decade Tables

4.1.2 Standard Tables properties

All Standard Tables assume a set of properties that are defined by this standard. Listed below are the Tables’ default properties to be assumed when not explicitly qualified by this standard. The default Table properties are shown in Figure 4-2.

Global Default Table Property Value

Table Property*	Required	FLC Table	FLC+1 Table	FLC+n Table (n>1)
name	Yes			
number	Yes			
label	No	<i>Table name</i>	<i>Table name</i>	<i>Table name</i>
type	Yes			
role	No	“LIMITING”	“ACTUAL”	“UNDEFINED”
associate	No	Scope of the Decade where defined	Scope of the Decade where defined	Scope of the Decade where defined
atomic	No	“false”	“false”	“false”
accessibility	No	“READONLY”	“READWRITE”	“READWRITE”
deprecated	No	“false”	“false”	“false”
metrological	No	“false”	“false”	“false”
volatile**	No	“AUTO_AS_PER_ROLE”	“AUTO_AS_PER_ROLE”	“AUTO_AS_PER_ROLE”
class***	No	“STD”	“STD”	“STD”

* For more information, see I.2.25, “<procedure> Element (Child of <tdl> or <decade>)”.

** When the volatile attribute is set to “AUTO_AS_PER_ROLE”, then the values shall be assumed for Table volatility based on their assumed role as follows:

role = “UNDEFINED” implies volatility is “true”.

role = “LIMITING” implies volatility is “false”.

role = “ACTUAL” implies volatility is “false”.

role = “CONTROL” implies volatility is “false”.

role = “DATA” implies volatility is “true”.

role = “PRIVATE” implies volatility is “true”.

*** A tag that identifies an application run-time class (module) that should be used to manage this Table. Standard classes defined are STD.UDT, STD.EUDT, and STD.

Figure 4-2—Default Standard Tables properties

4.1.3 Standard Procedure properties

All Standard Procedures (Table 7, Table 8, and procedures defined within) assume a set of properties that are defined by this standard. Listed below are the default properties to be assumed when not explicitly qualified by this standard. The default Procedure properties are shown in Figure 4-3.

Procedure Property*	Required	Global Default Procedure Property Value	
		Table 7 / Procedure Request	Table 8 / Procedure Response
name	Yes		
number	Yes		
label	No	TABLE name / Procedure Name	TABLE name / Procedure Name
type	Yes		
role	No	“CONTROL”	“DATA”
associate	No	Scope of Procedure where defined then PROC_INITIATE_TBL. STD_PARM_RCD then its parents’ scope	Scope of Procedure where defined then PROC_RESPONSE_TBL. STD_RESP_RCD then its parents’ scope
atomic	No	“true”	“true”
accessibility	No	“READWRITE”	“READONLY”
deprecated	No	“false”	“false”
metrological	No	“false”	“false”
volatile	No	“AUTO_AS_PER_ROLE”	“AUTO_AS_PER_ROLE”
class	No	“STD”	“STD”
target	Yes	Not Applicable / “PROC_INITIATE_TBL. STD_PARM_RCD”	Not Applicable / “PROC_RESPONSE_TBL. STD_RESP_RCD”
strict	No	Not applicable / “true”	Not applicable / “true”

* For more information see I.2.25, “<procedure> Element (Child of <tdl> or <decade>)”.

Figure 4-3—Default Standard Tables 7, 8, and Procedures properties

4.2 Manufacturer Tables

Manufacturer Tables are those structures specified by individual End Device vendors. They should be used to introduce innovations or to provide customer-requested data structures that are not defined by this standard. It is noted that the standard also provides for the latter to be accomplished in a limited manner through user-defined Tables (Decade 8) and extended user-defined Tables (Decade 14). Manufacturer Tables also provide a path for new functions to evolve into a Standard Tables.

4.2.1 Manufacturer Tables grouping

This standard provides access for a total of 2040 Manufacturer Tables. When Manufacturer Tables may introduce new Decades, then the implementation guidance enumerated in Figure 4-1 shall be followed. When Manufacturer Tables are introduced singly (i.e., as FLC+n Tables), they shall be associated with an existing Decade. More details are found in I.2, “TDL XML File Format of the TDL Document”, and more specifically see the associate attribute described in I.2.25, “<procedure> Element (Child of <tdl> or <decade>)”.

4.2.2 Manufacturer Tables properties

All Manufacturer Tables assume a set of properties that are defined by this standard. Listed below are the Tables’ default properties to be assumed when not explicitly qualified by the manufacturer’s TDL. The default Manufacturer Table properties are shown in Figure 4-4.

