

*A Recommended Standard of the Joint Committee on the NTCIP*

# **NTCIP 1201 v04.08a**

---

## **National Transportation Communications for ITS Protocol Global Object (GO) Definitions**

---

January 9, 2025

This is a draft document, which is distributed for review and comment purposes only. You may reproduce and distribute this document within your organization, but only for the purposes of and only to the extent necessary to facilitate review and comment to the NTCIP Coordinator. Please ensure that all copies include this notice. This document contains preliminary information that is subject to change.

*Published by*

**American Association of State Highway and Transportation Officials (AASHTO)**  
555 12<sup>th</sup> Street, N.W., Suite 1000  
Washington, D.C. 20004

**Institute of Transportation Engineers (ITE)**  
1627 I Street, N.W., Suite 550  
Washington, D.C. 20006-3438

**National Electrical Manufacturers Association (NEMA)**  
1300 17th St N #900  
Arlington, VA 22209-3801

© 2025 AASHTO / ITE / NEMA. All rights reserved.

### Revision History for NTCIP 1201 v04

to be removed once approved

Revision	Date	Author	Description (latest on top)
08a	01/09/2025	Vaughn	Update to RS in preparation for ballot
08	08/30/2024	Vaughn	Editorial changes preparing for pRS
07	08/29/2023	Vaughn	Post UCD Edits
06	07/01/2023	Vaughn	Removed systems engineering process information to prevent conflicts with previously published standards.
05	06/08/2023	Vaughn	Added systems engineering process information; moved transaction mode version 2 and recording mechanism version 2 to ISO 26048-1.
04	02/13/2023	Vaughn	Updated status to proposed User Comment Draft; updated copyright year to 2023; removed internal WG comments
03	01/09/2023	Vaughn	Revised text in Acknowledgements to correct page numbering; moved many of the comments into description fields to conform with SMIV2 rules that prohibit normative comments; corrected typos. Ensured consistent use of subclauses within DESCRIPTION clause. Added the NTCIP1201-Global MIB to define nodes that need to be imported by multiple modules. Removed comments that have been addressed by the WG. Updated MIB contact information and parent MIB structure.
02	11/20/2022	Vaughn	Updated front matter and Section 1, simplified OBJECT-GROUPS, renamed modules for consistency, updated annexes.
01	10/21/2022	Vaughn	Initial draft document to incorporate security improvements as recommended by NTCIP 9014 and the Infrastructure Standards Security Assessment (ISSA) Project.

## NOTICES

### Copyright Notice

© 1996, 1998, 2005, 2011, 2025 by the American Association of State Highway and Transportation Officials (AASHTO), the Institute of Transportation Engineers (ITE), and the National Electrical Manufacturers Association (NEMA). All intellectual property rights, including, but not limited to, the rights of reproduction, translation, and display are reserved under the laws of the United States of America, the Universal Copyright Convention, the Berne Convention, and the International and Pan American Copyright Conventions. Except as licensed or permitted, you may not copy these materials without prior written permission from AASHTO, ITE, or NEMA. Use of these materials does not give you any rights of ownership or claim of copyright in or to these materials.

Visit [www.ntcip.org](http://www.ntcip.org) for other copyright information, for instructions to request reprints of excerpts, and to request reproduction that is not granted below.

### PDF File License Agreement

To the extent that these materials are distributed by AASHTO / ITE / NEMA in the form of an Adobe® Portable Document Format (PDF) electronic data file (the "PDF file"), AASHTO / ITE / NEMA authorizes each registered PDF file user to view, download, copy, or print the PDF file available from the authorized website, subject to the terms and conditions of this license agreement:

1. You may download one copy of each PDF file for personal, noncommercial, and intraorganizational use only.
2. Ownership of the PDF file is not transferred to you; you are licensed to use the PDF file.
3. You may make one more electronic copy of the PDF file, such as to a second hard drive or burn to a CD.
4. You agree not to copy, distribute, or transfer the PDF file from that media to any other electronic media or device.
5. You may print one paper copy of the PDF file.
6. You may make one paper reproduction of the printed copy.
7. Any permitted copies of the PDF file must retain the copyright notice, and any other proprietary notices contained in the file.
8. The PDF file license does not include (1) resale of the PDF file or copies, (2) republishing the content in compendiums or anthologies, (3) publishing excerpts in commercial publications or works for hire, (4) editing or modification of the PDF file except those portions as permitted, (5) posting on network servers or distribution by electronic mail or from electronic storage devices, and (6) translation to other languages or conversion to other electronic formats.
9. Other use of the PDF file and printed copy requires express, prior written consent.

### Data Dictionary and MIB Distribution Permission

To the extent that these materials are distributed by AASHTO / ITE / NEMA in the form of a Data Dictionary ("DD") or Management Information Base ("MIB"), AASHTO / ITE / NEMA extend the following permission:

You may make or distribute unlimited copies, in part or full, of the DD or MIB, including copies for commercial distribution, provided that:

1. each copy you make or distribute includes the citation "From NTCIP 0000 [insert the standard number]. Copyright by AASHTO / ITE / NEMA. Used by permission.";

2. the copies are not edited from the published version of this document;

These materials are delivered "AS IS" without any warranties as to their use or performance.

**AASHTO / ITE / NEMA and their suppliers do not warrant the performance or results you may obtain by using these materials. AASHTO / ITE / NEMA and their suppliers make no warranties, express or implied, as to noninfringement of third party rights, merchantability, or fitness for any particular purpose. In no event will AASHTO / ITE / NEMA or their suppliers be liable to you or any third party for any claim or for any consequential, incidental or special damages, including any lost profits or lost savings, arising from your reproduction or use of these materials, even if an AASHTO / ITE / NEMA representative has been advised of the possibility of such damages.**

Some states or jurisdictions do not allow the exclusion or limitation of incidental, consequential, or special damages, or the exclusion of implied warranties, so the above limitations may not apply to a given user.

Use of these materials does not constitute an endorsement or affiliation by or between AASHTO, ITE, or NEMA and the user, the user's company, or the products and services of the user's company.

If the user is unwilling to accept the foregoing restrictions, he or she should immediately return these materials.

#### **Content and Liability Disclaimer**

The information in this publication was considered technically sound by the consensus of persons engaged in the development and approval of the document at the time it was developed. Consensus does not necessarily mean that there is unanimous agreement among every person participating in the development of this document.

AASHTO, ITE, and NEMA standards and guideline publications, of which the document contained herein is one, are developed through a voluntary consensus standards development process. This process brings together volunteers and seeks out the views of persons who have an interest in the topic covered by this publication. While AASHTO, ITE, and NEMA administer the process and establish rules to promote fairness in the development of consensus, they do not write the document and they do not independently test, evaluate, or verify the accuracy or completeness of any information or the soundness of any judgments contained in their standards and guideline publications.

AASHTO, ITE, and NEMA disclaim liability for any personal injury, property, or other damages of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, application, or reliance on this document. AASHTO, ITE, and NEMA disclaim and make no guaranty or warranty, express or implied, as to the accuracy or completeness of any information published herein and disclaims and makes no warranty that the information in this document will fulfill any of your purposes or needs. AASHTO, ITE, and NEMA do not undertake to guarantee the performance of any individual manufacturer or seller's products or services by virtue of this standard or guide.

In publishing and making this document available, AASHTO, ITE, and NEMA are not undertaking to render professional or other services for or on behalf of any person or entity, nor are AASHTO, ITE, and NEMA undertaking to perform any duty owed by any person or entity to someone else. Anyone using this document should rely on his or her own independent judgment or, as appropriate, seek the advice of a competent professional in determining the exercise of reasonable care in any given circumstance. Information and other standards on the topic covered by this publication may be available from other sources, which the user may wish to consult for additional views or information not covered by this publication.

AASHTO, ITE, and NEMA have no power, nor do they undertake to police or enforce compliance with the contents of this document. AASHTO, ITE, and NEMA do not certify, test, or inspect products, designs, or installations for safety or health purposes. Any certification or other statement of compliance with any health or safety-related information in this document shall not be attributable to AASHTO, ITE, or NEMA and is solely the responsibility of the certifier or maker of the statement.

**Trademark Notice**

NTCIP is a trademark of AASHTO / ITE / NEMA. All other marks mentioned in this standard are the trademarks of their respective owners.



## ACKNOWLEDGEMENTS

NTCIP 1201 was prepared by the NTCIP Base Standards and Profiles Working Group (BSP2 WG), which is a subdivision of the Joint Committee on the NTCIP. The Joint Committee on the NTCIP is organized under a Memorandum of Understanding among the American Association of State Highway and Transportation Officials (AASHTO), the Institute of Transportation Engineers (ITE), and the National Electrical Manufacturers Association (NEMA). The Joint Committee on the NTCIP consists of six representatives from each of the standards development organizations (SDOs) and provides guidance for NTCIP development.

When this document was prepared, the following individuals were members of BSP2 WG:

- |                          |                      |
|--------------------------|----------------------|
| 1. Doug Crawford (Chair) | 8. Robert Lopes      |
| 2. Steve Bostrom         | 9. Alex Mousadi      |
| 3. Wolfgang Buckel       | 10. Satya Muthuswamy |
| 4. Michael Forbis        | 11. Bob Rausch       |
| 5. Joe Gorman            | 12. Rodney Schilling |
| 6. Terry Haukom          | 13. Shea Tomsin      |
| 7. Patrick Leung         | 14. Walt Townsend    |

Other individuals providing input include the following:

- |                     |                       |
|---------------------|-----------------------|
| 1. Md Ashraf Ahmed  | 18. Stephane Lapierre |
| 2. Justin Anderson  | 19. Christopher Lyons |
| 3. Kingsley Azubike | 20. Gary Molnar       |
| 4. Ralph Boaz       | 21. Keith Patton      |
| 5. Russ Brookshire  | 22. Mahesh Pawar      |
| 6. Patrick Chan     | 23. Frank Perry       |
| 7. Stan Chow        | 24. Nu Rosenbohm      |
| 8. Deborah Curtis   | 25. Jay Schultz       |
| 9. Dustin DeVoe     | 26. Kellen Shain      |
| 10. Brian Doherty   | 27. Douglas Tarico    |
| 11. Edward Fok      | 28. Nicola Tavares    |
| 12. Sai Kiran Golla | 29. John Thai         |
| 13. Jacob Grivette  | 30. Andrew Valdez     |
| 14. Mohammad Iraki  | 31. Kenneth Vaughn    |
| 15. Haydar Issa     | 32. Sankeer Vulupala  |
| 16. Denver Kruse    | 33. Robert White      |
| 17. AJ Lahiri       |                       |

In addition to the many volunteer efforts, recognition is also given to those organizations that supported the efforts of BSP2 WG by providing comments and funding, including the following:

- |   |                       |
|---|-----------------------|
| 1. U.S. Department of Transportation, Research and Innovative Technology Administration |                       |
| 2. U.S. Department of Transportation, Federal Highway Administration                    |                       |
| 3. AECOM  | 12. Kapsch            |
| 4. Applied Information  | 13. KLD Engineering   |
| 5. City of Anaheim, CA  | 14. Michigan DOT      |
| 6. Consensus Systems Technologies   | 15. Minnesota DOT     |
| 7. Caltrans   | 16. Nevada DOT        |
| 8. Daktronics   | 17. Noblis            |
| 9. Econolite Control Products, Inc.   | 18. Parsons           |
| 10. Florida Department of Transportation  | 19. Pillar Consulting |
| 11. Georgia DOT   | 20. Q-Free            |

NTCIP 1201 v04.08a  
Page ii

21. Siemens ITS
22. Signalisation Ver-Mac, Inc.
23. TransCore
24. Transport Canada
25. Trevilon
26. Washington State DOT
27. WSP



## FOREWORD

This document is an NTCIP Data Dictionary Standard. NTCIP Data Dictionary Standards provide definitions of data elements for use within NTCIP systems. The data is defined using the format specified in NTCIP 8004, which is based on the formats defined in RFC 2578. This data is typically exchanged using the Simple Network Management Protocol (SNMP) as defined in NTCIP 2301.

NTCIP Data Dictionary Standards are separately balloted and approved by AASHTO, ITE, and NEMA, after recommendation by the Joint Committee on the NTCIP. Each organization has approved this document as the following standard type:

AASHTO–Standard Specification;

ITE–Software Standard;

NEMA–Standard;

For more information about NTCIP standards, visit the NTCIP website at [www.ntcip.org](http://www.ntcip.org).

### User Comment Instructions

The term "User Comment" includes any type of written inquiry, comment, question, or proposed revision, from an individual person or organization, about any part of this standards publication's content. A "Request for Interpretation" is also classified as a User Comment. User Comments are solicited at any time. In preparation of this NTCIP standards publication, input of users and other interested parties was sought and evaluated.

All User Comments will be referred to the committee responsible for developing and/or maintaining this standards publication. The committee chairperson, or their designee, may contact the submitter for clarification of the User Comment. When the committee chairperson or designee reports the committee's consensus opinion related to the User Comment, that opinion will be forwarded to the submitter. The committee chairperson may report that action on the User Comment may be deferred to a future committee meeting and/or a future revision of the standards publication. Previous User Comments and their disposition may be available for reference and information at [www.ntcip.org](http://www.ntcip.org).

A User Comment should be submitted to this address:

NTCIP Coordinator  
National Electrical Manufacturers Association  
1300 17th St N #900  
Arlington, VA 22209  
e-mail: [ntcip@nema.org](mailto:ntcip@nema.org)

A User Comment should be submitted in the following form:

**Standards Publication number and version:**  
**Page:**  
**Section, Paragraph, or Clause:**  
**Comment:**  
**Editorial or Substantive?:**  
**Suggested Alternative Language:**

Please include your name, organization, and address in your correspondence.

**History**

<b>Version</b>	<b>Date</b>	<b>Description (latest on top)</b>
V04	TBD	Incorporated NTCIP 1103 MIBS, updated to SMIv2, deprecated objects to reference ISO and IETF standards.
v03	March 2011	Revised daylight-saving design
v02	October 2005	Developed to reflect additional lessons learned, to incorporate better documentation (in the Annex) of some of the logic required to implement the standards, and to add new features requested by the ITS community.
v01A	December 1999	Technical corrections
v01	December 1997	Original version balloted and approved by AASHTO and ITE to become an NTCIP standard.
NEMA TS 3.4	October 1996	Original version approved by NEMA prior to the formation of the Joint Committee process.

Details about the revisions made since the last major version are provided in Annex D.

## **INTRODUCTION**

This document identifies and defines the common object definitions that may be supported by transportation devices that are NTCIP-conformant.

This document contains two normative and four informative annexes.

The following keywords apply to this document: AASHTO, ITE, NEMA, NTCIP, global, data, data dictionary, object.

This document only uses metric units.