Mfg. Table Property*	Required	Global Default Table Property Value
name	Yes	
number	Yes	
label	No	Table name
type	Yes	
role	No	“UNDEFINED”
associate	No	Scope of the TDL or Decade where defined
atomic	No	“false”
accessibility	No	“READWRITE”
deprecated	No	“false”
metrological	No	“false”
volatile	No	“AUTO_AS_PER_ROLE”
class*	No	“STD”

*A tag that identifies an application run-time class (module) that should be used to manage this Manufacturer Table. STD signifies that the Standards Table-read and Table-write and TDL rules apply, with no additional assumptions for interpretation. Other predefined algorithms are STD.UDT and STD.EUDT.

Figure 4-4—Default Manufacturer Tables properties

4.2.3 Manufacturer Procedure properties

All Manufacturer Procedures share, by default, a set of properties that are defined by this standard. Listed below are the default properties to be assumed when not explicitly qualified by the Manufacturer’s TDL. The default Procedure properties are shown in Figure 4-5.

Procedure Property	Required	Global Default Procedure Property Value	
		MFG Procedure Request	MFG Procedure Response
name	Yes		
number	Yes		
label	No	Procedure Name	Procedure Name
type	Yes		
role	No	“CONTROL”	“DATA”
associate	No	Scope of Procedure where defined then PROC_INITIATE_TBL. MFG_PARM_RCD then its parents’ scope	Scope of Procedure where defined then PROC_RESPONSE_TBL. MFG_RESP_RCD then its parents’ scope
atomic	No	“true”	“true”
accessibility	No	“READWRITE”	“READWRITE”
deprecated	No	“false”	“false”
metrological	No	“false”	“false”
volatile	No	“AUTO_AS_PER_ROLE”	“AUTO_AS_PER_ROLE”
class	No	“STD”	“STD”
target	Yes	“PROC_INITIATE_TBL. MFG_PARM_RCD”	“PROC_RESPONSE_TBL. MFG_RESP_RCD”
strict	No	“true”	“true”

Figure 4-5—Default Manufacturer Procedures properties

4.3 Packed Record, Bit Field, and Element properties

Packed Records, Bit Fields, and the contained Element definitions also have default properties. These default properties are fully described in Annex I, “(normative) XML file format of TDL and EDL files”.

4.4 Extended User-defined Tables properties

Extended User-defined Tables default properties are identical to the properties shown in 4.1.2, “Standard Tables properties”, except for the class=“STD.EUDT”. The class “STD.EUDT” invokes the standard’s processing requirements for Extended User-defined Tables.

5 Syntax

5.1 Descriptive syntax

Describing data definitions is usually accomplished within the confines of a given language and the grammar rules of that language. Since the data definitions embodied within this document are meant to be independent of a specific language and capable of being implemented within the confines of any language, a method for describing the data definitions has been adopted. The descriptive syntax is expressed using the Extended BNF definitions defined in Annex G, “(normative) Document-form descriptive syntax”. The definition syntax is a merge between the traditional Backus Naur Format and XML simple BNF as described in XML-2006, with extensions that facilitate the binding of production rules needed for the definition of the published pseudo PASCAL text (Clause 9, “Tables”) and the machine-readable XML based TDL. In addition, the Extended BNF syntax provides definitions for constraints on binary transmission and interpretation of types.

6 Special data types

In order to add flexibility to the definition of some of the data types used within the tables, special data types are provided. The selection of special data type definitions are placed in **GEN_CONFIG_TBL** (Table 00) and remain in effect for all Tables. This clause defines the encoding used to transmit the special data type. All special data types, just like built-in types, are Atomic.

6.1 Character set selection

This selection is used to determine the encoding of characters that are used in the Tables. **GEN_CONFIG_TBL.FORMAT_CONTROL_1.CHAR_FORMAT** is the controlling selector for the encoding of **CHAR**, **STRING**, **FLOAT_CHAR6**, **FLOAT_CHAR12**, and **FLOAT_CHAR21**.