**CONTENTS**

**Page**

SECTION 1 GENERAL.....	1
1.1 Scope.....	1
1.2 References.....	2
1.2.1 Normative References.....	2
1.2.2 Other References.....	2
1.2.3 Contact Information.....	3
1.3 General Statements.....	3
1.3.1 Use of SMIv2.....	3
1.3.2 See Other Documents for Additional Details.....	3
1.3.3 SNMPv3.....	4
1.3.4 Superseding and Supplanting Objects.....	4
1.4 Terms.....	4
SECTION 2 DEPRECATED GLOBAL MIB.....	6
2.1 NTCIP Header.....	6
2.2 Object IDENTITIES.....	7
2.2.1 Global Configuration Node.....	7
2.3 Objects.....	7
2.3.1 Global Set ID Parameter.....	7
2.3.2 Maximum Modules Parameter.....	7
2.3.3 Module Table.....	8
2.3.4 Base Standards Parameter.....	10
2.4 Global Database Management Node.....	11
2.4.1 Database Creation Transaction.....	12
2.4.2 Database Error Type Parameter.....	14
2.4.3 Database Error ID Parameter.....	15
2.4.4 Database Transaction ID Parameter.....	15
2.4.5 Database Make ID Parameter.....	15
2.4.6 Database Verify Status Parameter.....	16
2.4.7 Database Verify Error Parameter.....	16
2.5 Global Time Management Node.....	17
2.5.1 Global Time Parameter.....	17
2.5.2 Global Daylight Saving Parameter.....	17
2.6 TimeBase Event Scheduler Node.....	19
2.6.1 Maximum Number of Time Base Schedule Entries Parameter.....	19
2.6.2 Time Base Schedule Table.....	19
2.6.3 Maximum Number of Day Plans-Parameter.....	23
2.6.4 Maximum Number of Day Plan Events-Parameter.....	24
2.6.5 Day Plan Table.....	24
2.6.6 Day Plan Status Parameter.....	27
2.6.7 Schedule Status Parameter.....	27
2.6.8 Global Local Time Differential Parameter.....	28
2.6.9 Standard Time Zone Parameter.....	28
2.6.10 Local Time Parameter.....	28
2.7 Daylight Saving Time (DST) Node.....	29
2.7.1 Maximum Daylight Saving Time (DST) Table Entries Parameter.....	29
2.7.2 Daylight Saving Time (DST) Table Parameter.....	30
2.8 PMPP Object Node.....	36
2.8.1 Maximum HDLC Group Address Parameter.....	37
2.8.2 HDLC Group Address Table.....	37
SECTION 3 DEPRECATED AUXILIARY I/O V2 MIB.....	39

- 3.1 Auxiliary I/O V2 Header ..... 39
- 3.2 Objects..... 40
  - 3.2.1 Maximum Number of Digital Auxiliary I/Os Parameter ..... 40
  - 3.2.2 Maximum Number of Analog Auxiliary I/Os Parameter ..... 40
  - 3.2.3 Auxiliary I/O Table Parameter ..... 40
- SECTION 4 DEPRECATED AUXILIARY I/O MIB ..... 45
  - 4.1 Auxiliary I/O Header ..... 45
  - 4.2 Objects..... 46
    - 4.2.1 Maximum Number of Digital Auxiliary I/Os Parameter ..... 46
    - 4.2.2 Maximum Number of Analog Auxiliary I/Os Parameter ..... 46
    - 4.2.3 Auxiliary I/O Table Parameter ..... 47
- SECTION 5 DEPRECATED SNMP MIB ..... 50
  - 5.1 Header ..... 50
  - 5.2 Objects..... 50
    - 5.2.1 Max Packet Size ..... 50
- SECTION 6 DEPRECATED SFMP MIB ..... 52
  - 6.1 Header ..... 52
  - 6.2 Object Identities ..... 53
    - 6.2.1 SFMP Statistics..... 53
  - 6.3 Objects..... 53
    - 6.3.1 Number of Incoming SFMP Packets ..... 53
    - 6.3.2 Number of Outgoing SFMP Packets..... 53
    - 6.3.3 Number of Incoming SFMP Packets with Bad Version Numbers ..... 53
    - 6.3.4 Number of Incoming SFMP Packets with Bad Community Names ..... 54
    - 6.3.5 Number of Incoming SFMP Packets with Bad Use of a Community Name ..... 54
    - 6.3.6 Number of Incoming SFMP Packets with Parsing Errors ..... 54
    - 6.3.7 Reserved ..... 54
    - 6.3.8 Number of Incoming SFMP Packets indicating a Too Big Error ..... 54
    - 6.3.9 Number of Incoming SFMP Packets indicating a No Such Name Error ..... 55
    - 6.3.10 Number of Incoming SFMP Packets indicating a Bad Value Error ..... 55
    - 6.3.11 Number of Incoming SFMP Packets indicating a Read-Only Error ..... 55
    - 6.3.12 Number of Incoming SFMP Packets indicating a General Error ..... 55
    - 6.3.13 Reserved..... 56
    - 6.3.14 Reserved ..... 56
    - 6.3.15 Number of Incoming SFMP Get Requests ..... 56
    - 6.3.16 Reserved ..... 56
    - 6.3.17 Number of Incoming SFMP Set Requests ..... 56
    - 6.3.18 Number of Incoming SFMP Get Responses..... 56
    - 6.3.19 Reserved ..... 57
    - 6.3.20 Number of Outgoing SFMP Packets indicating a Too Big Error ..... 57
    - 6.3.21 Number of Outgoing SFMP Packets indicating a No Such Name Error ..... 57
    - 6.3.22 Number of Outgoing SFMP Packets indicating a Bad Value Error ..... 57
    - 6.3.23 Number of Outgoing SFMP Packets indicating a Read-Only Error ..... 58
    - 6.3.24 Number of Outgoing SFMP Packets indicating a General Error ..... 58
    - 6.3.25 Number of Outgoing SFMP Get Requests ..... 58
    - 6.3.26 Reserved ..... 58
    - 6.3.27 Number of Outgoing SFMP Set Requests ..... 58
    - 6.3.28 Number of Outgoing SFMP Get Responses..... 59
    - 6.3.29 Number of Outgoing SFMP Trap Messages..... 59
    - 6.3.30 Reserved ..... 59
    - 6.3.31 Number of Incoming SFMP Set Requests - No Replies ..... 59
    - 6.3.32 Number of Incoming SFMP Set Responses ..... 59
    - 6.3.33 Number of Incoming SFMP Error Responses ..... 60
    - 6.3.34 Number of Outgoing SFMP Set Requests - No Replies ..... 60

6.3.35	Number of Outgoing SFMP Set Responses .....	60
6.3.36	Number of Outgoing SFMP Error Responses .....	61
SECTION 7 DEPRECATED DYNAMIC OBJECT MIB .....		62
7.1	Header .....	62
7.2	Textual Conventions .....	63
7.3	Object Identities .....	63
7.3.1	Data Node .....	63
7.4	Dynamic Object Definition .....	63
7.4.1	Maximum Dynamic Object Table Entries .....	63
7.4.2	Dynamic Object Definition Table .....	63
7.5	Dynamic Object Information .....	66
7.5.1	Dynamic Object Data .....	66
7.5.2	Dynamic Object Configuration .....	70
SECTION 8 DEPRECATED STMP MIB .....		72
8.1	Header .....	72
8.2	Object Identities .....	73
8.2.1	STMP Statistics Node .....	73
8.3	Objects .....	73
8.3.1	Number of Incoming STMP Packets .....	73
8.3.2	Number of Outgoing STMP Packets .....	73
8.3.3	Reserved .....	73
8.3.4	Reserved .....	74
8.3.5	Reserved .....	74
8.3.6	Number of Incoming STMP Packets with Parsing Errors .....	74
8.3.7	Reserved .....	74
8.3.8	Number of Incoming STMP Packets indicating a Too Big Error .....	74
8.3.9	Number of Incoming STMP Packets indicating a No Such Name Error .....	74
8.3.10	Number of Incoming STMP Packets indicating a Bad Value Error .....	75
8.3.11	Number of Incoming STMP Packets indicating a Read-Only Error .....	75
8.3.12	Number of Incoming STMP Packets indicating a General Error .....	75
8.3.13	Reserved .....	76
8.3.14	Reserved .....	76
8.3.15	Number of Incoming STMP Get Requests .....	76
8.3.16	Number of Incoming STMP Get Next Requests .....	76
8.3.17	Number of Incoming STMP Set Requests .....	77
8.3.18	Number of Incoming STMP Get Responses .....	77
8.3.19	Reserved .....	77
8.3.20	Number of Outgoing STMP Packets indicating a Too Big Error .....	77
8.3.21	Number of Outgoing STMP Packets indicating a No Such Name Error .....	78
8.3.22	Number of Outgoing STMP Packets indicating a Bad Value Error .....	78
8.3.23	Number of Outgoing STMP Packets indicating a Read-Only Error .....	78
8.3.24	Number of Outgoing STMP Packets indicating a General Error .....	79
8.3.25	Number of Outgoing STMP Get Requests .....	79
8.3.26	Number of Outgoing STMP Get Next Requests .....	80
8.3.27	Number of Outgoing STMP Set Requests .....	80
8.3.28	Number of Outgoing STMP Get Responses .....	80
8.3.29	Reserved .....	81
8.3.30	Reserved .....	81
8.3.31	Number of Incoming STMP Set Request - No Replies .....	81
8.3.32	Number of Incoming STMP Set Responses .....	81
8.3.33	Number of Incoming STMP Error Responses .....	81
8.3.34	Number of Outgoing STMP Set Request - No Replies .....	82
8.3.35	Number of Outgoing STMP Set Responses .....	82
8.3.36	Number of Outgoing STMP Error Responses .....	82

SECTION 9 DEPRECATED STMP CONFIGURATION MIB .....	84
9.1 Header .....	84
9.2 Objects .....	85
9.2.1 Dynamic Object Persistence .....	85
9.2.2 Dynamic Object Configuration ID .....	85
SECTION 10 DEPRECATED LOGICAL NAMES MIB .....	86
10.1 Header .....	86
10.2 Objects .....	87
10.2.1 Maximum Logical Name Translations .....	87
10.2.2 Logical Name Translation Table .....	87
SECTION 11 DEPRECATED REPORT MIB .....	90
11.1 Header .....	90
11.2 Objects .....	91
11.2.1 Event Classes .....	91
11.2.2 Event Class Table .....	91
11.2.3 Maximum Event Log Configurations Parameter .....	94
11.2.4 Event Log Configuration Table .....	94
11.2.5 Maximum Event Log Size Parameter .....	101
11.2.6 Event Log Table .....	101
11.2.7 Total Event Log Counter Parameter .....	105
11.2.8 Event Log Time Latency Parameter .....	105
SECTION 12 DEPRECATED SECURITY MIB .....	106
12.1 Header .....	106
12.2 Objects .....	107
12.2.1 Community Name Administrator Parameter .....	107
12.2.2 Maximum Community Names Parameter .....	107
12.2.3 Community Names Table .....	107
SECTION 13 DEPRECATED TRAP MIB .....	110
13.1 Header .....	110
13.2 Object Identities .....	111
13.2.1 Watch Blocks .....	111
13.2.2 Report Blocks .....	111
13.2.3 Clear Objects .....	111
13.2.4 NTCIP Trap Management .....	112
13.2.5 NTCIP Trap Data .....	112
13.2.6 NTCIP Trap Notifications .....	112
13.3 Watch Blocks .....	112
13.3.1 Maximum Watch Objects .....	112
13.3.2 Maximum Watch Blocks .....	113
13.3.3 Watch Object Definition Table .....	113
13.3.4 Watch Block Table .....	115
13.4 Report Blocks .....	117
13.4.1 Maximum Report Objects .....	117
13.4.2 Maximum Report Blocks .....	118
13.4.3 Report Object Configuration Table .....	118
13.4.4 Report Block Table .....	120
13.5 Trap Management .....	122
13.5.1 Trap Control .....	122
13.5.2 Trap Data .....	122
13.5.3 Trap Management Maximum Entries .....	123
13.5.4 Trap Maximum Aggregation Events .....	123
13.5.5 Trap Maximum Aggregation Size .....	123

13.5.6	Trap Management Table.....	123
13.5.7	Trap Table.....	131
13.6	NTCIP Trap Data.....	134
13.6.1	Event Trap.....	134
13.7	Clear Event Data.....	135
13.7.1	Clear Event Class.....	135
13.7.2	Clear Event Configuration.....	135
13.7.3	Clear Event Log Table.....	136
13.7.4	Clear Report Objects.....	136
13.7.5	Clear Report Block Table.....	136
13.7.6	Clear Watch Objects.....	137
13.7.7	Clear Watch Block Table.....	137
13.7.8	Clear Trap Management Table.....	137
SECTION 14 DEPRECATED RECORDING MECHANISM MIB.....		139
14.1	Header.....	139
14.2	Objects.....	139
14.2.1	Maximum Recording Mechanism Classes Parameter.....	139
14.2.2	Recording Mechanism Class Table.....	140
14.2.3	Maximum Recording Configurations.....	142
14.2.4	Minimum Recording Sample Period.....	143
14.2.5	Maximum Recording Sample Period.....	143
14.2.6	Recording Sample Period Resolution.....	143
14.2.7	Recording Configuration Table.....	144
14.2.8	Maximum Recordings Parameter.....	152
14.2.9	Recording Table.....	152
14.2.10	Maximum Recording Entries Parameter.....	156
14.2.11	Recording Entries Table.....	157
14.2.12	Total Recordings Counter Parameter.....	158
14.2.13	Clear Recording Classes.....	159
14.2.14	Clear Recording Configurations.....	159
14.2.15	Clear Recording Data.....	159
ANNEX A OBJECT TREE AND UML CLASS DIAGRAMS [INFORMATIVE].....		161
A.1	Object Identifier Naming Tree.....	161
A.2	UML Class Diagrams.....	174
A.2.1	Controller Class Diagram.....	174
A.2.2	Configuration Information.....	175
A.2.3	Transaction Information.....	176
A.2.4	Time and Daylight Saving Time (DST) Information.....	177
A.2.5	Generic Schedule Information.....	178
A.2.6	PMPP Information.....	180
A.2.7	Auxiliary Input/Output Information.....	181
A.2.8	SNMP Information.....	182
A.2.9	SFMP Information.....	182
A.2.10	Dynamic Object Management Information.....	184
A.2.11	STMP Information.....	185
A.2.12	Profiles STMP.....	186
A.2.13	Logical Names Information.....	187
A.2.14	Event Report Information.....	187
A.2.15	Community Name Security Information.....	189
A.2.16	Trap Management Information.....	189
A.2.17	Recording Mechanism (Version 1) Information.....	192
ANNEX B SUMMARY OF CHANGES [INFORMATIVE].....		195
B.1	Inclusion of MIBs from NTCIP 1103 v03.....	195
B.2	Upgrade to SMIv2 and Deprecate NTCIP 1103 v03 and NTCIP 1201 v03 Objects.....	195



B.3 Summary ..... 196

**FIGURES**

	<b>Page</b>
Figure 1: Focus of this standard .....	1
Figure 2: MODULE-IDENTITY nodes.....	161
Figure 3: Nodes Defined in NTCIP 1201-Global MIB (Part 1 of 3: Base Objects) .....	162
Figure 4: Nodes Defined in NTCIP 1201-Global MIB (Part 2 of 3: Time Management) .....	163
Figure 5: Nodes Defined in NTCIP 1201-Global MIB (Part 3 of 3: PMPP Objects) .....	163
Figure 6: Nodes Defined in NTCIP1201-AuxIOv2 MIB .....	164
Figure 7: Nodes Defined in NTCIP1201-AuxIO MIB .....	164
Figure 8: Nodes Defined in NTCIP1201-SNMPCConfig MIB .....	164
Figure 9: Nodes Defined in NTCIP1201-SFMP MIB .....	165
Figure 10: Nodes Defined in NTCIP1201-DynObjMgmt MIB .....	166
Figure 11: Nodes Under NTCIP1201-StmpStatistics MIB .....	167
Figure 12: Nodes Under NTCIP1201-ProfilesSTMP MIB.....	168
Figure 13: Nodes under NTCIP1201-LogicalNames MIB .....	168
Figure 14: Nodes Defined in NTCIP1201-GlobalReport MIB .....	169
Figure 15: Nodes Defined in NTCIP1201-Security MIB .....	170
Figure 16: Nodes Defined in NTCIP1201-NtcipTraps MIB (Part 1 of 2: Application Objects).....	171
Figure 17: Nodes Defined in NTCIP1201-NtcipTraps MIB (Part 2 of 2: Trap Management) .....	172
Figure 18: Nodes Defined in NTCIP1201-RecMech MIB .....	173
Figure 19: Controller Class Diagram .....	174
Figure 20: Class Diagram of the Configuration Information .....	175
Figure 21: Class Diagram of the Transaction Service .....	176
Figure 22: Class Diagram of Time/DST Information.....	177
Figure 23: Class Diagram of Generic Schedule-Related Information .....	179
Figure 24: Class Diagram of the PMPP Data .....	180
Figure 25: Class Diagram for Auxiliary Input/Output (NTCIP 1203 v01) .....	181
Figure 26: Class Diagram for Auxiliary Input/Output Version 2 (NTCIP 1201 v02).....	181
Figure 27: Class Diagram of the NTCIP-Defined SNMP Data .....	182

Figure 28: Class Diagram of the SFMP Data .....	183
Figure 29: Class Diagram for Dynamic Object Management .....	184
Figure 30: Class Diagram of STMP Data .....	185
Figure 31: Class Diagram of the STMP Profile Data .....	186
Figure 32: Class Diagram of the Logical Names Data .....	187
Figure 33: Class Diagram of the Event Report Data .....	188
Figure 34: Class Diagram of the Community Name Security Data .....	189
Figure 35: Class Diagram of the Trap Management Data .....	190
Figure 36: Class Diagram of the Recording Mechanism (Version 1) Data .....	193

**TABLES**

	<b>Page</b>
Table 1: Configuration Information Mapping to Class Diagram .....	176
Table 2: Transaction Service Mapping to Class Diagram .....	176
Table 3: Time/DST Information Mapping to Class Diagram .....	178
Table 4: Time Information Mapping to Class Diagram .....	180
Table 5: PMPP Mapping to Class Diagram .....	181
Table 6: Auxiliary Input and Output Mapping to Class Diagram .....	182
Table 7: NTCIP-Defined SNMP Data Mapping to Class Diagram .....	182
Table 8: SFMP Data Mapping to Class Diagram .....	183
Table 9: Dynamic Object Management Mapping to Class Diagram .....	184
Table 10: STMP Data Mapping to Class Diagram .....	186
Table 11: STMP Profile Data Mapping to Class Diagram .....	187
Table 12: Logical Names Data Mapping to Class Diagram .....	187
Table 13: Event Report Data Mapping to Class Diagram .....	188
Table 14: Community Name Security Data Mapping to Class Diagram .....	189
Table 15: Trap Management Data Mapping to Class Diagram .....	190
Table 16: Recording Mechanism (Version 1) Data Mapping to Class Diagram .....	193

< This page is intentionally left blank. >



## Section 1 GENERAL

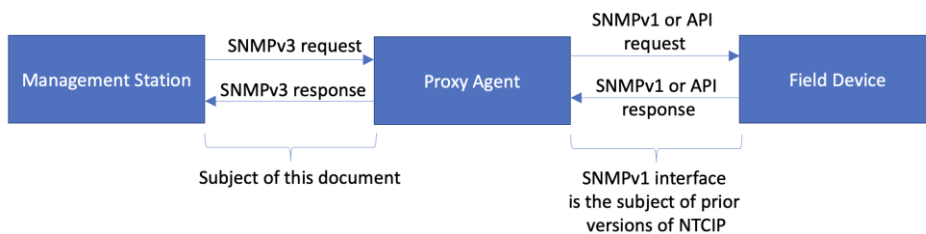
### 1.1 SCOPE

The NTCIP 1200 series defines standardized data elements that can be exchanged between a traffic management center (or any other management station) and a field device. This document identifies and defines data elements that can be supported by multiple types of devices (e.g., actuated signal controllers, dynamic message signs, connected vehicle roadside units).