TDL Type Definitions

```
{ Enumerator CHAR_FORMAT_ENUM }
```

TDL Element Descriptions

Identifier	Value	Definition
CHAR_FORMAT_ENUM	0	Reserved.
	1	ISO 7-bit coded character set for information interchange, per ISO/IEC 646-1991, a fixed-width encoding using 8 bits as its base unit for length calculations and UINT8 as the base unit for transmission.
	2	ISO 8-bit coded character as per ISO 8859-1-1998 or ECMA 94-1996 Latin 1 character set, a fixed-width encoding using 8 bits as its base unit for length calculations and UINT8 as the base unit for transmission.
	3	UTF-8 as per BNF.utf8 ., variable-width encoding using 8 bits as its base unit for length calculations and UINT8 as the base unit for transmission. This type maximizes compatibility with CHAR_FORMATs 1 and 2 as per Section 2.5, “Encoding Forms / UTF-8” and Section 15.9, “Specials” of The Unicode Standard [B13]. In UTF-8, the byte order mark, BOM, corresponds to the byte sequence [#xEF] [#xBB] [#xBF]. The BOM of UTF-8 shall be permitted only when it can be accommodated by the size of the element [at least three (3) Octets], but it shall be ignored (not produce displayable text) by the processing application.
	4..7	Reserved.

6.2 Noninteger formats

This selection is used to determine the structure of noninteger numbers used in the Tables. The Table Elements **GEN_CONFIG_TBL.FORMAT_CONTROL_3.NI_FORMAT1** and **GEN_CONFIG_TBL.FORMAT_CONTROL_3.NI_FORMAT2** are the controlling selectors for the built-in types **NI_FMAT1** and

NI_FMAT2. They provide the final associations between **BNF.anyFloat** noninteger data types and the actual atomic encoding used to transmit **NI_FMAT1** and **NI_FMAT2** data items as defined below.

TDL Type Definitions

{ *Enumerator* **NI_FORMAT_ENUM** }

TDL Element Descriptions

Identifier	Value	Definition
NI_FORMAT_ENUM		Values that may be assumed by GEN_CONFIG_TBL.FORMAT_CONTROL_3.NI_FORMAT1 (which controls the interpretation of the built-in type NI_FMAT1) and GEN_CONFIG_TBL.FORMAT_CONTROL_3.NI_FORMAT2 (which controls the interpretation of the built-in type NI_FMAT2).
	0	FLOAT64
	1	FLOAT32
	2	FLOAT_CHAR12 (A STRING Number)
	3	FLOAT_CHAR6 (A STRING Number)
	4	INT32 (Implied decimal point between fourth and fifth digits from least significant digit. For example, 0.0001 is represented as 1)
	5	FIXED_BCD6
	6	FIXED_BCD4
	7	INT24
	8	INT32
	9	INT40
	10	INT48
	11	INT64
	12	FIXED_BCD8
	13	FLOAT_CHAR21 (A STRING Number)
	14..15	Reserved

6.2.1 STRING numbers

When **STRING** is used and **GEN_CONFIG_TBL.FORMAT_CONTROL_3.NI_FORMAT1** or **GEN_CONFIG_TBL.FORMAT_CONTROL_3.NI_FORMAT2** has a value of 2, 3, or 13, the number shall be represented according to **BNF.floatingPointStringNumber**.

A **STRING** number (**BNF.floatingPointStringNumber**) representation can have any number of leading spaces, followed by an optional plus or minus sign, followed by a mandatory one or more digits, followed by an optional period and zero or more digits, followed by an optional exponent. An exponent is composed of the letter “E”, “e”; followed by an optional plus or minus sign, followed by a mandatory one or more digits. Any number of spaces can lead or follow the **BNF.floatingPointStringNumber**, but spaces shall not be embedded within. This format ensures that clear identification of any **BNF.floatingPointStringNumber** is possible in a stream (sequence) of **BNF.floatingPointStringNumber**⁺, using the space character as a separator.

Examples:

Valid production based on **BNF.floatingPointStringNumber**
‘+1.0E-7’
‘123.6478e+03’
‘1.2345’

‘1.0E-3’

Invalid production based on **BNF** *floatingPointStringNumber*

‘.5’

‘e+03’

‘e3’

‘-1.e3’

6.3 Date and time formats

This selection is used to determine the structure of dates and times used in the tables. The Element **GEN_CONFIG_TBL.FORMAT_CONTROL_2.TM_FORMAT** is the controlling selector for **HTIME_DATE**, **LTIME_DATE**, **STIME_DATE**, **TIME**, **STIME**, and **HTIME**.

TDL Type Definitions

```
{ Enumerator TM_FORMAT_ENUM }
```

TDL Element Descriptions

Identifier	Value	Definition
TM_FORMAT_ENUM		Values that may be assumed by GEN_CONFIG_TBL.FORMAT_CONTROL_2.TM_FORMAT (which controls the interpretation of the built-in types HTIME_DATE, LTIME_DATE, STIME_DATE, TIME, STIME, and HTIME).
	0	No clock in the End Device.
	1	BCD type with discrete fields for year, month, day, hour, minute seconds, and fractional seconds.
	2	UINT8 type with discrete fields for year, month, day, hour, minute seconds, and fractional seconds.
	3	UINT32 counters where HTIME_DATE , LTIME_DATE , and STIME_DATE types are encoded relative to 01/01/1970 @ 00:00:00 UTC, with discrete fields for minutes and fraction of a minute.
	4	UINT32 counters where HTIME_DATE , LTIME_DATE , and STIME_DATE types are encoded relative to 01/01/1970 @ 00:00:00 UTC), with discrete fields for seconds and fraction of a second.
	5..7	Reserved.

6.3.1 HTIME_DATE, LTIME_DATE, STIME_DATE, TIME, STIME, HTIME types

The date and time structures are described next as packed records, but **they** are transmitted as Atomic Table Elements.

TDL Type Definitions

```
TYPE HTIME_DATE_RCD = PACKED RECORD
  SWITCH GEN_CONFIG_TBL.TM_FORMAT OF
    CASE 0 :   NIL           : NIL;
    CASE 1 :   YEAR          : BCD;
              MONTH         : BCD;
```

```

        DAY                : BCD;
        HOUR               : BCD;
        MINUTE             : BCD;
        SECOND             : BCD;
        BCD_SEC_FRACTION   : BCD(3);
CASE 2 :   YEAR            : UINT8;
          MONTH           : UINT8;
          DAY             : UINT8;
          HOUR            : UINT8;
          MINUTE          : UINT8;
          SECOND          : UINT8;
          UINT_SEC_FRACTION : UINT24;
CASE 3 :   U_TIME         : UINT32;
          SECOND          : UINT8;
          UINT_SEC_FRACTION : UINT24;
CASE 4 :   U_TIME_SEC     : UINT32;
          U_SEC_FRACTION  : UINT32;
END;
END;

TYPE LTIME_DATE_RCD = PACKED RECORD
  SWITCH GEN_CONFIG_TBL.TM_FORMAT OF
    CASE 0 :   NIL                : NIL;
    CASE 1 :   YEAR                : BCD;
              MONTH               : BCD;
              DAY                 : BCD;
              HOUR                : BCD;
              MINUTE              : BCD;
              SECOND              : BCD;
    CASE 2 :   YEAR                : UINT8;
              MONTH               : UINT8;
              DAY                 : UINT8;
              HOUR                : UINT8;
              MINUTE              : UINT8;
              SECOND              : UINT8;
    CASE 3 :   U_TIME              : UINT32;
              SECOND              : UINT8;
    CASE 4 :   U_TIME_SEC          : UINT32;
END;
END;

TYPE STIME_DATE_RCD = PACKED RECORD
  SWITCH GEN_CONFIG_TBL.TM_FORMAT OF
    CASE 0 :   NIL                : NIL;
    CASE 1 :   YEAR                : BCD;
              MONTH               : BCD;
              DAY                 : BCD;
              HOUR                : BCD;
              MINUTE              : BCD;
    CASE 2 :   YEAR                : UINT8;
              MONTH               : UINT8;
              DAY                 : UINT8;
              HOUR                : UINT8;
              MINUTE              : UINT8;
    CASE 3 :   U_TIME              : UINT32;
```

```

CASE 4 :    U_TIME_SEC                : UINT32;
END;
END;

```

```

TYPE HTIME_RCD = PACKED RECORD
  SWITCH GEN_CONFIG_TBL.TM_FORMAT OF
    CASE 0 :    NIL                    : NIL;
    CASE 1 :    HOUR                    : BCD;
               MINUTE                    : BCD;
               SECOND                    : BCD;
               BCD_SEC_FRACTION          : BCD(3);
    CASE 2 :    HOUR                    : UINT8;
               MINUTE                    : UINT8;
               SECOND                    : UINT8;
               UINT_SEC_FRACTION         : UINT24;
    CASE 3 :    D_TIME                  : UINT32;
               U_SEC_FRACTION            : UINT32;
    CASE 4 :    D_TIME                  : UINT32;
               U_SEC_FRACTION            : UINT32;
  END;
END;

```

```

TYPE TIME_RCD = PACKED RECORD
  SWITCH GEN_CONFIG_TBL.TM_FORMAT OF
    CASE 0 :    NIL                    : NIL;
    CASE 1 :    HOUR                    : BCD;
               MINUTE                    : BCD;
               SECOND                    : BCD;
    CASE 2 :    HOUR                    : UINT8;
               MINUTE                    : UINT8;
               SECOND                    : UINT8;
    CASE 3 :    D_TIME                  : UINT32;
    CASE 4 :    D_TIME                  : UINT32;
  END;
END;