**All data elements defined in this document have been deprecated.** Current data elements to fulfill a similar set of user needs can be found in ISO 26048-1.

The data defined in this standard was originally defined using the Structure of Management Information Version 1 (SMIv1) format, as defined in RFC 1212, and intended for implementations using the Simple Network Management Protocol Version 1 (SNMPv1). This document migrates the original data element definitions to use SMIv2, as defined by RFC 2578, to enable its unambiguous exchange when using SNMPv3.

This document does not apply to interfaces using SNMPv1 and the continued use of SNMPv1 is discouraged. New implementations should use data elements defined in ISO 26048-1. This document is intended to allow the data originally developed for SNMPv1 devices to be exchanged via an SNMPv3 interface. Specifically, this document is envisioned to address the interface between a management station and a proxy agent, as shown in Figure 1, and similar environments.



**Figure 1: Focus of this standard**

NOTE — Figure 1 reflects the expectation that management stations will be upgraded to SNMPv3 before field devices due to their limited numbers, but the data defined in this document applies for any SNMPv3 interface using the previously defined data included in this document, even those where the field device is upgraded first.

The management station is typically a traffic management center but could be any device acting as a SNMPv3 command generator (e.g., a field technician's laptop, a connected vehicle roadside unit). The field device is the device that is to be controlled and/or monitored by the management station. The proxy agent is responsible for providing secure communications between the traffic management center to a location very near the field device where a level of physical security can be provided. The interface between the proxy agent and the field device is not addressed by this standard; however, the expectation is that this interface is likely either:

1. an SNMPv1 interface (when the proxy agent is a separate physical device) or
2. an application programming interface (when the proxy agent is implemented as a software routine within the field device).

The manager communicates to the proxy agent using legacy versions of NTCIP data (as defined by this document) contained in SNMPv3 requests. If the request contains valid security credentials for the requested data, the proxy agent translates the SNMPv3 request into a format understood by the field device (e.g., SNMPv1) and forwards it to the older field device. The field device responds to the request and the proxy agent translates this response into SNMPv3. The proxy agent could be internal or external to the field device.

NOTE — Throughout the remainder of this document, the term "data element" is replaced with the term "object type" to align with SNMPv3 terminology.

## 1.2 REFERENCES

The following documents are referenced by this document. At the time of publication, the editions indicated were valid.

### 1.2.1 Normative References

Normative references contain provisions that, through reference in this text, constitute provisions of this document. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed.

AASHTO / ITE / NEMA NTCIP 8004 v03	Structure and Identification of Management Information (SMI) published (Pending)
IETF RFC 2578	Structure of Management Information Version 2 (SMIv2), April 1999
IETF RFC 2579	Textual Conventions for SMIv2, April 1999
IETF RFC 3418	Management Information Base (MIB) for the Simple Network Management Protocol (SNMP), December 2002
ISO/IEC/IEEE 24765:2017	Systems and software engineering – Vocabulary

### 1.2.2 Other References

Other references are included to provide a more complete understanding of this document and its relationship to other documents.

AASHTO / ITE / NEMA NTCIP 2301 v03	Simple Transportation Management Framework Application Profile (Pending)
AASHTO / ITE / NEMA NTCIP 8005 v02	Procedures for Creating Management Information Base (MIB) Files published (Pending)
IETF RFC 854	Telnet Protocol Specification
IETF RFC 3411	An Architecture for Describing Simple Network Management Protocol (SNMP) Management Framework, December 2002
IETF RFC 3412	Message Processing and Dispatching for the Simple Network Management Protocol (SNMP), December 2002
IETF RFC 3413	Simple Network Management Protocol (SNMP) Applications, December 2002
IETF RFC 3415	View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP), December 2002
IETF RFC 3416	Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP), December 2002
IETF RFC 3417	Transport Mappings for the Simple Network Management Protocol (SNMP)
IETF RFC 4181	Guidelines for Authors and Reviewers of MIB Documents, September 2005
IETF RFC 5591	Transport Security Model for the Simple Network Management Protocol (SNMP), June 2009

IETF RFC 6353	Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP), July 2011
IETF RFC 6933	Entity MIB (Version 4), May 2013
IETF RFC 9456	Updates to the TLS Transport Model for SNMP, November 2023
ISO 15784-2:2024	Intelligent transport systems — Data exchange involving roadside equipment communication — Part 2: Centre to field device communications using SNMP
ISO 21217:2020	Intelligent transport systems — Station and communications architecture
ISO 26048-1:–1	Intelligent transport systems – Field device SNMP data interface – Part 1: Global objects
NEMA TS 2-2016	Traffic Controller Assemblies with NTCIP Requirements

### 1.2.3 Contact Information

#### 1.2.3.1 IAB and IETF Documents

For Internet Architecture Board (IAB) and Internet Engineering Task Force documents, contact:

Internet Architecture Board (IAB)  
[www.rfc-editor.org](http://www.rfc-editor.org)

#### 1.2.3.2 NTCIP Documents

Copies of NTCIP documents may be obtained from:

NTCIP Coordinator  
National Electrical Manufacturers Association  
1300 N.17th Street, Suite 1752  
Rosslyn, Virginia 22209-3801  
[www.ntcip.org](http://www.ntcip.org)  
e-mail: [ntcip@nema.org](mailto:ntcip@nema.org)

Draft amendments, which are under discussion by the relevant NTCIP Working Group, and amendments recommended by the NTCIP Joint Committee are available.

#### 1.2.3.3 National Electrical Manufacturers Association (NEMA) Standards

Obtain NEMA standards from:

National Electrical Manufacturers Association  
1300 North 17th Street, Suite 1752  
Rosslyn, Virginia 22209  
[www.nema.org](http://www.nema.org)

## 1.3 GENERAL STATEMENTS

### 1.3.1 Use of SMIPv2

Prior versions of NTCIP 1201 were based on SNMPv1, which used the Structure of Management Information (SMI) Version 1 (v1) management information base (MIB) structure. SNMPv3 requires data to be defined in the SMI Version 2 (v2) format, as defined in RFC 2578. This document defines the official

<sup>1</sup> Publication of this document is expected in late 2024 or early 2025; drafts for commenting purposes are available upon request

translation of the MIBs previously defined in NTCIP 1201v03 and NTCIP 1103v03 into the SMIv2 format, per the rules of NTCIP 8004.

This translation provides an unambiguous, standardized definition of how to represent the previously defined data when using SNMPv3. For example, using this MIB, an SNMPv3 traffic management system can exchange data with an SNMPv1 device through a proxy agent, as shown in Figure 1. This arrangement allows for secure communications between the traffic management system and the proxy agent while steps can be made to physically secure the connection between the proxy agent and device. This is one way in which systems may choose to migrate to a more secure environment.

While the MIBs contained in this document can be used with SNMPv3, doing so does not resolve all known security issues.

### 1.3.2 See Other Documents for Additional Details

This document only provides MIBs translated into SMIv2 format. See NTCIP 1103 v03 and NTCIP 1201 v03 for the formal SMIv1 MIB definitions and additional details about their use. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

Commented [KV1]: Need to define the Github site

### 1.3.3 SNMPv3

The information defined in this document can theoretically be exchanged over any communications stack, but is designed to be exchanged using SNMPv3, as described in NTCIP 2301 v03, which requires support for the Transport Layer Security (TLS) Transport Model, as defined in RFC 6353 and RFC 9456.

### 1.3.4 Proxy Agents

In addition to a proxy agent serving as a translator for the object types defined in this document, the proxy agent shall allow the management of its security credentials. A proxy that supports the NTCIP 2301 v03 interface shall support the object types identified in NTCIP 2301 v03 for this purpose.

NOTE — For example, a management station can send an SNMPv3 request to the proxy that the proxy will translate and forward to the field device. A management station can separately send an SNMPv3 request to the proxy to configure the proxy's security credentials (this request is not forwarded to the field device). The intended destination of each request is determined based on the context specified within the SNMPv3 request.

### 1.3.5 Superseding and Supplanting Objects

Many of the object types defined in this document identify superseding or supplanting object types by using a "<Superseded by>" or "<Supplanted by>" clause. Supplanting object type(s) provide similar functionality while superseding objects provide a new interface for the same device capability.

For example, the eventConfigMode identifies the fdCondTriggerMode of ISO 26048-1 as a supplanting object. In this case, both objects provide similar functionality (i.e., identifying a type of condition to use to trigger an event), but setting an instance of eventConfigMode does not affect any instance of fdCondTriggerMode in an implementation that supports both objects.

By contrast, globalTime identifies fdClockUtcDate and fdClockUtcTime as superseding objects. In other words, if a device supports both globalTime and fdClockUtcTime, setting globalTime will affect the time reported by fdClockUtcTime. The core device capability is the same (e.g., keeping time), but there is a new interface that provides greater range and resolution.

### 1.3.6 Agent Capability Statements

Prior versions of NTCIP granted implementors the right to edit standardized MIB files to reflect the capabilities offered by their implementation. For example, in NTCIP 1103 v03, eventConfigID was defined



with a syntax of "INTEGER (0..65535)". Implementors were allowed to edit this file to subrange the syntax of eventConfigID to correspond with the range supported by their implementation. However, this presented its own challenges for managers trying to load multiple versions of the same MIB file.

**Implementors are not allowed to edit standardized SMiv2 MIB files.** Instead, implementations should be provided with a file containing an AGENT-CAPABILITIES macro, as defined in RFC 2580. The AGENT-CAPABILITIES macro allows implementors to describe the variances in their implementation in a standardized way so that managers can quickly discover capabilities of each connected implementation.

#### 1.4 TERMS

For the purposes of this document, the following terms and definitions apply. Terms not defined here are in accordance with their definitions in NTCIP 8004. Systems and software engineering terms not defined here are used in accordance with their definitions in ISO/IEC/IEEE 24765. English words not defined here or in ISO/IEC/IEEE 24765 are used in accordance with their definitions in Webster's New Collegiate Dictionary.

ASC	actuated signal controller
component	central system, field device, etc., that supports NTCIP
control object	writeable object types used to request real-time activation of a feature of a device NOTE—In some cases, control objects can also be used to report status.
cyclic redundancy check (CRC)	polynomial algorithm performed on a specified range of data resulting in a 16 or 32 bit value
DST	daylight saving time
feature	capability of a component
interchangeable	condition that exists when two or more items possess such functional and physical characteristics as to be equivalent in performance and durability, and are capable of being exchanged one for the other without alteration of the items themselves, or adjoining items, except for adjustment, and without selection for fit and performance. NOTE—See National Telecommunications and Information Administration, U.S. Department of Commerce.
interrelated parameter object	writeable object type used to configure an SNMP agent where the parameter has sufficient interrelationships with other object types to typically require multiple SNMP set operations using multiple SetRequest messages or a complex validation check that might consume more time than is reasonable for a traditional SNMP response
parameter object	writeable object type used to configure the SNMP agent where the parameter can be set and validated using a single SNMP set operation
Point-to-Multi-Point Protocol (PMPP)	transportation specific subnetwork layer protocol that was designed to enable communication between multiple devices on the same communications line/channel NOTE—The protocol is considered a legacy design because the communication links for which it was designed are unlikely to be able to support the data demands of a secure environment.
Simple Transportation Management Protocol (STMP)	Part of the legacy Transportation Management Protocols of the NTCIP effort. NOTE—See NTCIP 1103. STMP provided a simple and bandwidth efficient mechanism to communicate with field devices. NOTE—The protocol is considered a legacy design because the communication links for which it was designed are unlikely to be able to support the data demands of a secure environment.
status object	read-only object that reports a condition monitored by the SNMP agent

## Section 2 DEPRECATED GLOBAL MIB

The text provided from Section 2.1 through the end of Section 2 (except the headings) constitutes the standard NTCIP1201-Global MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 2.1 NTCIP HEADER

```
NTCIP1201-Global DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Integer32, Unsigned32,
    zeroDotZero
    FROM SNMPv2-SMI
    -- RFC 2578
AutonomousType, VariablePointer
    FROM SNMPv2-TC
    -- RFC 2579
devices, profiles
    FROM NTCIP8004-Transportation;

global MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of various globally
    applicable objects that were previously defined in NTCIP 1201 v03.
    *** All objects in this MIB have been deprecated. ***"
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8"
REVISION "20230701145436Z"
DESCRIPTION
    "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "200803240000Z"
DESCRIPTION
    "NTCIP 1201 v03 - Added DST table objects. Created standalone MIBs for the
    main portion of the MIB and the 2 different versions of AuxIO objects."
REVISION "200409270000Z"
DESCRIPTION
    "NTCIP 1201 v02 - Removed global report, logicalNameTranslation and
    communityName nodes as they are now included in NTCIP 1103v0124. Removed
    state transition diagram form 2.3.1. Changed 'FROM NTCIP8004-A' to 'FROM
    NTCIP8004-A-2004' and restructured associated imports. Changed status of
    moduleNumber, timeBaseScheduleNumber, dayPlanNumber, dayPlanEventNumber,
    eventClassNumber, eventConfigID and their associated Entry to mandatory
    from optional to eliminate incompatible status errors."
REVISION "199810010000Z"
```

DESCRIPTION  
"Amendment 1 to NTCIP 1201 v01 (a.k.a. Amendment 1 to NEMA TS 3.4 and NTCIP 1201:1998): Editorial corrections. Revised DST object."

REVISION "199610010000Z"

DESCRIPTION  
"Original version approved as NEMA TS 3.4 (a.k.a. NTCIP 1201 v01)."

::= { devices 6 }

## 2.2 OBJECT IDENTITIES

### 2.2.1 Global Configuration Node

globalConfiguration OBJECT-IDENTITY

STATUS deprecated

DESCRIPTION  
"<Definition> This node is an identifier used to group all objects for support of configuration functions that are common to most device types.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1"

::= { global 1 }

## 2.3 OBJECTS

### 2.3.1 Global Set ID Parameter

globalSetIDParameter OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION  
"<Definition> Specifies a relatively unique ID (e.g., this could be a counter, a check-sum, etc.) for all user-changeable parameters of the particular device-type currently implemented in the device. Often this ID is calculated using a CRC algorithm. This value shall be calculated when a change of any parameter object has occurred. The value reported by this object shall not change unless there has been a change in the static data since the last request. If the actual objects, which are to be included to create this object value, are not defined in the actual device-level standard such as 1202 or 1203, then the general guidance is to include all configuration objects that are stored in a type of memory that survives power outages. A management station can use this object to detect any change in the parameter objects by monitoring this value after it has established a baseline.

<Superseded by> FIELD-DEVICE-MAIN-MIB.fdConfigurationID (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.1"

::= { globalConfiguration 1 }

### 2.3.2 Maximum Modules Parameter

globalMaxModules OBJECT-TYPE

SYNTAX Integer32 (1..255)

UNITS "modules"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION  
"<Definition>The number of rows that are listed in the globalModuleTable.

<Informative> The module table has been replaced with features from other standards so this object is no longer needed.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.2"

::= { globalConfiguration 2 }

### 2.3.3 Module Table

moduleTable OBJECT-TYPE

SYNTAX SEQUENCE OF ModuleEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> A table containing information regarding manufacturer of software and hardware and the associated module models and version numbers as well as an indicator if the module is hardware or software related. The number of rows in this table shall equal the value of the globalMaxModules object.

<Informative> The module table has been replaced with features from other standards.

<Table Type> static

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.3"

::= { globalConfiguration 3 }

moduleEntry OBJECT-TYPE

SYNTAX ModuleEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> This object defines an entry in the module table.

<Informative> The module table has been replaced with features from other standards.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.3.1"

INDEX { moduleNumber }

::= { moduleTable 1 }

ModuleEntry ::= SEQUENCE {

moduleNumber INTEGER,  
moduleDeviceNode AutonomousType,  
moduleMake OCTET STRING,  
moduleModel OCTET STRING,  
moduleVersion OCTET STRING,  
moduleType INTEGER}

#### 2.3.3.1 Module Number Parameter

moduleNumber OBJECT-TYPE

SYNTAX Integer32 (1..255)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> This object contains the row number (1..255) within this table for the associated module.

<Informative> The module table has been replaced with features from other standards so a row number is no longer needed.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.3.1.1"

::= { moduleEntry 1 }

### 2.3.3.2 Module Device Node Parameter

moduleDeviceNode OBJECT-TYPE

SYNTAX AutonomousType

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>This object contains the device node number of the device-type, e.g., an ASC signal controller would have an OID of 1.3.6.1.4.1.1206.4.2.1.