```

```

TYPE STIME_RCD = PACKED RECORD
  SWITCH GEN_CONFIG_TBL.TM_FORMAT OF
    CASE 0 :    NIL                    : NIL;
    CASE 1 :    HOUR                    : BCD;
               MINUTE                    : BCD;
    CASE 2 :    HOUR                    : UINT8;
               MINUTE                    : UINT8;
    CASE 3 :    D_TIME                  : UINT32;
    CASE 4 :    D_TIME                  : UINT32;
  END;
END;

```

TDL Element Descriptions

<u>Identifier</u>	<u>Value</u>	<u>Definition</u>
HTIME_DATE_RCD		High-precision (sub-second) time stamp. This is the prototype definition of the built-in type HTIME_DATE .

YEAR	00..89	Years 2000..2089.
	90..99	Years 1990..1999.
	>100	Reserved.
MONTH	0	Reserved.
	1..12	Month of year.
	>12	Reserved.
DAY	0	Reserved.
	1..31	Day of month.
	>31	Reserved.
HOURL	00..23	Hour of day, 24-hour basis.
	>23	Reserved.
MINUTE	00..59	Minute of hour.
	>59	Reserved.
SECOND	00..59	Seconds of minute.
	>59	Reserved.
BCD_SEC_FRACTION	0..999999	The sub-minute portion of SECOND expressed in units of 1 / 1 000 000 of a second.
UINT_SEC_FRACTION	0..16777215	The sub-minute portion of SECOND expressed in units of 1 / 16 777 216 of a second.
U_TIME	0..4294967295	Time in minutes since 01/01/1970 @ 00:00:00 GMT.
U_TIME_SEC	0..4294967295	Time in seconds since 01/01/1970 @ 00:00:00 GMT.
U_SEC_FRACTION	0..4294967295	The sub-second portion of U_TIME_SEC expressed in units of 1 / 4 294 967 296 of a second.
LTIME_DATE_RCD		Normal-precision (seconds) time stamp. This is the prototype definition of the built-in type LTIME_DATE .
YEAR	00..89	Years 2000..2089.
	90..99	Years 1990..1999.
	>100	Reserved.
MONTH	0	Reserved.
	1..12	Month of year.
	>12	Reserved.
DAY	0	Unassigned.
	1..31	Day of month.
	>31	Reserved.
HOURL	00..23	Hour of day, 24-hour basis.
	>23	Reserved.
MINUTE	00..59	Minute of hour.

	>59	Reserved.
SECOND	00..59 >59	Seconds of minute. Reserved.
U_TIME	0..4294967295	Time in minutes since 01/01/1970 @ 00:00:00 GMT.
U_TIME_SEC	0..4294967295	Time in seconds since 01/01/1970 @ 00:00:00 GMT.
STIME_DATE_RCD		Low-precision (minutes) time stamp. This is the prototype definition of the built-in type STIME_DATE .
YEAR	00..89 90..99 >100	Years 2000..2089. Years 1990..1999. Reserved.
MONTH	0 1..12 >12	Reserved. Month of year. Reserved.
DAY	0 1..31 >31	Unassigned. Day of month. Reserved.
HOURL	00..23 >23	Hour of day, 24-hour basis. Reserved.
MINUTE	00..59 >59	Minute of hour. Reserved.
U_TIME	0..4294967295	Time in minutes since 01/01/1970 @ 00:00:00 GMT.
HTIME_RCD		High-precision (sub-second) time interval or localized time. This is the prototype definition of the built-in type HTIME .
HOURL	00..23 >23	Hour of day, 24-hour basis. Reserved.
MINUTE	00..59 >59	Minute of hour. Reserved.
SECOND	00..59 >59	Seconds of minute. Reserved.
BCD_SEC_FRACTION	0..999999	The sub-minute portion of SECOND expressed in units of 1 / 1 000 000 of a second.
UINT_SEC_FRACTION	0..16777215	The sub-minute portion of SECOND expressed in units of 1 / 16 777 216 of a second.
D_TIME	0..4294967295	Time in seconds since 00:00:00 localized time.
U_SEC_FRACTION	0..4294967295	The sub-second portion of D_TIME expressed in units of 1 / 4 294 967 296 of a second.

TIME_RCD		Normal-precision (seconds) time interval or localized time. This is the prototype definition of the built-in type TIME .
HOUR	00..23 >23	Hour of day, 24-hour basis. Reserved.
MINUTE	00..59 >59	Minute of hour. Reserved.
SECOND	00..59 >59	Seconds of minute. Reserved.
D_TIME	0..4294967295	Time in seconds since 00:00:00 localized time.
STIME_RCD		Low-precision (minutes) time interval or localized time. This is the prototype definition of the built-in type STIME .
HOUR	00..23 >23	Hour of day, 24-hour basis. Reserved.
MINUTE	00..59 >59	Minute of hour. Reserved.
D_TIME	0..4294967295	Time in seconds since 00:00:00 localized time.

6.3.2 RDATE type

This type defines a recurrent date. This recurrence can be yearly, monthly, or weekly, based on a constant cycle of days or minutes.

TDL Type Definitions

```

TYPE RDATE_BFLD = BIT FIELD OF UINT16
    MONTH                                : UINT(0..3);
    SWITCH .MONTH OF
        CASE 0 :
            PERIOD_IN_MINUTES            : UINT(4..15);
        CASE 1..13 :
            OFFSET                        : UINT(4..7);
            WEEKDAY                      : UINT(8..10);
            DAY                          : UINT(11..15);
        CASE 14 :
            FILLER1                      : FILL(4..7);
            WEEKDAY                      : UINT(8..10);
            FILLER2                      : FILL(11..15);
        CASE 15 :
            PERIOD_IN_DAYS               : UINT(4..9);
            DELTA                        : UINT(10..15);
    END;
END;
```

TDL Element Descriptions

<u>Identifier</u>	<u>Value</u>	<u>Definition</u>
RDATE_BFLD		This is the prototype definition of the built-in type RDATE .
MONTH	0	Action is repeated each PERIOD_IN_MINUTES . The starting date and time shall be supplied separately.
	1	Action is repeated each January.
	2	Action is repeated each February.
	3	Action is repeated each March.
	4	Action is repeated each April.
	5	Action is repeated each May.
	6	Action is repeated each June.
	7	Action is repeated each July.
	8	Action is repeated each August.
	9	Action is repeated each September.
	10	Action is repeated each October.
	11	Action is repeated each November.
	12	Action is repeated each December.
	13	Action is repeated monthly.
	14	Action is repeated weekly.
	15	Action is repeated each PERIOD_IN_DAYS referenced by an ANCHOR_DATE plus DELTA . The ANCHOR_DATE shall be supplied separately.
PERIOD_IN_MINUTES	0..4095	0 to 4095 minutes (or 68 hours).
OFFSET	0	No offset.
	1	Advance to WEEKDAY before date entered.
	2	Postpone to the first WEEKDAY on or after date entered.
	3	Postpone to the second WEEKDAY on or after date entered.
	4	Postpone to the third WEEKDAY on or after date entered.
	5	Postpone to the fourth WEEKDAY on or after date entered.
	6	Postpone to the last WEEKDAY of the MONTH on or after date entered.
	7	Observe on date entered as well as day following date entered.
	8	Postpone to Monday if Sunday.
	9	Advance to Friday if Sunday.
	10	Postpone to Monday if Saturday.
	11	Advance to Friday if Saturday.
	12	Postpone to Monday if Sunday or Saturday.
	13	Advance to Friday if Sunday or Saturday.
	14	Postpone to Monday if Sunday, advance to Friday if Saturday.
	15	Do not observe date entered. Observe on day following date entered.
WEEKDAY	0	Sunday.
	1	Monday.
	2	Tuesday.
	3	Wednesday.
	4	Thursday.
	5	Friday.
	6	Saturday.

	7	Reserved.
DAY	0	Reserved.
	1..31	Day of month.
PERIOD_IN_DAYS	0..63	0 to 63 days.
DELTA	0..63	0 to 63 days.

6.3.3 DATE Type

This type defines a nonrecurrent date in the localized time using a field having the same size as the recurrent date type "RDATE".

TDL Type Definitions

```

TYPE DATE_BFLD = BIT FIELD OF UINT16
    YEAR      : UINT(0..6);
    MONTH : UINT(7..10);
    DAY       : UINT(11..15);
END;
```

TDL Element Descriptions

<u>Identifier</u>	<u>Value</u>	<u>Definition</u>
DATE_BFLD		This is the prototype definition of the built-in type DATE .
YEAR	00..89 90..99 >100	Years 2000..2089. Years 1990..1999. Reserved.
MONTH	0 1..12 >12	Reserved. Month of year. Reserved.
DAY	0 1..31	Reserved. Day of month.