<Supplanted by> SNMPv2-MIB.sysORID (from RFC 3418)

<Informative> The intent of this object was to provide an indication of the type of data within the device. This is achieved at a much finer level of detail with the replacement object, which is a required object per SNMPv3 standards.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.3.1.2"

::= { moduleEntry 2 }

### 2.3.3.3 Module Make Parameter

moduleMake OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>This object specifies the manufacturer of the associated module. A null-string shall be transmitted if this object has no entry.

<Supplanted by> ENTITY-MIB.entPhysicalMfgName (RFC 6933)

<Informative> The entPhysicalTable allows (but does not require) an implementation to show relationships among components thereby providing more meaningful information for devices that need to provide this level of detail. It is also defined as a multi-lingual string that will allow managers to automatically display the text in the appropriate format.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.3.1.3"

::= { moduleEntry 3 }

### 2.3.3.4 Module Model Parameter

moduleModel OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>This object specifies the model number (hardware) or firmware reference (software) of the associated module. A null-string shall be transmitted if this object has no entry.

<Supplanted by> ENTITY-MIB.entPhysicalModelName (RFC 6933)

<Informative> The entPhysicalTable allows (but does not require) an implementation to show relationships among components thereby providing

more meaningful information for devices that need to provide this level of detail. It is also defined as a multi-lingual string that will allow managers to automatically display the text in the appropriate format.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.3.1.4"

::= { moduleEntry 4 }

### 2.3.3.5 Module Version Parameter

moduleVersion OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>This object specifies the version of the associated module. If the moduleType has a value of software, the value of this object shall include the date on which the software was released as a string in the form of YYYYMMDD, it shall be followed by a space, a hyphen, another space, the lower-case letter 'v', followed by a version or configuration number. Preceding zeros shall be required for the date. For example, version 7.03.02 of the software released on July 5, 2002 would be presented as 20020705: v7.03.02

A null-string shall be transmitted if this object has no entry.

<Supplanted by> ENTITY-MIB.entPhysicalHardwareRev & ENTITY-MIB.entPhysicalSoftwareRev (RFC 6933)

<Informative> The entPhysicalTable allows (but does not require) an implementation to show relationships among components thereby providing more meaningful information for devices that need to provide this level of detail. It is also defined as a multi-lingual string that will allow managers to automatically display the text in the appropriate format.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.3.1.5"

::= { moduleEntry 5 }

### 2.3.3.6 Module Type Parameter

moduleType OBJECT-TYPE

SYNTAX INTEGER { other (1),  
                  hardware (2),  
                  software (3) }

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> This object specifies whether the associated module is a hardware or software module.

<Informative> The ENTITY-MIB.entPhysicalTable (RFC 6933) allows the definition of physical entities that contain hardware and/or software and allows associations among them.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.3.1.6"

::= { moduleEntry 6 }

### 2.3.4 Base Standards Parameter

controllerBaseStandards OBJECT-TYPE

SYNTAX OCTET STRING (SIZE (0..256))

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> For use in this object, an ASCII string that shall identify all of the standard document numbers that define or reference MIBs upon which the device is based. Where applicable, profiles shall be referenced rather than the base standards.

<Format> The version string shall be constructed as follows:  
The acronym of the standards development organization (or other body) that developed and approved the standard; a space; the standards document number; a colon; and the documents version number as designated by the standards development organization (or other body). Separate entries in the list of standards shall be separated by a carriage return (0x0d) and line feed (0x0a).

In the case of NTCIP documents prior to formal approval, the version number shall be the version number in the form of lower case 'v' followed by the major version followed by a period followed by the minor revision. In the case of approved NTCIP standards, the publication year shall precede the version number. In the case of amended NTCIP standards, the version number shall be replaced by the four digit year of publication of the published standard followed by the upper case letter 'A', followed by the amendment number.

For example, a message sign may have the following value for this object:  
NTCIP 1201:v02.19  
NTCIP 1203:1997A1  
NTCIP 2101:2001 v01.19  
NTCIP 2103:v01.13  
NTCIP 2201:v01.14  
NTCIP 2301:2001 v01.08

<Supplanted by> SNMPv2-MIB.sysORID (from RFC 3418)

<Informative> sysORID provides more refined information using OIDs to identify compliance statements that can be processed by computers without having to parse strings.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.4"

::= { globalConfiguration 4 }

## 2.4 GLOBAL DATABASE MANAGEMENT NODE

globalDBManagement OBJECT-IDENTITY

STATUS deprecated

DESCRIPTION

"This node is an identifier used to group those objects used to manage a transaction.

A transaction is a SET of one or more database parameters that have inter-relationships with other database parameters, as such a SET for any one of these objects must be validated against a set of consistency checks and may potentially require the setting of a large number of objects simultaneously. Thus, the mode described by these objects allow for such a large database download.

Any device standard that allows this feature shall define which objects are database parameters versus status or control objects.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2"

::= { global 2 }

#### 2.4.1 Database Creation Transaction

dbCreateTransaction OBJECT-TYPE  
SYNTAX INTEGER { normal (1),  
transaction (2),  
verify (3),  
done (6) }

MAX-ACCESS read-write

STATUS deprecated

##### DESCRIPTION

"<Definition> This object provides transaction control for device configuration. The transaction mode changes the behavior of the agent to force buffering of parameter objects until all related parameter objects have been modified. In the normal mode, SET operations to any parameter object shall either be stored in a device's database immediately with no regard to whether other changes will be made or be rejected (as defined in the device-specific Information Profile). In the transaction mode, SET operations to any parameter object shall be buffered until a verify state performs a consistency check. When the consistency check completes, the device automatically transitions to the done state where a normal or transaction command may be issued.

A parameter object is a user-provided piece of setup information (or it may be defined in an information profile) that is necessary for the proper operation of a device. It is static in nature in that the agent would never change it without direction from the management station. For example, an object that defines a default mode of operation would be a parameter object. An object that indicates the current state of the device would not be a parameter object.

<Format> The states and commands are defined as:

**NORMAL:** SET operations behave as normal SETs and shall have an immediate effect on the value of any parameter objects used by the device if none of the objects contained in the operation require the use of the transaction mode (as defined in the device-specific Information Profile). A SET operation containing any transaction object (i.e., a parameter object that requires the use of transaction mode) shall result in a genErr. This is the default state of this object.

The only command that may be written to dbCreateTransaction while in this state is TRANSACTION. Any other values written to this object in this state shall result in an error response of 'badValue'.

**TRANSACTION:** A SET operation of one or more parameter objects that use the same community name as used in the request for the TRANSACTION state are buffered by the agent device for later consistency checks and a normal response is returned. A SET operation of one or more parameter objects using different community names shall result in a genErr with the index set to zero. A SET operation without a community name field (e.g., an STMP operation) shall be buffered by the agent device for later consistency checks and a normal response is returned. Standard SYNTAX checking shall take place at the time of the SET operation. A transaction may consist of multiple SET operations over multiple frames.

A SET operation for one or more non-parameter objects shall be processed as normal even if it uses another community name, except for this (i.e., the dbCreateTransaction) object.

A SET operation containing both parameter and non-parameter objects shall be processed in full according to these two rules. Thus, if it contains the same community name as used in the request for the TRANSACTION state, the



non-parameter objects shall be stored immediately while the parameter objects shall be buffered. If it uses a different community name, the entire request will be rejected and a genErr with an index of zero shall be returned.

GET operations on any object shall return the values of the data stored in the controller and shall ignore any values contained in the buffer. Any valid community name may read this (dbCreateTransaction) object when in this state, but only the community name used to command the object to the transaction mode and the administrator community name can set this object. A set from any other community name shall result in a genErr with an index of zero. The only commands that can be written to dbCreateTransaction while in this state are VERIFY and NORMAL. A VERIFY command will change the state to VERIFY. If a NORMAL command is received, all buffered data is discarded and the state is returned to NORMAL. Any other values written to this object when in this state shall result in an error response of 'badValue'.

VERIFY: Specific parameter objects are checked for consistency. When consistency checks are complete the device will automatically advance to the DONE state.

The state of dbCreateTransaction cannot be changed when in the VERIFY state. Any values written to this object in this state shall result in an error response of 'badValue'.

The consistency check analyzes certain critical objects 'in context' and treats them as an interrelated whole rather than separate non-related data items. The consistency check rules are not defined in NTCIP 1201 v03, since these are device and implementation specific. Where applicable, the consistency check rules are defined in application specific object definition standards. A specific implementation may add additional checks beyond those defined in NTCIP standards.

A SET operation containing any parameter objects while in the VERIFY state shall result in a genErr with the index set to zero.

DONE: This state is entered automatically once consistency checks have completed in the VERIFY mode. The value of dbVerifyStatus and dbVerifyError indicate whether the consistency check found any errors.

A SET operation containing any parameter objects while in the DONE state shall result in a genErr with the index set to zero.

Any valid community name may read this (dbCreateTransaction) object when in this state, but only the community name used to command the object to the transaction mode and the administrator community name can set this object. A set from any other community name shall result in a genErr with an index of zero. The only commands that can be written to dbCreateTransaction while in this state are NORMAL and TRANSACTION. Any other values written to this object in this state shall result in an error response of 'badValue'. If a NORMAL command is issued and dbVerifyStatus indicates doneWithNoError, the buffered data is transferred to the device memory and the state is returned to NORMAL. If a NORMAL command is issued and dbVerifyStatus indicates something other than doneWithNoError then the buffered data is discarded and the state is returned to NORMAL. If a TRANSACTION command is issued, regardless of dbVerifyStatus, no action takes place (the buffered data is not changed) and the TRANSACTION state is re-entered.

```
          |          COMMANDED STATE (9)
          |-----|
CURRENT STATE | transaction | verify | normal | done
```

```

-----+-----+-----+-----+-----
normal      | transaction (1) | normal (2) | normal (2) | normal (2)
transaction | transaction (2) | verify (3) | normal (4) | transaction (2)
verify (7)  | verify (2)      | verify (2) | verify (2) | verify (2)
done (8)    | transaction (5) | done (2)   | normal (6) | done (2)

```

Operational procedures and error responses:

- (1) Once a copy of all parameter objects is placed in a buffer, the state is changed to transaction and error response indicates noError. If the operation fails, the state remains the same and error response indicates genErr.
  - (2) No action takes place, the state remains the same, but response indicates badValue.
  - (3) The state is changed to verify, a consistency check is started, and response indicates noError. Once the consistency check is completed, the state automatically changes to done.
  - (4) The buffered copy of all parameter objects is discarded, the state is changed to normal, and response indicates noError.
  - (5) The buffered copy of all parameter objects is not changed or reloaded, the state is changed to transaction, and response indicates noError.
  - (6) If dbVerifyStatus indicates doneWithNoError, then the copy of all parameter objects is transferred to memory, the state is changed to normal and response indicates noError. If dbVerifyStatus indicates doneWithError then the buffered data is discarded, the state is changed to NORMAL, and response indicates noError.
  - (7) The state automatically changes to done when the consistency check completes.
  - (8) dbVerifyStatus and dbVerifyError are only valid in this state.
  - (9) All SET operations on this (dbCreateTransaction) parameter shall be made using a protocol that uses a community name, or equivalent field (e.g., SNMP).
- <Supplanted by> ISO26048-1-Transaction.fdTransactionMode
- <Informative> The original version of this object referenced SNMPv1 error codes, community names, and an administrator name; the V2 object updates these details to be SNMPv3 specific.
- <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.1"

```

DEFVAL      { normal }
::= { globalDBManagement 1 }

```

**2.4.2 Database Error Type Parameter**

```

dbErrorType OBJECT-TYPE
SYNTAX      INTEGER { tooBig (1),
                    noSuchName (2),
                    badValue (3),
                    readOnly (4),
                    genError (5),

```

```
        updateError (6),
        noError (7) }
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
  "<Definition> This object returns the current error status of the
  transaction. The value of this object is only valid when the
  dbCreateTransaction object is in the Done or Error state.

  <Informative> This object was deprecated in NTCIP 1201 v02

  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.2"
 ::= { globalDBManagement 2 }
```

#### 2.4.3 Database Error ID Parameter

```
dbErrorID OBJECT-TYPE
SYNTAX OBJECT IDENTIFIER
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
  "<Definition> This object contains the object identifier of the first
  object in the transaction buffer that caused an error while
  dbCreateTransaction object was in the Verifying or Updating state. The
  value of this object is only valid when the dbCreateTransaction object is
  in the Error state. It is undefined when the dbCreateTransaction object is
  in other states.

  <Informative> This object was deprecated in NTCIP 1201 v02

  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.3"
 ::= { globalDBManagement 3 }
```

#### 2.4.4 Database Transaction ID Parameter

```
dbTransactionID OBJECT-TYPE
SYNTAX Integer32 (0..255)
MAX-ACCESS read-write
STATUS deprecated
DESCRIPTION
  "<Definition> This object contains the transaction ID value that is to be
  contained in all SET operation writes while the dbCreateTransaction object
  is not in the Normal state. During transaction operations every SET command
  shall begin with a write to this object with the current value of this
  object. If a SET operation is performed without writing to this object, or
  with a value that does not match the current value, then an error response
  of 'genError' shall be returned. This mechanism is used to determine that
  the same management station that started the transaction is performing the
  SET operations that are being buffered or modifying the state of
  dbCreateTransaction.

  <Informative> This object was deprecated in NTCIP 1201 v02

  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.4"
 ::= { globalDBManagement 4 }
```

#### 2.4.5 Database Make ID Parameter

```
dbMakeID OBJECT-TYPE
SYNTAX Integer32(0..255)
MAX-ACCESS read-only
STATUS deprecated
```

DESCRIPTION

"<Definition> This object is used to create unique transaction ID's for management stations to use when starting transactions using the dbCreateTransaction object. This object will be incremented by one every time it is read, so that different values will be returned for each read. Management stations wishing to start a transaction should first read the dbCreateTransaction object to verify that it is in the Normal state. If so then the management shall GET dbMakeID to obtain a transaction ID to use, then SET dbCreateTransaction to startCmd and dbTransactionID to the value just received. If the response to the SET operation is 'noError' then the management station has started a transaction. If the response to the SET operation is 'genError' then the management station should read the dbCreateTransaction and dbTransactionID objects to ensure that the error was not due to a communications retry. If the dbCreateTransaction is in the Transaction state, and the dbTransactionID is the same value returned by the read of this object, then the management station is the owner of the transaction. If the dbTransactionID does not match the value originally returned by this object, then the management station is not the owner of the transaction and must wait until the dbCreateTransaction object returns to the Normal state before attempting to start the transaction.

<Informative> This object was deprecated in NTCIP 1201 v02

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.5"

::= { globalDBManagement 5 }

#### 2.4.6 Database Verify Status Parameter

dbVerifyStatus OBJECT-TYPE

SYNTAX INTEGER { notDone (1),  
doneWithError (2),  
doneWithNoError (3) }

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> This object indicates the current status of verify (consistency checking) processing. The value of this object is only meaningful when the dbCreateTransaction object is in the Verify or Done state.

<Supplanted by> ISO26048-1-Transaction.fidTransactionStatus

<Informative> The V2 object adds support for error codes related to Implementing the database after consistency checks have passed.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.6"

::= { globalDBManagement 6 }

#### 2.4.7 Database Verify Error Parameter

dbVerifyError OBJECT-TYPE

SYNTAX OCTET STRING (SIZE (0..255))

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> This object contains a textual description of or a reference to an error that was found by the verify (consistency checking) processing. The value of this object is only meaningful when the dbCreateTransaction object is in the Done state and the dbVerifyStatus object is in the doneWithError state.

<Supplanted by> ISO26048-1-Transaction.fdTransactionError  
<Informative> The V2 object revises the syntax so that it can be automatically recognized as a multi-lingual text string (i.e., SnmpAdminString).  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.7"  
 ::= { globalDBManagement 7 }

## 2.5 GLOBAL TIME MANAGEMENT NODE

globalTimeManagement OBJECT-IDENTITY  
STATUS deprecated  
DESCRIPTION  
"This node is an identifier used to organize all objects for support of time-related functions that are common to most device types.  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3"  
 ::= { global 3 }

### 2.5.1 Global Time Parameter

globalTime OBJECT-TYPE  
SYNTAX Unsigned32  
UNITS "seconds"  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
"<Definition> The number of seconds since the epoch of 00:00:00 (midnight) January 1, 1970 UTC (a.k.a. Zulu or GMT).  
<Superseded by> ISO26048-1-Clock.fdClockUtcDate & ISO26048-1-Clock.fdClockUtcTime (ISO 26048-1)  
<Informative> The original specification defined this parameter using a Counter; however, by convention, Counter objects are not supposed to be writable in SNMPv1 and SNMPv3 prohibits writable Counter objects. Therefore, when presenting this object in SNMPv3, it is encoded as an Unsigned32 rather than a Counter32; proxy agents will need to address this encoding change within their implementation. In addition, the object has a potential rollover problem in 2038 and there are NTCIP needs to support millisecond-level time information. The superseding objects address all of these issues.  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.1"  
DEFVAL { 0 }  
 ::= { globalTimeManagement 1 }

### 2.5.2 Global Daylight Saving Parameter

globalDaylightSaving OBJECT-TYPE  
SYNTAX INTEGER { other (1),  
 disableDST (2),  
 enableUSDST (3),  
 enableEuropeDST (4),  
 enableAustraliaDST (5),  
 enableTasmaniaDST (6),  
 enableEgyptDST (7),  
 enableNamibiaDST (8),  
 enableIraqDST (9),

```
enableMangoliaDST (10),  
enableIranDST (11),  
enableFijiDST (12),  
enableNewZealandDST (13),  
enableTongaDST (14),  
enableCubaDST (15),  
enableBrazilDST (16),  
enableChileDST (17),  
enableFalklandsDST (18),  
enableParaguayDST (19),  
enableDaylightSavingNode (20) }
```

MAX-ACCESS read-write  
STATUS deprecated

DESCRIPTION

"<Definition>This object specifies whether the daylight saving time (DST) is enabled, disabled or some other form of DST is active.