6.4 Common table or procedure identifier formats

These types are provided to ease the specification of table access parameters and to provide consistent specification of these table access parameters throughout the document.

6.4.1 TABLE_IDA_BFLD bit field

Type **TABLE_IDA_BFLD** provides the table or procedure number; a flag indicating standard or manufacturer table or procedure; a flag indicating pending status; a flag indicating extended-user-defined table and two additional fields for definition and use within the defining data structure. When the **EUDT_FLAG** flag is set, then **MFG_FLAG** shall be set to zero.

TDL Type Definitions

```
TYPE TABLE_IDA_BFLD = BIT FIELD OF UINT16
    TBL_PROC_NBR          : UINT(0..10);
    MFG_FLAG              : BOOL(11);
    PENDING_FLAG          : BOOL(12);
    EUDT_FLAG             : BOOL(13);
    FLAG2                 : BOOL(14);
    FLAG3                 : BOOL(15);
END;
```

TDL Element Descriptions

Identifier	Value	Definition
TABLE_IDA_BFLD		
TBL_PROC_NBR	0..2039	Table number or Procedure number selected or identified.
MFG_FLAG	FALSE	Alias: STD_VS_MFG_FLAG. The Table or Procedure selected or identified by TBL_PROC_NBR is defined by this standard. The Table or Procedure selected or identified by TBL_PROC_NBR is defined by the End Device manufacturer.
	TRUE	
PENDING_FLAG	FALSE	The Table or Procedure selected or identified by TBL_PROC_NBR is an active instance of a Table or Procedure of the End Device. The Table or Procedure selected or identified by TBL_PROC_NBR is a pending (deferred) instance of a Table or Procedure of the End Device.
	TRUE	
EUDT_FLAG	FALSE	Alias: FLAG1. The entity selected or identified by TBL_PROC_NBR shall be an instance of a Table within the End Device. This table shall be listed in one of the sets GEN_CONFIG_TBL.STD_TBLS_USED or GEN_CONFIG_TBL.MFG_TBLS_USED . The entity selected or identified by TBL_PROC_NBR shall be an instance of an extended-user-defined Table of the End Device.
	TRUE	

FLAG2	FALSE	General-purpose flag to be Redefined where used. As defined where used.
	TRUE	As defined where used.
FLAG3	FALSE	General-purpose flag to be Redefined where used. As defined where used.
	TRUE	As defined where used.

6.4.2 TABLE_IDB_BFLD bit field

Type **TABLE_IDB_BFLD** provides the table or procedure number; a flag indicating standard or manufacturer table or procedure; and a 4-bit unsigned integer for definition and use within the defining data structure.

TDL Type Definitions

```

TYPE TABLE_IDB_BFLD = BIT FIELD OF UINT16
    TBL_PROC_NBR          : UINT(0..10);
    MFG_FLAG              : BOOL(11);
    SELECTOR              : UINT(12..15);
END;
```

TDL Element Descriptions

<u>Identifier</u>	<u>Value</u>	<u>Definition</u>
TABLE_IDB_BFLD TBL_PROC_NBR	0..2039	Table number or Procedure number selected or identified. This reference number shall be to an active Table or Procedure of the End Device; i.e., it shall not reference a pending Table, pending Procedure, or extended-user-defined Table.
MFG_FLAG	FALSE	Alias: STD_VS_MFG_FLAG. The Table or Procedure selected or identified by TBL_PROC_NBR is defined by this standard.
	TRUE	The Table or Procedure selected or identified by TBL_PROC_NBR is defined by the End Device manufacturer.
SELECTOR	0..15	A general-purpose selector that is encoded as an enumerated Sub-element which shall be Redefined where used.

6.4.3 TABLE_IDC_BFLD bit field

Type **TABLE_IDC_BFLD** provides the table or procedure number; a flag indicating standard or manufacturer; a flag indicating table or procedure; and three additional fields for definition and use within the defining data structure.

TDL Type Definitions

```
TYPE TABLE_IDC_BFLD = BIT FIELD OF UINT16
    TBL_PROC_NBR          : UINT(0..10);
    MFG_FLAG              : BOOL(11);
    PROC_FLAG             : BOOL(12);
    FLAG1                 : BOOL(13);
    FLAG2                 : BOOL(14);
    FLAG3                 : BOOL(15);
END;
```