<Format>

other - DST adjustments by a mechanism not defined within this standard.  
disabledDST - DST clock adjustments shall NOT occur.  
enableUSDST - DST shall begin the first Sunday in April and shall end the last Sunday of October. All changes of time occur at 2:00 AM. (This is the pre-2007 DST settings for the USA.)  
enableEuropeDST - DST shall start the last Sunday of March at 2:00 AM and ends the last Sunday of October at 3:00 AM.  
enableAustraliaDST - DST shall start the last Sunday in October at 2:00 AM and ends the last Sunday in March at 2:00 AM.  
enableTasmaniaDST - DST shall start the first Sunday in October at 2 a.m. and ends the last Sunday in March at 3 a.m.  
enableEgyptDST: DST shall start the last Friday in April and end the last Thursday in September.  
enableNamibiaDST: DST shall start the first Sunday in September and end the first Sunday in April.  
enableIraqDST: DST shall start on April 1 and end on October 1.  
enableMongoliaDST: DST shall start the last Sunday in March and end the last Sunday in September.  
enableIranDST: DST shall start the first day of Farvardin and end the first day of Mehr  
enableFijiDST: DST shall start the first Sunday in November and end the last Sunday in February.  
enableNewZealandDST: DST shall start the first Sunday in October and end the first Sunday on or after March 5th.  
enableTongaDST: DST shall start the first Saturday in October and end the first Saturday on or after April 15th.  
enableCubaDST: DST shall start April 1st and end last Sunday in October.  
enableBrazilDST: DST shall start the first Sunday in October and end the last Sunday in February.  
enableChileDST: DST shall start the first Sunday on or after October 9th and end the first Sunday on or after March 9th.  
enableFalklandsDST: DST shall start the first Sunday on or after September 8th and end the first Sunday on or after April 8th.  
enableParaguayDST: DST shall start the first Sunday in October and end the last Saturday in February.  
enableDaylightSavingNode: DST operation is controlled by objects located under the daylightSavingNode.

<Superseded by> DST table

<Informative> This object was deprecated in NTCIP 1201 v03. This object is maintained for backward compatibility and it is envisioned that only the following values are supported with all other values are deprecated:

- other (1),
- disableDST (2),
- enableDaylightSavingNode (20)

Users should ensure that the values of globalDaylightSaving and the entries in the new DST Table are coordinated. The globalDaylightSaving object is intended to be used to enable and disable DST and should not be set to the value '20', enableDaylightSavingNode until after the dstTable entries have been fully configured. Further, the globalDaylightSaving object supersedes the settings in the DST Table.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.2"

```
REFERENCE "NEMA TS 2 Clause 3.8.2;  
http://fatty.law.cornell.edu/uscode/15/260a.html;  
http://webexhibits.org/daylightsaving/g.html"  
DEFVAL { enableDaylightSavingNode }  
::= { globalTimeManagement 2 }
```

## 2.6 TIMEBASE EVENT SCHEDULER NODE

timebase OBJECT-IDENTITY

STATUS deprecated

DESCRIPTION

"This node is an identifier used to organize the main objects for event scheduling. Device type-specific objects (tables) pointed to are defined within the appropriate MIB.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3"

```
::= { globalTimeManagement 3 }
```

### 2.6.1 Maximum Number of Time Base Schedule Entries Parameter

maxTimeBaseScheduleEntries OBJECT-TYPE

SYNTAX Integer32(1..65535)

UNITS "TimeBaseScheduleEntry"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The value of this object specifies the maximum number of different entries supported by the device as shown by the number of rows in the timeBaseScheduleTable.

<Informative> The timeBaseScheduleTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.1"

```
::= { timebase 1 }
```

### 2.6.2 Time Base Schedule Table

timeBaseScheduleTable OBJECT-TYPE

SYNTAX SEQUENCE OF TimeBaseScheduleEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> A table containing the time base schedule parameters for the device. The number of rows in this table shall be equal to the maxTimeBaseScheduleEntries object. The table references the appropriate day plan for the device. The plan is determined by comparing the current month (MONTH), day of week (DOW) and date of month (DOM) to the appropriate fields. The settings for MONTH, DOW and DOM are connected with a logical AND. To determine which timebased event to select, determine the event which has the most specific date specified. Select the more specific event based on their MONTH settings; if the same, select the most specific DOM; if that is still the same, select the most specific DOW; if still the same, the first occurrence within the time base event table shall be selected. 'More specific' means the least number of bits set within an object. All entries in Time Base Schedule Table are expressed in local time and date. A row in the table may be deactivated by setting the Month, Day, Date, or DayPlan parameters to zero (0)

<Table Type> static

<Supplanted by> DAY-PLAN-MIB.fdDayPlanScheduleTable (ISO 26048-1)

<Informative> The original timeBaseScheuleDate object was defined as INTEGER (0..4294967295); however, by convention, SNMPv1 does not allow unsigned 32-bit integers and SNMPv3 prohibits them. The revise table addresses this issue and is also treated using trigger logic so that it can be used to activate a day plan (as originally envisioned) and/or to call other actions (e.g., creating a log entry).

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.2"

::= { timebase 2 }

timeBaseScheduleEntry OBJECT-TYPE

SYNTAX TimeBaseScheduleEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> Event Parameters for the time based schedule programming of the device.

<Superseded by> ISO26048-1-DayPlan.fdDayPlanScheduleEntry (ISO 26048-1)

<Informative> The replacement object extends the row to support a description and supports full SNMPv3 row management.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.2.1"

INDEX { timeBaseScheduleNumber }

::= { timeBaseScheduleTable 1 }

```
TimeBaseScheduleEntry ::= SEQUENCE {
    timeBaseScheduleNumber Integer32,
    timeBaseScheduleMonth Integer32,
    timeBaseScheduleDay Integer32,
    timeBaseScheduleDate Integer32,
    timeBaseScheduleDayPlan Integer32}
```

### 2.6.2.1 Time Base Schedule Number Parameter

timeBaseScheduleNumber OBJECT-TYPE

SYNTAX Integer32(1..65535 )

MAX-ACCESS read-only



STATUS deprecated  
DESCRIPTION  
"<Definition> The time base schedule number for objects in this row. The value of this object shall not exceed the value of the maxTimeBaseScheduleEntries object. The activation of a scheduled entry shall occur whenever allowed by all other objects within this table.  
<Supplanted by> ISO26048-1-DayPlan.fdayPlanScheduleNumber (ISO 26048-1)  
<Informative> The replacement object extends the range to an Unsigned32  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.2.1.1"  
 ::= { timeBaseScheduleEntry 1 }

### 2.6.2.2 Time Base Schedule Month of Year Parameter

timeBaseScheduleMonth OBJECT-TYPE  
SYNTAX Integer32(0..65535)  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
"<Definition> The Month(s) Of the Year that the schedule entry shall be allowed.  
<Format> Each bit represents a specific month. If the bit is set to one (1), then the scheduled entry shall be allowed during the associated month. If the bit is set to zero (0), then the scheduled entry shall not be allowed during the associated month. The bits are defined as:

Bit	Month of Year
0	Reserved
1	January
2	February
3	March
4	April
5	May
6	June
7	July
8	August
9	September
10	October
11	November
12	December
13 - 15	Reserved

Thus, a value of six (6) would indicate that the entry would only be allowed during the months of January and February. A value of zero (0) shall indicate that this row has been disabled.

<Supplanted by> ISO26048-1-DayPlan.fdDayPlanScheduleMonth (ISO 26048-1)  
<Informative> The replacement object is defined using the BITS structure.  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.2.1.2"

::= { timeBaseScheduleEntry 2 }

### 2.6.2.3 Time Base Schedule Day of Week Parameter

timeBaseScheduleDay OBJECT-TYPE

SYNTAX Integer32(0..255)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> The Day(s) Of Week that the schedule entry shall be allowed.

<Format> Each bit represents a specific day of the week. If the bit is set to one (1), then the scheduled entry shall be allowed during the associated DOW. If the bit is set to zero (0), then the scheduled entry shall not be allowed during the associated DOW. The bits are defined as:

Bit	Day of Week
0	Reserved ('Holiday', not defined by this standard)
1	Sunday
2	Monday
3	Tuesday
4	Wednesday
5	Thursday
6	Friday
7	Saturday

Thus, a value of six (6) would indicate that the entry would only be allowed on Sundays and Mondays. A value of zero (0) shall indicate that this row has been disabled.

<Supplanted by> ISO26048-1-DayPlan.fdDayPlanScheduleDayOfWeek (ISO 26048-1)

<Informative> The replacement object is defined using the BITS structure

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.2.1.3"

::= { timeBaseScheduleEntry 3 }

### 2.6.2.4 Time Base Schedule Date Parameter

timeBaseScheduleDate OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> The Day(s) Of a Month that the schedule entry shall be allowed.

<Format> Each bit represents a specific date of the month. If the bit is set to one (1), then the scheduled entry shall be allowed during the

associated date. If the bit is set to zero (0), then the scheduled entry shall not be allowed during the associated date. The bits are defined as:

Bit	Day Number
0	Reserved
1	Day 1
2	Day 2
31	Day 31

Thus, a value of six (6) would indicate that the entry would only be allowed on the first and second of the allowed months. A value of zero (0) shall indicate that this row has been disabled.

<Supplanted by> DAY-PLAN- MIB.fdayPlanScheduleDayOfMonth (ISO 26048-1)  
<Informative> The original specification defined this parameter using INTEGER (0..4294967295); however, by convention, INTEGERS are limited to the range (- 2147483648..2147483647) in SNMPv1 and SNMPv3 enforces this limitation. Therefore, when presenting this object in SNMPv3, it is encoded as an Unsigned32 rather than an INTEGER (0..4294967295); proxy agents will need to address this encoding change within their implementation. The replacement object in the DAY-PLAN-MIB implements this concept using the BITS structure.  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.2.1.4"  
 ::= { timeBaseScheduleEntry 4 }

### 2.6.2.5 Time Base Schedule Day Plan Parameter

timeBaseScheduleDayPlan OBJECT-TYPE  
SYNTAX Integer32(0..255)  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object specifies what Plan number shall be associated with this timeBaseScheduleDayPlan object.  
<Format> The value of this object cannot exceed the value of the maxDayPlans object. A value of zero (0) shall indicate that this row has been disabled.  
<Supplanted by> ISO26048-1-DayPlan.fdayPlanScheduleDayPlan (ISO 26048-1)  
<Informative> The replacement object extends the range to Unsigned32  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.2.1.5"  
 ::= { timeBaseScheduleEntry 5 }

### 2.6.3 Maximum Number of Day Plans-Parameter

maxDayPlans OBJECT-TYPE  
SYNTAX Integer32(1..255)  
UNITS "DayPlan"  
MAX-ACCESS read-only

STATUS deprecated  
DESCRIPTION  
" <Definition>The value of this object specifies the maximum, fixed number of different timebased Day Plans supported by the device. The value of this object represents the number of day plans (primary key into the table) available in the timeBaseDayPlanTable.  
  
<Informative> The timeBaseDayPlanTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.3"  
 ::= { timebase 3 }

#### 2.6.4 Maximum Number of Day Plan Events-Parameter

maxDayPlanEvents OBJECT-TYPE  
SYNTAX Integer32(1..255)  
UNITS "DayPlanEvent"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition>The value of this object specifies the fixed number of different timebased Day Plan Events within each Day Plan supported by the device. The value of this object represents the number of rows (secondary key into the table) available within each of the day plans that are available in the timeBaseDayPlanTable. All day plans shall have the same number of day plan events available for use.  
  
<Informative> The timeBaseDayPlanTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.4"  
 ::= { timebase 4 }

#### 2.6.5 Day Plan Table

timeBaseDayPlanTable OBJECT-TYPE  
SYNTAX SEQUENCE OF TimeBaseDayPlanEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition>A table containing day plan numbers, the times when to implement them and the associated actions. The number of rows in this table shall be equal to the product of the maxDayPlans object and the maxDayPlanEvents object. The dayPlanNumbers within this table shall begin with day plan number 1 and increment by one to the maxDayPlans. The dayPlanEventNumbers within this table shall begin with day plan event number 1 and increment by one to the maxDayPlanEvents.  
  
This table is always used in association with device-type specific objects specifying device-type specific actions such as activating a message on a VMS sign or initiating a pattern for a signal controller. A device MIB that defines an action table should define the relative priority of the action table as compared to the priority of system and other commands. The device-type specific action is only initiated when (1) the specific DayPlan has been activated, (2) the scheduler has sufficient priority to override the current operation of the device, and (3) at the indicated time.

After a power recovery, or after a change to any object that affects `controlerLocalTime`, the operational mode called for by the scheduler shall be per the last event that would have been called for by the currently defined schedule; the logic searches for all events that may have occurred for at least the previous 24 hours.

<Table Type> static

<Supplanted by> ISO26048-1-DayPlan.fdayPlanTable & ISO26048-1-DayPlan.fdayPlanTriggerTable (ISO 26048-1)

<Informative> The replacement tables are designed to trigger an action, which can include activating a day plan (as originally envisioned by this object) and/or to call other actions (e.g., creating a log entry).

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.5"

::= { timebase 5 }

timeBaseDayPlanEntry OBJECT-TYPE

SYNTAX TimeBaseDayPlanEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition>Day plan parameters for the time based schedule programming of a device.

<Supplanted by> ISO26048-1-DayPlan.fdayPlanEntry & ISO26048-1-DayPlan.fdayPlanTriggerEntry

<Informative> The replacement objects extend the row to support a description of the day plan and supports full SNMPv3 row management.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.5.1"

INDEX { dayPlanNumber, dayPlanEventNumber }

::= { timeBaseDayPlanTable 1 }

```
TimeBaseDayPlanEntry ::= SEQUENCE {
    dayPlanNumber          INTEGER,
    dayPlanEventNumber    INTEGER,
    dayPlanHour           INTEGER,
    dayPlanMinute         INTEGER,
    dayPlanActionNumberOID VariablePointer}
```

### 2.6.5.1 Day Plan Number

dayPlanNumber OBJECT-TYPE

SYNTAX Integer32(1..255)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> This object specifies the day plan number for objects in this row. The value shall not exceed the value of the `maxDayPlans` object. Day plan numbers are used in the TimeBase Event Table to specify day plan numbers to be implemented on specific days of the year or as part of the week plans.

<Supplanted by> ISO26048-1-DayPlan.fdayPlanNumber (ISO 26048-1)

<Informative> The replacement object extends the range to an Unsigned32

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.5.1.1"

```
::= { timeBaseDayPlanEntry 1 }
```

### 2.6.5.2 Day Plan Event Number

```
dayPlanEventNumber OBJECT-TYPE  
SYNTAX      Integer32(1..255)  
MAX-ACCESS  read-only  
STATUS      deprecated  
DESCRIPTION
```

"<Definition> This object identifies day plan event number(s) to be scheduled on a specific day plan number. Several different events can be scheduled to take place during a day, and each of these events is one entry or row within a specified day plan number. The total number of events for one day plan shall not exceed the value of the maxDayPlanEvents object. If multiple non-conflicting events are scheduled to occur at the same time, they shall be logically executed in order of their dayPlanEventNumber with the lowest number occurring first. An implementation shall omit lower number actions that are in conflict with higher number actions at the same time.

<Supplanted by> ISO26048-1-DayPlan.fdDayPlanTriggerTime (ISO 26048-1)

<Informative> The replacement object (and second index into the replacement table) is the millisecond-based time object.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.5.1.2"

```
::= { timeBaseDayPlanEntry 2 }
```

### 2.6.5.3 Day Plan Hour Parameter

```
dayPlanHour OBJECT-TYPE  
SYNTAX      Integer32(0..23)  
MAX-ACCESS  read-write  
STATUS      deprecated  
DESCRIPTION
```

"<Definition> The Hour of day, as measured by the controllerLocalTime object, that the associated event shall become active.

<Supplanted by> ISO26048-1-DayPlan.fdDayPlanTriggerTime (ISO 26048-1)

<Informative> The replacement object includes the complete daily timestamp and is used as an index to the replacement table.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.5.1.3"

```
DEFVAL      { 0 }
```

```
::= { timeBaseDayPlanEntry 3 }
```

### 2.6.5.4 Day Plan Minute Parameter

```
dayPlanMinute OBJECT-TYPE  
SYNTAX      Integer32(0..59)  
MAX-ACCESS  read-write  
STATUS      deprecated  
DESCRIPTION
```

"<Definition> The Minute of the hour (defined in the dayPlanHour), as measured by the controllerLocalTime object, that the associated event shall become active.