TDL Element Descriptions

Identifier	Value	Definition
TABLE_IDC_BFLD		
TBL_PROC_NBR	0..2039	Table number or Procedure number selected or identified. This reference number shall be to an active Table or Procedure of the End Device. i.e., it shall not reference a pending Table, pending Procedure or extended-user-defined Table.
MFG_FLAG	FALSE TRUE	Alias: STD_VS_MFG_FLAG. The Table or Procedure selected or identified by TBL_PROC_NBR is defined by this standard. The Table or Procedure selected or identified by TBL_PROC_NBR is defined by the End Device manufacturer.
PROC_FLAG	FALSE TRUE	The TBL_PROC_NBR is a table number. The TBL_PROC_NBR is a procedure number.
FLAG1	FALSE TRUE	General-purpose flag to be Redefined where used. As defined where used. As defined where used.
FLAG2	FALSE TRUE	General-purpose flag to be Redefined where used. As defined where used. As defined where used.
FLAG3	FALSE TRUE	General-purpose flag to be Redefined where used. As defined where used. As defined where used.

6.4.4 SOURCE_SELECT_RCD

Type **SOURCE_SELECT_RCD** provides a data source selection index and related qualifiers that describe the data source completely and unambiguously used in selections of data sources in control areas. This data type is used each time a measurement needs to be selected from a source defined in Decade 1.

TDL Type Definitions

```
TYPE SOURCE_SELECT_RCD = PACKED RECORD
  SWITCH GEN_CONFIG_TBL.MODEL_SELECT OF
    CASE 0..1 :
      SOURCE_INDEX          : UINT8;
  END;
END;
```

TDL Element Descriptions

<u>Identifier</u>	<u>Value</u>	<u>Definition</u>
SOURCE_SELECT_RCD SOURCE_INDEX	0..255	When MODEL_SELECT is set to 0 or 1, this index selects a source defined in Decade 1 using 8-bit selection indices.

7 Compliance and compatibility

7.1 Compliance

An End Device is considered to be in compliance with this standard if all of the following conditions are met:

- a) Two-way communicating End Devices shall accept and act upon all services defined in Clause 8, “Table transportation issues”.
- b) The **DEVICE_CLASS** field in **GEN_CONFIG_TBL** (Table 00) shall contain a relative universal object identifier relative to the End Device class registered root context.
- c) Two-way communicating End Devices shall transmit **GEN_CONFIG_TBL** (Table 00) on request.
- d) One-way communicating End Devices shall include the **DEVICE_CLASS** field in each of their transmissions. In addition, these End Devices shall be registered using the TDL file format. This file shall contain the definition of the corresponding Table 00 and any other Tables necessary to understand all possible valid unsolicited messages.
- e) Any Table transported between an End Device and any other device shall be either a Standard Table as defined by this standard or a Manufacturer Table defined by the End Device manufacturer. All Standard and Manufacturer Tables supported for the current End Device implementation shall be specified in **GEN_CONFIG_TBL** (Table 00). A Table shall be considered to be supported if and only if it is specified in **GEN_CONFIG_TBL** (Table 00).
- f) Any procedure written to the End Device shall be either a procedure as defined by this standard or a manufacturer procedure defined by the device manufacturer. All standard and manufacturer procedures supported for the current End Device implementation shall be specified in **GEN_CONFIG_TBL** (Table 00).
- g) Some Standard Tables contain data structures that can be collapsed by a control flag. The control flag shall be determinable based on this standard. Collapsed Elements shall not be transported. An array of zero (0) Elements is a collapsed Element.
- h) An End Device is not required to use every data structure within a Table; however, it shall transport all noncollapsed data structures within a Table. Unused fields are considered in compliance if **Octet Count** and format are correct and if manufacturers identify unused fields. It is recommended that unused fields be filled with zeros or spaces.
- i) This standard specifies variable formats for some data structures. The variable formats shall be consistent with the format as specified in **GEN_CONFIG_TBL** (Table 00). Refer to Clause 5, “Syntax”, and Clause 6, “Special data types”, for details.
- j) Function Limiting Tables (FLCs) and Actual Function Limiting Tables (FLC+1) specify the designed and usable limits for variables used within a given Decade. The permissible combinations of these Tables are listed in Figure 4-1.

7.2 Backward and forward compatibility

Any future revision of this standard shall be backward compatible with the previous two (2) revisions of the standard according to the governing standard's revision cycle requirement. It is recommended that new implementations of the standard avoid using Deprecated Elements and attributes whenever practical. Forward compatibility is not ensured as Deprecated Elements may become obsolete in future versions of the standard or End Device.

Element extensions to the tail end of the Tables of the previous versions of this standard are defined in a manner that does not change the structure of the Tables Elements defined by the previous standard version. Also, when Element fields were modified, the changes do not impact the interoperation of those Elements according to the previous standard version.