<Supplanted by> ISO26048-1-DayPlan.fdDayPlanTriggerTime (ISO 26048-1)

<Informative> The replacement object includes the complete daily timestamp and is used as an index to the replacement table.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.5.1.4"

```
DEFVAL      { 0 }  
::= { timeBaseDayPlanEntry 4 }
```

### 2.6.5.5 Day Plan Action Number OID Parameter

```
dayPlanActionNumberOID OBJECT-TYPE  
SYNTAX      VariablePointer  
MAX-ACCESS  read-write  
STATUS      deprecated  
DESCRIPTION
```

"<Definition> This object provides a reference to the device-type specific action that shall be executed. The object shall reference the action by its associated object identifier, including its instance (i.e., the full OID of the scalar or columnar object). Only objects whose description field explicitly states that they may be called by the action table may be referenced. If a management system attempts to set this value to any other object identifier, the device shall respond with a genErr.

Any object allowing the action table to reference it shall define precisely what action takes place when it is activated, and whether the action is transitional or continuous until deactivated. The object shall also define what, if any, restrictions may be placed on other operations the device may be able to perform.

If the action to be performed is defined by a row of a table, one of the index columns should be identified as the explicit object that is referenced.

<Supplanted by> ISO26048-1-DayPlan.fdDayPlanTriggerActionGroup

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.5.1.5"

```
DEFVAL      { zeroDotZero }  
::= { timeBaseDayPlanEntry 5 }
```

### 2.6.6 Day Plan Status Parameter

```
dayPlanStatus OBJECT-TYPE  
SYNTAX      Integer32(0..255)  
MAX-ACCESS  read-only  
STATUS      deprecated  
DESCRIPTION
```

"<Definition>This object indicates the current value of the active dayPlanNumber-object.

<Format> A value of zero (0) indicates that there is no dayPlanNumber that is currently active.

<Supplanted by> ISO26048-1-DayPlan.fdDayPlanSchedulerCurrentDayPlan (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.6"

```
::= { timebase 6 }
```

### 2.6.7 Schedule Status Parameter

```
timeBaseScheduleTableStatus OBJECT-TYPE  
SYNTAX      Integer32(0..65535)  
MAX-ACCESS  read-only  
STATUS      deprecated  
DESCRIPTION
```

"<Definition>This object indicates the number of the TimeBaseSchedule which is currently selected by the scheduling logic; the device may or may not be

using the selected schedule. The value of zero (0) indicates that there is no timeBaseScheduleNumber that is currently selected.

<Supplanted by> ISO26048-1-DayPlan.fdayPlanSchedulerSelectedRule (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.7"

::= { timebase 7 }

### 2.6.8 Global Local Time Differential Parameter

globalLocalTimeDifferential OBJECT-TYPE

SYNTAX Integer32(-43200..43200)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"Indicates the number of seconds offset between local time and GMT. Positive values indicate local times in the Eastern Hemisphere up to the International Date Line and negative values indicate local times in the Western Hemisphere back to the International Date Line. If one of the daylight saving times is activated, this value will change automatically at the referenced time. For example, Central Standard Time (CST) is -21600 and Central Daylight Time (CDT) is -18000.

<Supplanted by> controllerStandardTimeZone

<Informative> This object was deprecated in NTCIP 1202 v02 to prevent confusion when setting time near a DST event.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.4"

::= { globalTimeManagement 4 }

### 2.6.9 Standard Time Zone Parameter

controllerStandardTimeZone OBJECT-TYPE

SYNTAX Integer32(-43200..43200)

UNITS "seconds"

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> Indicates the number of seconds offset between local Standard Time and GMT. Positive values indicate local times in the Eastern Hemisphere up to the International Date Line and negative values indicate local times in the Western Hemisphere back to the International Date Line. This value does not change in response to a DST event.

<Superseded by> ISO26048-1-Clock.fdclockLocalStandardTimeZone (ISO 26048-1)

<Informative> The replacement object extends the range to (-46800..46800) to support all defined time zones (i.e., +/- 13 hours).

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.5"

DEFVAL { 0 }

::= { globalTimeManagement 5 }

### 2.6.10 Local Time Parameter

controllerLocalTime OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION



"<Definition> The current local time expressed in seconds since 00:00:00 (midnight) January 1, 1970 of the same time offset. This value changes by 3600 seconds in response to a DST event.

<Superseded by> ISO26048-1-Clock.fdClockLocalDate & ISO26048-1-Clock.fdClockLocalTime (ISO 26048-1)

<Informative> The original specification defined this parameter using a Counter; however, by convention, Counter objects are supposed to follow the defined semantics for a counter in SNMPv1 and SNMPv3 requires this compliance. This object fails to meet this standard because it is not always increasing. Therefore, when presenting this object in SNMPv3, it is encoded as an Unsigned32 rather than a Counter32; proxy agents will need to address this encoding change within their implementation.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.6"

::= { globalTimeManagement 6 }

## 2.7 DAYLIGHT SAVING TIME (DST) NODE

daylightSavingNode OBJECT-IDENTITY

STATUS deprecated

DESCRIPTION

"<Definition> This node is an identifier used to organize all objects for support of defining DST. This function is common to most device types. The objects under this node only affect device operation when globalDaylightSaving = enableDaylightSavingNode (20). See Annex A.2.2 for examples.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7"

::= { globalTimeManagement 7 }

### 2.7.1 Maximum Daylight Saving Time (DST) Table Entries Parameter

maxDaylightSavingEntries OBJECT-TYPE

SYNTAX Integer32(1..100)

UNITS "entries"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The maximum number of entries (begin and end date pairs) that the DST Table can contain within the device. As of July 2007, devices used within the United States only require 1 entry when using the generic begin and end date method.

<Informative>It is expected that, for devices using the absolute date method, the device would need to support at least 1 entry per year programmed.

For multi-step DST transitions, a minimum of 2 rows are required (see Annex A.2.1 Figure 6).

More than one row may be required if absolute date method (see Section 2.4.8.2.2) is used for more than one year, or if more than one time change is implemented in a given year.

<Superseded by> ISO26048-1-Clock.fdClockDstMaxEntries (ISO 26048-1)

<Informative> The replacement object has a range of Unsigned32

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.1"

```
::= { daylightSavingNode 1 }
```

### 2.7.2 Daylight Saving Time (DST) Table Parameter

dstTable OBJECT-TYPE

SYNTAX SEQUENCE OF DstEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> A table containing DST Begin and End dates. The table is useful for agencies with multiple daylight saving time incremental steps per year. The number of rows in this table is equal to the maxDaylightSavingEntries object.

<Table Type> static

<Superseded by> ISO26048-1-Clock.fdClockDstTable (ISO 26048-1)

<Informative> The original table contained two SecondsToTransition objects that have an invalid syntax for SNMPv3.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2"

```
::= { daylightSavingNode 2 }
```

dstEntry OBJECT-TYPE

SYNTAX DstEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> The DST Begin and End dates parameters.

<Superseded by> ISO26048-1-Clock.fdClockDstEntry (ISO 26048-1)

<Informative> The replacement object extends the row to support a status object for each row (showing whether the plan is activated) as well as a RowStatus object to disable defined rows.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1"

```
INDEX { dstEntryNumber }
```

```
::= { dstTable 1 }
```

```
DstEntry ::= SEQUENCE {
```

dstEntryNumber	INTEGER,
dstBeginMonth	INTEGER,
dstBeginOccurrences	INTEGER,
dstBeginDayOfWeek	INTEGER,
dstBeginDayOfMonth	INTEGER,
dstBeginSecondsToTransition	INTEGER,
dstEndMonth	INTEGER,
dstEndOccurrences	INTEGER,
dstEndDayOfWeek	INTEGER,
dstEndDayOfMonth	INTEGER,
dstEndSecondsToTransition	INTEGER,
dstSecondsToAdjust	INTEGER}

#### 2.7.2.1 Daylight Saving Time (DST) Entry Number Parameter

dstEntryNumber OBJECT-TYPE

SYNTAX Integer32(1..100)

MAX-ACCESS read-only

STATUS deprecated  
DESCRIPTION  
 "<Definition> The entry number for the DST objects in this row. This value shall not exceed the maxDaylightSavingEntries object value."  
 <Superseded by> ISO26048-1-Clock.fdClockDstIndex (ISO 26048-1)  
 <Informative> The replacement object has a range of ITSPositive8  
 <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.1"  
 ::= { dstEntry 1 }

### 2.7.2.2 Daylight Saving Time (DST) Beginning Month Parameter

dstBeginMonth OBJECT-TYPE  
SYNTAX INTEGER { january (1),  
 february (2),  
 march (3),  
 april (4),  
 may (5),  
 june (6),  
 july (7),  
 august (8),  
 september (9),  
 october (10),  
 november (11),  
 december (12),  
 absolute (13),  
 disabled (14) }

MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
 "<Definition> The month during which daylight saving time (DST) begins."  
 <Format> An entry of 'absolute' means that dstBeginSecondsToTransition defines an absolute time to begin DST relative to midnight January 1, 1970. In this case, any value indicated in the dstEndMonth, dstBeginOccurrences, dstBeginDayOfWeek, dstBeginDayOfMonth, dstEndOccurrences, dstEndDayOfWeek, and dstEndDayOfMonth objects are irrelevant, and the dstEndSecondsToTransition object defines an absolute time to end DST relative to midnight January 1, 1970.  
 If the daylightSavingNode is enabled (i.e. globalDaylightSaving = enableDaylightSavingNode), and the value of this object is disabled(14), then the values in the remaining objects in this row of the dstTable are irrelevant and therefore ignored by the device.  
 <Superseded by> ISO26048-1-Clock.fdClockDstBeginMonth (ISO 26048-1)  
 <Informative> The replacement object does not support absolute mode and disabling a row is achieved through the RowStatus object.  
 <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.2"  
 DEFVAL { march }  
 ::= { dstEntry 2 }

### 2.7.2.3 Daylight Saving Time (DST) Beginning Occurrence Parameter

dstBeginOccurrences OBJECT-TYPE  
SYNTAX INTEGER { first (1),  
 second (2),  
 third (3),

```
fourth (4),  
last (5),  
secondLast (6),  
thirdLast (7),  
fourthLast (8),  
specificDayOfMonth (9) }
```

MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition>For values of 1-4, the number of occurrences of the specific day of week that shall occur on or after dstBeginDayOfMonth until the daylight saving transition shall take place.

For values of 5-8, the number of occurrences of the specific day of week that shall occur on or before dstBeginDayOfMonth until the daylight saving transition shall take place.

For value = 9, dstBeginDayOfMonth defines the specific day of the month that the DST transition occurs regardless of value in dstBeginDayOfWeek object.

<Superseded by> ISO26048-1-Clock.fdClockDstBeginOccurrences (ISO 26048-1)

<Informative> To specify the last occurrence of a specified day of the month, simply specify the last occurrence of the specified day of the week on or before the last day of the month (e.g., 31).  
The replacement object has an identical range.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.3"

DEFVAL { second }  
::= { dstEntry 3 }

#### 2.7.2.4 Daylight Saving Time (DST) Beginning Day of Week Parameter

dstBeginDayOfWeek OBJECT-TYPE  
SYNTAX INTEGER { sunday (1),  
monday (2),  
tuesday (3),  
wednesday (4),  
thursday (5),  
friday (6),  
saturday (7) }

MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The Day of the week on which daylight saving time (DST) begins. This object shall only apply if dstBeginOccurrences = 1-8.

<Superseded by> ISO26048-1-Clock.fdClockDstBeginDayOfWeek (ISO 26048-1)

<Informative> The replacement object shifts Sunday to the end of the enumeration (7) and all other days up one to conform to international conventions in other standards.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.4"

DEFVAL { sunday }  
::= { dstEntry 4 }

#### 2.7.2.5 Daylight Saving Time (DST) Beginning Day of Month Parameter

dstBeginDayOfMonth OBJECT-TYPE  
SYNTAX Integer32 (1..31)

UNITS "day of month"  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> If dstBeginOccurrences = 1-8: The day of the month from which to begin counting occurrences of a specific day of the week (forward for values 1-4, and backwards for values 5-8).  
If dstBeginOccurrences = 9: The specific day of the month on which the transition occurs.  
<Superseded by> ISO26048-1-Clock.fdClockDstBeginDayOfMonth (ISO 26048-1)  
<Informative> The replacement object has an identical range.  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.5"  
DEFVAL { 1 }  
::= { dstEntry 5 }

#### 2.7.2.6 Daylight Saving Time (DST) Beginning Seconds to Transition Parameter

dstBeginSecondsToTransition OBJECT-TYPE  
SYNTAX Integer32  
UNITS "seconds"  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> If dstBeginMonth = absolute, then this object defines when DST begins based on the seconds from midnight January 1, 1970 (UTC/GMT).  
If dstBeginMonth = 1-12 (January to December), then this object defines the time when DST begins in seconds past midnight relative to local time (see the controllerLocalTime object).  
<Superseded by> ISO26048-1-Clock.fdClockDstBeginTime (ISO 26048-1)  
<Informative> A set of parameters that causes a day transition that crosses the midnight boundary may result in unexpected behavior.  
The original specification defined this parameter using INTEGER (0..4294967295); however, by convention, INTEGERS are limited to the range (-2147483648..2147483647) in SNMPv1 and SNMPv3 enforces this limitation. Therefore, when presenting this object in SNMPv3, it is encoded as an Unsigned32 rather than an INTEGER (0..4294967295); proxy agents will need to address this encoding change within their implementation.  
The replacement object is a daily timestamp to the millisecond and does not support an absolute time.  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.6"  
DEFVAL { 7200 }  
::= { dstEntry 6 }

#### 2.7.2.7 Daylight Saving Time (DST) Ending Month Parameter

dstEndMonth OBJECT-TYPE  
SYNTAX INTEGER { january (1),  
february (2),  
march (3),  
april (4),  
may (5),  
june (6),  
july (7),  
august (8),

```
    september (9),  
    october (10),  
    november (11),  
    december (12) }
```

MAX-ACCESS read-write  
STATUS deprecated

DESCRIPTION

"<Definition> The month during which daylight saving time (DST) ends. If the value of dstBeginMonth object = 'absolute' or 'disabled', then the agent shall ignore the value of this object. Otherwise, the value of this object is valid.

<Superseded by> ISO26048-1-Clock.fdClockDstEndMonth (ISO 26048-1)

<Informative> The replacement object has an identical syntax.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.7"

DEFVAL { november }  
::= { dstEntry 7 }

### 2.7.2.8 Daylight Saving Time (DST) Ending Occurrences Parameter

dstEndOccurrences OBJECT-TYPE

```
SYNTAX      INTEGER { first (1),  
                    second (2),  
                    third (3),  
                    fourth (4),  
                    last (5),  
                    secondLast (6),  
                    thirdLast (7),  
                    fourthLast (8),  
                    specificDayOfMonth (9) }
```

MAX-ACCESS read-write  
STATUS deprecated

DESCRIPTION

"<Definition>For values of 1-4, the number of occurrences of the specific day of week that shall occur on or after dstEndDayOfMonth until the daylight saving transition shall take place.

For values of 5-8, the number of occurrences of the specific day of week that shall occur on or before dstEndDayOfMonth until the daylight saving transition shall take place.

For value = 9, dstEndDayOfMonth defines the specific day of the month that the DST transition occurs regardless of value in dstEndDayOfWeek object.

<Superseded by> ISO26048-1-Clock.fdClockDstEndOccurrences (ISO 26048-1)

<Informative> To specify the last occurrence of a specified day of the month, simply specify the last occurrence of the specified day of the week on or before the last day of the month (e.g. 31).  
The replacement object has an identical range.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.8"

DEFVAL { first }  
::= { dstEntry 8 }

### 2.7.2.9 Daylight Saving Time (DST) Ending Day of Week Parameter

dstEndDayOfWeek OBJECT-TYPE

```
SYNTAX      INTEGER { sunday (1),  
                    monday (2),
```

```
tuesday (3),  
wednesday (4),  
thursday (5),  
friday (6),  
saturday (7) }
```

```
MAX-ACCESS read-write  
STATUS deprecated
```

DESCRIPTION

"<Definition> The Day of the week on which daylight saving time (DST) ends. This object shall only apply if dstEndOccurrences = 1-8.

<Superseded by> ISO26048-1-Clock.fclockDstEndDayOfWeek (ISO 26048-1)

<Informative> The replacement object shifts Sunday to the end of the enumeration (7) and all other days up one to conform to international conventions in other standards.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.9"

```
DEFVAL { sunday }  
::= { dstEntry 9 }
```

#### 2.7.2.10 Daylight Saving Time (DST) Ending Day of Month Parameter

dstEndDayOfMonth OBJECT-TYPE

```
SYNTAX Integer32(1..31)
```

```
UNITS "day of month"
```

```
MAX-ACCESS read-write
```

```
STATUS deprecated
```

DESCRIPTION

"<Definition> If dstEndOccurrences = 1-8: The day of the month from which to begin counting occurrences of a specific day of the week (forward for values 1-4, and backwards for values 5-8).  
If dstEndOccurrences = 9: The specific day of the month on which the transition occurs.

<Superseded by> ISO26048-1-Clock.fclockDstEndDayOfMonth (ISO 26048-1)

<Informative> The replacement object has an identical range.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.10"

```
DEFVAL { 1 }  
::= { dstEntry 10 }
```

#### 2.7.2.11 Daylight Saving Time (DST) Ending Seconds to Transition Parameter

dstEndSecondsToTransition OBJECT-TYPE

```
SYNTAX Integer32
```

```
UNITS "seconds"
```

```
MAX-ACCESS read-write
```

```
STATUS deprecated
```

DESCRIPTION

"<Definition> If dstBeginMonth = absolute, then this object defines when DST ends based on the seconds from midnight January 1, 1970 (UTC/GMT).

If dstBeginMonth = 1-12 (January to December), then this object defines the time when DST ends in seconds past midnight relative to local time (see the controllerLocalTime object).

<Superseded by> ISO26048-1-Clock.fclockDstEndTime (ISO 26048-1)

<Informative> A set of parameters that causes a day transition that crosses the midnight boundary may result in unexpected behavior.

The original specification defined this parameter using INTEGER (0..4294967295); however, by convention, INTEGERS are limited to the range (-2147483648..2147483647) in SNMPv1 and SNMPv3 enforces this limitation. Therefore, when presenting this object in SNMPv3, it is encoded as an Unsigned32 rather than an INTEGER (0..4294967295); proxy agents will need to address this encoding change within their implementation. The replacement object is a daily timestamp to the millisecond and does not support an absolute time.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.11"

DEFVAL { 7200 }  
::= { dstEntry 11 }

### 2.7.2.12 Daylight Saving Time (DST) Seconds to Adjust Parameter

dstSecondsToAdjust OBJECT-TYPE  
SYNTAX Integer32 (0..21600)  
UNITS "seconds"  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> This is the absolute offset in seconds that will be added to the local time reference point to determine the local time when DST is in effect as specified by this row entry. Values of this object in adjacent rows, even if they overlap, are not cumulative. That is, the row with the latest dstBegin time, which has not terminated due to passing the dstEnd time, shall determine the setting of the local TOD clock; the dstSecondsToAdjust for the latest dstBegin governs the Local TOD clock settings.

The maximum offset to adjust is 21600 seconds, an equivalent of 6 hours.

<Superseded by> ISO26048-1-Clock.fdclockDstOffset (ISO 26048-1)

<Informative> This object allows what may be considered an exception, in that it is possible and allowed to configure an adjustment backward past midnight. The replacement object uses an ITSInteger16 range.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.7.2.1.12"

DEFVAL { 3600 }  
::= { dstEntry 12 }

## 2.8 PMPP OBJECT NODE

profilesPMPP OBJECT-IDENTITY  
STATUS deprecated  
DESCRIPTION

"<Definition> This node is an identifier used to group all objects for support of the PMPP function that are common to all device types. The objects under this node are placed under the Protocols\Profiles\PMPP subtree within the NEMA node, but they have been listed here due to the lack of a separate document that lists these objects.

<Informative> PMPP is a historic protocol designed for multi-drop serial communication networks with typical data capacities of 9600 bits per second or less. The overhead of X.509 security certificates and the availability of alternate communication technologies has resulted in the decision to no longer maintain this protocol and the deprecation of all of its management objects.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.2.3"



```
::= { profiles 3 }
```

### 2.8.1 Maximum HDLC Group Address Parameter

```
maxGroupAddresses OBJECT-TYPE
SYNTAX      Integer32(1..255)
UNITS       "addresses"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition>The maximum number of group addresses this device supports.
  This object indicates the maximum number of rows in the
  hdlcGroupAddressTable.

  <Informative> The PMPP protocol has been deprecated and there is no
  replacement object.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.3.1"
```

```
::= { profilesPMPP 1 }
```

### 2.8.2 HDLC Group Address Table

```
hdlcGroupAddressTable OBJECT-TYPE
SYNTAX      SEQUENCE OF HdlcGroupAddressEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition> A table containing group addresses at which a device may
  receive frames.

  <Table Type> static

  <Informative> The PMPP protocol has been deprecated and there is no
  replacement object.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.3.2"
```

```
::= { profilesPMPP 2 }
```

```
hdlcGroupAddressEntry OBJECT-TYPE
SYNTAX      HdlcGroupAddressEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition> An entry in the group address table that contains a device's
  data link layer group address at which it will accept frames.

  <Informative> The PMPP protocol has been deprecated and there is no
  replacement object.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.3.2.1"
```

```
INDEX      { hdlcGroupAddressIndex }
::= { hdlcGroupAddressTable 1 }
```

```
HdlcGroupAddressEntry ::= SEQUENCE {
    hdlcGroupAddressIndex  INTEGER,
    hdlcGroupAddress       INTEGER, -- deprecated previously
    hdlcGroupAddressNumber INTEGER}
```

### 2.8.2.1 HDLC Group Address Index Parameter

```
hdlcGroupAddressIndex OBJECT-TYPE
SYNTAX      Integer32(1..255)
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The index number for the group address in this row.
  <Informative> The PMPP protocol has been deprecated and there is no
  replacement object.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.3.2.1.1"
 ::= { hdlcGroupAddressEntry 1 }
```

### 2.8.2.2 HDLC Group Address Parameter

```
hdlcGroupAddress OBJECT-TYPE
SYNTAX      Integer32
MAX-ACCESS  read-write
STATUS      deprecated
DESCRIPTION
  "<Definition> A group address for the data link layer. For PMPP, the syntax
  is an 8 or 16 bit entry with the second low order bit set to a one
  indicating that this is a group address.
  <Informative> This object was deprecated in NTCIP 1201 v03. The PMPP
  protocol has been deprecated and there is no replacement object.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.3.2.1.2"
REFERENCE   "NEMA TS 3.3 Clause 3.3.3.1"
 ::= { hdlcGroupAddressEntry 2 }
```

### 2.8.2.3 HDLC Group Address Number Parameter

```
hdlcGroupAddressNumber OBJECT-TYPE
SYNTAX      Integer32(0..62)
MAX-ACCESS  read-write
STATUS      deprecated
DESCRIPTION
  "<Definition> A group address number prior to any encoding for the data
  link layer. The address of 63 is reserved for the all stations address. The
  value of zero (0) shall disable this row of the table.
  <Informative> In PMPP all group addresses are encoded in one byte.
  The PMPP protocol has been deprecated and there is no replacement object.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.3.2.1.3"
REFERENCE   "NTCIP 2101"
DEFVAL     { 0 }
 ::= { hdlcGroupAddressEntry 3 }
END
```

### Section 3 DEPRECATED AUXILIARY I/O V2 MIB

The text provided from Section 3.1 through the end of Section 3 (except the headings) constitutes the standard NTCIP1201-AuxIOv2 MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

#### 3.1 AUXILIARY I/O V2 HEADER

```
NTCIP1201-AuxIOv2 DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Integer32
    FROM SNMPv2-SMI
    -- RFC 2578
    DisplayString
    FROM SNMPv2-TC
    -- RFC 2579
    global
    FROM NTCIP1201-Global;

auxIOv2 MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of the AuxIOv2
    objects that were previously defined in NTCIP 1201 v03.

    Auxiliary I/O was originally defined in NTCIP 1203 v01 under the
    experimental node. NTCIP 1201 v02 moved the objects to the location defined
    by this MIB but retained the same object names. Experience demonstrated
    challenges in compiling MIB files with duplicate object names and as a
    result NTCIP 1201 v03 changed the names of objects while retaining the
    NTCIP 1201 v02 object identifiers. NTCIP 1201 v04 deprecated these objects
    in favor of the general purpose I/O design defined in ISO 26048-1.

    For those agents that may support these objects and those originally
    defined under the experimental node (see Section 2.10), the object
    definitions are treated as aliases such that a write to an object in one
    group acts as write to the corresponding object in the other group. As
    aliases, a read of an object in this group is equivalent to a read of the
    corresponding object in the auxIO group.

    *** All objects in this MIB have been deprecated. ***

    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7"
REVISION "20230701145436Z"
DESCRIPTION
    "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "200610020000Z"
DESCRIPTION
```

"NTCIP 1201 v03 - Created this as a standalone MIB. Changed all statuses to 'mandatory' to eliminate checking errors. Revised object names."

```
::= { global 7 }
```

## 3.2 OBJECTS

### 3.2.1 Maximum Number of Digital Auxiliary I/Os Parameter

```
maxAuxIOv2TableNumDigitalPorts OBJECT-TYPE
SYNTAX      Integer32 (0..255)
UNITS       "digital ports"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The number of rows contained in the 'auxIOv2Table' with the
  auxIOv2PortType set to 'digital'."
  <Supplanted by> ISO26048-1-Srsa.fdSrsaTypeCount (ISO 26048-1)
  <Informative> The GPIO count objects are managed by port type.
  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.1"
```

```
::= { auxIOv2 1 }
```

### 3.2.2 Maximum Number of Analog Auxiliary I/Os Parameter

```
maxAuxIOv2TableNumAnalogPorts OBJECT-TYPE
SYNTAX      Integer32 (0..255)
UNITS       "analog ports"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition>The number of rows contained in the 'auxIOv2Table' with the
  auxIOv2PortType set to 'analog'."
  <Supplanted by> ISO26048-1-Srsa.fdSrsaTypeCount (ISO 26048-1)
  <Informative> The GPIO count objects are managed by port type.
  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.2"
```

```
::= { auxIOv2 2 }
```

### 3.2.3 Auxiliary I/O Table Parameter

```
auxIOv2Table OBJECT-TYPE
SYNTAX      SEQUENCE OF AuxIOv2Entry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition>A table providing the means to access any non- mission-
  critical or non-safety-related auxiliary I/O of the controller, including
  reading inputs and setting outputs. The number of rows in this table equals
  the sum of the values of the 'maxAuxIOv2TableNumDigitalPorts' and
  'maxAuxIOv2TableNumAnalogPorts' objects.

  This table shall not be used to control or monitor any safety related
  equipment. The electrical levels used by the ports are not standardized by
  auxIOv2Table objects; such information should be contained in the hardware
  manual.

  <Table Type> static"
```

```
<Supplanted by> ISO26048-1-Srsa.fdSrsaTable & FIELD-DEVICE-GPIO-
MIB.fdGPIOPortTable (ISO 26048-1)

<Informative> The GPIO objects provide a summary table for each port type.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3"

 ::= { auxIOv2 3 }

auxIOv2Entry OBJECT-TYPE
SYNTAX      AuxIOv2Entry
MAX-ACCESS not-accessible
STATUS      deprecated
DESCRIPTION
    "<Definition>Parameters of the auxiliary I/O table.

    <Supplanted by> ISO26048-1-Srsa.fdSrsaEntry & FIELD-DEVICE-GPIO-
    MIB.fdGPIOPortEntry (ISO 26048-1)

    <Informative> The GPIO tables add columns for a count for each port type, a
    summary status of each type of port, an indication of the units reported by
    the port, an indication of the minimum and maximum values that can be
    reliably reported by the port, minimum and maximum threshold values that
    indicate when an alarm should be raised, and a status object that indicates
    any availability or alarm conditions.

    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1"

INDEX      { auxIOv2PortType, auxIOv2PortNumber }
 ::= { auxIOv2Table 1 }

AuxIOv2Entry ::= SEQUENCE {
    auxIOv2PortType          INTEGER,
    auxIOv2PortNumber        INTEGER,
    auxIOv2PortDescription   DisplayString,
    auxIOv2PortResolution    INTEGER,
    auxIOv2PortValue         INTEGER,
    auxIOv2PortDirection     INTEGER,
    auxIOv2PortLastCommandedState INTEGER}


```

### 3.2.3.1 Auxiliary Port Type Parameter

```
auxIOv2PortType OBJECT-TYPE
SYNTAX      INTEGER{ other (1),
                    analog (2),
                    digital (3)
                }
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
    "<Definition>Indicates the type of auxiliary I/O, which can be analog or
    digital.

    <Supplanted by> ISO26048-1-Srsa.fdSrsaType (ISO 26048-1)

    <Informative> The GPIO tables classify ports using a three-letter code. ISO
    26048-1 defines 23 port types and additional port types can be defined by
    registering with ISO. Example port types currently defined include
    humidity, light intensity, temperature, battery current, battery voltage,
    battery charge, generator fuel level, generator engine speed, fan status,
    etc.


```

From NTCIP 1201 v01 to NTCIP v02 of these objects, it was determined that ports are either digital, analog, or other.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.1"

::= { auxIOv2Entry 1 }

### 3.2.3.2 Auxiliary Port Number Parameter

auxIOv2PortNumber OBJECT-TYPE

SYNTAX Integer32 (1..255)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>Indicates the port number for the associated port type. Port numbers are used sequentially from one to max for each port type. There can be a port 1 for analog port and port 1 for digital port.

<Supplanted by> ISO26048-1-Srsa.fdSrsaPortNumber (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.2"

::= { auxIOv2Entry 2 }

### 3.2.3.3 Auxiliary Description Parameter

auxIOv2PortDescription OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..255))

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>Informational text field describing the device at the associated auxiliary I/O.

<Supplanted by> ISO26048-1-Srsa.fdSrsaPortDescription (ISO 26048-1)

<Informative> The GPIO object is an SnmpAdminString, which supports multi-lingual text.

In NTCIP 1203 v01, the SYNTAX SIZE was listed as (0..50). In NTCIP 1201 v02 and NTCIP 1201 v03, this was changed to (0..255).

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.3"

::= { auxIOv2Entry 3 }

### 3.2.3.4 Auxiliary Resolution Parameter

auxIOv2PortResolution OBJECT-TYPE

SYNTAX Integer32 (1..255)

UNITS "bits"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>Defines number of bits used for the IO-port (e.g. width of digital, resolution of analog). Thus, this feature allows the digital monitoring (via NTCIP) of an analog port on the agent.

<Informative> In NTCIP 1203 v01, ACCESS was listed as read-write; however, in NTCIP 1201 v03, ACCESS changed to read-only . This changed because resolution is fixed by the hardware implementation and cannot be changed by the management station.

The SYNTAX also changed from NTCIP 1201 v02 to NTCIP 1201 v03; it is now as it was originally under the experimental node defined in NTCIP 1203v01.

This changed to address backward compatibility and the 'aliasing' between the version 1 objects (see Section 2.10) and the Version 02 objects.

<Supplanted by> ISO26048-1-Srsa.fdSrsaPortMinValue & ISO26048-1-Srsa.fdSrsaPortMaxValue (ISO 26048-1)

<Informative> The GPIO table indicates the range over which values can be considered reliable.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.4"

::= { auxIOv2Entry 4 }

### 3.2.3.5 Auxiliary Value Parameter

auxIOv2PortValue OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>For input or bidirectional ports, this contains the current value of the input. For output ports, this is the last commanded value of the port. A genError shall be generated, if this object is set and the port is an input. The actual value exchanged shall not exceed [2^(auxIOv2PortResolution): 1]; any SET operation to a value in excess of this number shall result in a genErr, and any GET response in excess of this value shall be considered erroneous.

<Supplanted by> ISO26048-1-Srsa.fdSrsaPortValue (ISO 26048-1)

<Informative> The original specification defined this parameter using INTEGER (0..4294967295); however, by convention, INTEGERS are limited to the range (-2147483648..2147483647) in SNMPv1 and SNMPv3 enforces this limitation. Therefore, when presenting this object in SNMPv3, it is encoded as an Unsigned32 rather than an INTEGER (0..4294967295); proxy agents will need to address this encoding change within their implementation. The replacement object in the GPIO table supports a signed 32-bit integer, thereby allowing sensors to report negative values.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.5"

::= { auxIOv2Entry 5 }

### 3.2.3.6 Auxiliary Port Direction Parameter

auxIOv2PortDirection OBJECT-TYPE

SYNTAX INTEGER { output (1),  
input (2),  
bidirectional (3) }

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>Indicates whether state of this port can be set (output), read (input) or both (bidirectional).

<Supplanted by> ISO26048-1-Srsa.fdSrsaPortDirection (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.6"

::= { auxIOv2Entry 6 }

### 3.2.3.7 Auxiliary Port Last Commanded State Parameter

auxIOv2PortLastCommandedState OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>For bi-directional ports, this object indicates the last state to which the auxIOv2PortValue object was set. For output ports, this value shall always be equal to the auxIOv2PortValue object. For input ports, this value shall always be zero (0).

<Supplanted by> ISO26048-1-Srsa.fdSrsaRequestedValue (ISO 26048-1)

<Informative> The original specification defined this parameter Using INTEGER (0..4294967295); however, by convention, INTEGERS Are limited to the range (-2147483648..2147483647) in SNMPv1 and SNMPv3 enforces this limitation. Therefore, when presenting this Object in SNMPv3, it is encoded as an Unsigned32 rather than an INTEGER (0..4294967295); proxy agents will need to address this encoding change within their implementation.

The replacement object in the GPIO table supports a signed 32- bit integer, thereby allowing negative output values.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.7"

::= { auxIOv2Entry 7 }  
END



## Section 4 DEPRECATED AUXILIARY I/O MIB

The text provided from Section 4.1 through the end of Section 4 (except the headings) constitutes the standard NTCIP1201-AuxiliaryIO MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 4.1 AUXILIARY I/O HEADER

```
NTCIP1201-AuxiliaryIO DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Integer32
    FROM SNMPv2-SMI
    DisplayString
    FROM SNMPv2-TC
    expGlobal
    FROM NTCIP8004-NEMA;

auxiliaryIO MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of the AuxIO
    objects that were previously defined in NTCIP 1201 v03.

    This MIB contains the auxiliary input/output (I/O) objects originally
    defined in NTCIP 1203 v01 (a.k.a. NTCIP 1203:1997) under the experimental
    node. When the objects were moved to NTCIP 1201 v02, a slightly revised
    structure was developed under the global node and the original experimental
    objects were deprecated. This MIB is provided for backwards compatibility
    to support access to the experimental objects via an SNMPv3 proxy agent.

    In the context of implementation that supports these objects, there is no
    difference between what appeared in NTCIP 1203:1997 and what appears here.

    The auxiliary I/O management objects listed herein define a mechanism for
    the support of unspecified I/O for an NTCIP device. The agency or device
    specifications should define the intended operation of these ports.
    NOTE: These objects are still logically located under the nemaExperimental
    node and use their originally defined textual names and OIDs. For the
    purposes of backward compatibility, the object STATUS has been changed to
    deprecated. For those agents that may support these objects and the new
    objects under the global node (see Section 2.9), the object definitions
    shall be treated as aliases in that a write to an object in one group acts
    as write to the corresponding object in the other group. As aliases, a read
    of an object in one group also acts as read of the corresponding object in
    the other group.
```

Early NTCIP deployments included the Aux I/O objects defined in NTCIP 1203 v01 located under an experimental node. These objects were moved to a permanent node with the release of NTCIP 1201 v02 and given new names. This can create confusion and backward compatibility issues. As noted in the object definition, both sets of objects refer to the same functions within the device; hence, both sets of objects cause the same device action or provide the same device status. Agency specifications which do NOT require support for the Aux I/O objects under the experimental node should exclude the support for these experimental objects (which have been deprecated) to ensure backward compatibility. Support of the Aux I/O objects under the permanent node identified in NTCIP 1201 v03 may be optional or mandatory depending on the agency- or project specification.

Use the PRL to exclude support of NTCIP 1201 v01-defined aux I/O objects. The relationship between mandatory and optional support of NTCIP 1201 v01 (experimental) and NTCIP 1201 v02 objects is unique to the Aux I/O objects.

\*\*\* All objects in this MIB have been deprecated. \*\*\*

<Object Identifier> 1.3.6.1.4.1.1206.2.2.1"

REVISION "20230701145436Z"

DESCRIPTION

"NTCIP 1201 v04 - Upgraded format to SMIV2 (all objects were previously deprecated)."

REVISION "200610020000Z"

DESCRIPTION

"NTCIP 1201 v03 - Created this as a standalone MIB. Deprecated all objects."

::= { expGlobal 1 }

## 4.2 OBJECTS

### 4.2.1 Maximum Number of Digital Auxiliary I/Os Parameter

maxAuxIODigital OBJECT-TYPE

SYNTAX Integer32 (0..255)

UNITS "digital ports"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The number of rows contained in the 'auxIOtable' with the auxIOPortType set to 'digital'.

<Superseded by> maxAuxIOv2TableNumDigitalPorts

<Object Identifier> 1.3.6.1.4.1.1206.2.2.1.1"

::= { auxiliaryIO 1 }

### 4.2.2 Maximum Number of Analog Auxiliary I/Os Parameter

maxAuxIOAnalog OBJECT-TYPE

SYNTAX Integer32 (0..255)

UNITS "analog ports"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The number of rows contained in the 'auxIOtable' with the auxIOPortType set to 'analog'.

<Superseded by> maxAuxIOv2TableNumAnalogPorts

<Object Identifier> 1.3.6.1.4.1.1206.2.2.1.2"  
 ::= { auxiliaryIO 2 }

#### 4.2.3 Auxiliary I/O Table Parameter

auxIOTable OBJECT-TYPE  
SYNTAX SEQUENCE OF AuxIOEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition> A table providing the means to access the auxiliary I/O of the Controller, including reading inputs and setting outputs. A maximum of 255 auxiliary I/Os may be defined for all, digital, analog or other types of ports.

<Table Type> static

<Superseded by> auxIOv2Table

<Object Identifier> 1.3.6.1.4.1.1206.2.2.1.3"

::= { auxiliaryIO 3 }

auxIOEntry OBJECT-TYPE  
SYNTAX AuxIOEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition> Parameters of the auxiliary I/O table.

<Superseded by> auxIOv2Entry

<Object Identifier> 1.3.6.1.4.1.1206.2.2.1.3.1"

INDEX { auxIOPortType, auxIOPortNumber }  
 ::= { auxIOTable 1 }

AuxIOEntry ::= SEQUENCE {  
 auxIOPortType INTEGER,  
 auxIOPortNumber INTEGER,  
 auxIODescription DisplayString,  
 auxIOResolution INTEGER,  
 auxIOValue INTEGER,  
 auxIOPortDirection INTEGER}

#### 4.2.3.1 Auxiliary Port Type Parameter

auxIOPortType OBJECT-TYPE  
SYNTAX INTEGER{ other (1),  
 analog (2),  
 digital (3)  
 }

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION  
" <Definition> Indicates the type of auxiliary I/O, which may be analog, digital or other.

<Superseded by> auxIOv2PortType

<Object Identifier> 1.3.6.1.4.1.1206.2.2.1.3.1.1"

```
::= { auxIOEntry 1 }
```

#### 4.2.3.2 Auxiliary Port Number Parameter

```
auxIOPortNumber OBJECT-TYPE
SYNTAX      Integer32 (1..255)
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> Indicates the port number for the associated port type. Port
  numbers are used sequentially from one to max for each port type. There can
  be a port 1 for analog port and port 1 for digital port.

  <Superseded by> auxIOv2PortNumber

  <Object Identifier> 1.3.6.1.4.1.1206.2.2.1.3.1.2"
```

```
::= { auxIOEntry 2 }
```

#### 4.2.3.3 Auxiliary Description Parameter

```
auxIODescription OBJECT-TYPE
SYNTAX      DisplayString (SIZE (0..50))
MAX-ACCESS  read-write
STATUS      deprecated
DESCRIPTION
  "<Definition> Informational text field describing the device at the
  associated auxiliary I/O

  <Informative> In NTCIP 1203 v01, the SYNTAX SIZE was listed as (0..50). In
  all versions of NTCIP 1201 v02, auxIO2Description (this object's alias) was
  changed to (0..255). This does not present a backward compatibility issue
  if a NTCIP 1201 v02 management station limits the size of the DisplayString
  to 50 characters.

  <Superseded by> auxIOv2PortDescription

  <Object Identifier> 1.3.6.1.4.1.1206.2.2.1.3.1.3"
```

```
::= { auxIOEntry 3 }
```

#### 4.2.3.4 Auxiliary Resolution Parameter

```
auxIOResolution OBJECT-TYPE
SYNTAX      Integer32 (1..255)
UNITS       "bits"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> Defines number of bits used for the IO-port (e.g. width of
  digital, resolution of analog).

  <Informative> In NTCIP 1203 v01, the ACCESS was listed as read- write.
  Resolution is fixed by the hardware implementation and cannot be changed by
  the management station.

  <Superseded by> auxIOv2PortResolution

  <Object Identifier> 1.3.6.1.4.1.1206.2.2.1.3.1.4"
```

```
::= { auxIOEntry 4 }
```

#### 4.2.3.5 Auxiliary Value Parameter

```
auxIOValue OBJECT-TYPE
SYNTAX      Integer32
MAX-ACCESS  read-write
```

```
STATUS      deprecated
DESCRIPTION
  "<Definition> For input or bidirectional ports, this contains the current
  value of the input. For output ports, this is the last commanded value of
  the port. A genError shall be generated, if this object is set and the port
  is an input.

  <Superseded by> auxIOv2PortValue

  <Informative> The original specification defined this parameter using
  INTEGER (0..4294967295); however, by convention, INTEGERS are limited to
  the range (-2147483648..2147483647) in SNMPv1 and SNMPv3 enforces this
  limitation. Therefore, when presenting this object in SNMPv3, it is encoded
  as an Unsigned32 rather than an INTEGER (0..4294967295); proxy agents will
  need to address this encoding change within their implementation.

  <Object Identifier> 1.3.6.1.4.1.1206.2.2.1.3.1.5"
 ::= { auxIOEntry 5 }

4.2.3.6 Auxiliary Port Direction Parameter
auxIOPortDirection OBJECT-TYPE
SYNTAX      INTEGER { output (1),
                    input (2),
                    bidirectional (3)}
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> Indicates whether state of this port can be set (output),
  read (input) or both (bidirectional).

  <Informative> The ACCESS has been changed from what originally appeared in
  NTCIP 1203 v01 because it was an error.

  <Superseded by> auxIOv2PortDirection

  <Object Identifier> 1.3.6.1.4.1.1206.2.2.1.3.1.6"
 ::= { auxIOEntry 6 }
END
```

## Section 5 DEPRECATED SNMP MIB

The text provided from Section 5.1 through the end of Section 5 (except the headings) constitutes the standard NTCIP1201-SnmpConfig MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 5.1 HEADER

```
NTCIP1201-SnmpConfig DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, Integer32
    FROM SNMPv2-SMI
    -- RFC 2578

    application
    FROM NTCIP8004-Transportation;

snmpConfig MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of
    snmpMaxPacketSize, which was previously defined in NTCIP 1103. This object
    is no longer needed with SNMPv3.

    *** All objects in this MIB have been deprecated. ***

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.1"
REVISION "20230701145436Z"
DESCRIPTION
    "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
    "NTCIP 1103 v03 - No change."
REVISION "200903310000Z"
DESCRIPTION
    "NTCIP 1103 v02 - Changed name to snmpMaxPacketSize. Separated SNMP object
    into its own MIB."
REVISION "200409270000Z"
DESCRIPTION
    "NTCIP 1103 v01: Original version with object named snmp-maxPacketSize."
 ::= { application 1 }
```

### 5.2 OBJECTS

#### 5.2.1 Max Packet Size

```
snmpMaxPacketSize OBJECT-TYPE
```

```
SYNTAX      Integer32 (484..65535)
UNITS       "octets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> Indicates the maximum packet size, in octets, that the SNMP
  agent supports for reception or transmission.

  <Supplanted by> SNMP-FRAMEWORK-MIB.snmpEngineMaxMessageSize (RFC 3411)

  <Informative> This object is no longer needed within SNMPv3 because the
  maximum packet size is contained in the HeaderData of each SNMPv3 data
  packet, as defined in RFC 3412. Nonetheless, RFC 3411 does define an object
  that can be retrieved to report the value that is applicable across all
  connections.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.1.1"
 ::= { snmpConfig 1 }
END
```

## Section 6 DEPRECATED SFMP MIB

The text provided from Section 6.1 through the end of Section 6 (except the headings) constitutes the standard NTCIP1201-Sfmp MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 6.1 HEADER

```
NTCIP1201-Sfmp DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, Counter32
FROM SNMPv2-SMI
    -- RFC 2578

    application
    FROM NTCIP8004-Transportation;

sfmp MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of the
    NTCIP1103v0352-SFMP MIB, which is defined in NTCIP 1103. It defines objects
    related to managing and monitoring the communication statistics for the
    Simple Fixed Message Protocol. (SFMP).

    *** All objects in this MIB have been deprecated. ***

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2"
REVISION "20230701145436Z"
DESCRIPTION
    "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
    "NTCIP 1103 v03 - No change."
REVISION "200903310000Z"
DESCRIPTION
    "NTCIP 1103 v02 - Updated text to conform to current conventions; separated
    SFMP objects into their own MIB."
REVISION "200409270000Z"
DESCRIPTION
    "NTCIP 1103 v01: Original version of objects."
 ::= { application 2 }
```



## 6.2 OBJECT IDENTITIES

### 6.2.1 SFMP Statistics

```
sfmpStatistics OBJECT-IDENTITY
STATUS          deprecated
DESCRIPTION
  "<Definition> This node contains communication statistics for the Simple
  Fixed Message Protocol (SFMP).
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1"
 ::= { sfmp 1 }
```

## 6.3 OBJECTS

### 6.3.1 Number of Incoming SFMP Packets

```
sfmpInPkts OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of Messages delivered to the SFMP entity for
  processing.
  <Informative> This object has been deprecated along with SFMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.1"
 ::= { sfmpStatistics 1 }
```

### 6.3.2 Number of Outgoing SFMP Packets

```
sfmpOutPkts OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP PDU's which were generated by the
  SFMP protocol entity.
  <Informative> This object has been deprecated along with SFMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.2"
 ::= { sfmpStatistics 2 }
```

### 6.3.3 Number of Incoming SFMP Packets with Bad Version Numbers

```
sfmpInBadVersions OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP Messages which were delivered to the
  SFMP protocol entity and were for an unsupported SFMP version.
  <Informative> This object has been deprecated along with SFMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.3"
 ::= { sfmpStatistics 3 }
```

#### 6.3.4 Number of Incoming SFMP Packets with Bad Community Names

```
sfmpInBadCommunityNames OBJECT-TYPE
SYNTAX Counter32
UNITS "packets"
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
    "<Definition> The total number of SFMP Messages delivered to the SFMP
    protocol entity which used a SFMP community name not known to said entity.
    <Informative> This object has been deprecated along with SFMP.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.4"
 ::= { sfmpStatistics 4 }
```

#### 6.3.5 Number of Incoming SFMP Packets with Bad Use of a Community Name

```
sfmpInBadCommunityUses OBJECT-TYPE
SYNTAX Counter32
UNITS "packets"
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
    "<Definition> The total number of SFMP Messages delivered to the SFMP
    protocol entity which represented an SFMP operation which was not allowed
    by the SFMP community named in the Message.
    <Informative> This object has been deprecated along with SFMP.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.5"
 ::= { sfmpStatistics 5 }
```

#### 6.3.6 Number of Incoming SFMP Packets with Parsing Errors

```
sfmpInParseErrs OBJECT-TYPE
SYNTAX Counter32
UNITS "packets"
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
    "<Definition> The total number of OER errors encountered by the SFMP
    protocol entity when decoding received SFMP Messages.
    <Informative> This object has been deprecated along with SFMP.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.6"
 ::= { sfmpStatistics 6 }
```

#### 6.3.7 Reserved

```
-- node 7 is reserved for bad types to parallel SNMP, but it does not
-- apply to SFMP
```

#### 6.3.8 Number of Incoming SFMP Packets indicating a Too Big Error

```
sfmpInTooBig OBJECT-TYPE
SYNTAX Counter32
UNITS "packets"
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
```

"<Definition> The total number of SFMP PDUs which were delivered to the SFMP protocol entity with a Message Type of Error and Error Number of tooBig.

<Informative> This object has been deprecated along with SFMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.8"

::= { sfmpStatistics 8 }

### 6.3.9 Number of Incoming SFMP Packets indicating a No Such Name Error

sfmpInNoSuchNames OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of SFMP PDUs which were delivered to the SFMP protocol entity with a Message Type of Error and Error Number of noSuchName.

<Informative> This object has been deprecated along with SFMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.9"

::= { sfmpStatistics 9 }

### 6.3.10 Number of Incoming SFMP Packets indicating a Bad Value Error

sfmpInBadValues OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of SFMP PDUs which were delivered to the SFMP protocol entity with a Message Type of Error and Error Number of badValue.

<Informative> This object has been deprecated along with SFMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.10"

::= { sfmpStatistics 10 }

### 6.3.11 Number of Incoming SFMP Packets indicating a Read-Only Error

sfmpInReadOnly OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of SFMP PDUs which were delivered to the SFMP protocol entity with a Message Type of Error and Error Number of readOnly.

<Informative> This object has been deprecated along with SFMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.11"

::= { sfmpStatistics 11 }

### 6.3.12 Number of Incoming SFMP Packets indicating a General Error

sfmpInGenErrs OBJECT-TYPE

```
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP PDUs which were delivered to the
  SFMP protocol entity with a Message Type of Error and Error Number of
  genError.

  <Informative> This object has been deprecated along with SFMP.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.12"
 ::= { sfmpStatistics 12 }
```

### 6.3.13 Reserved

```
-- node 13 is reserved for total request vars to
-- parallel SNMP, but it does not apply to SFMP
```

### 6.3.14 Reserved

```
-- node 14 is reserved for total set vars to parallel
-- SNMP, but it does not apply to SFMP
```

### 6.3.15 Number of Incoming SFMP Get Requests

```
sfmpInGetRequests OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP Get-Request PDUs which have been
  accepted and processed by the SFMP protocol entity.

  <Informative> This object has been deprecated along with SFMP.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.15"
 ::= { sfmpStatistics 15 }
```

### 6.3.16 Reserved

```
-- node 16 is reserved for in get nexts to parallel
-- SNMP, but it does not apply to SFMP
```

### 6.3.17 Number of Incoming SFMP Set Requests

```
sfmpInSetRequests OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP Set-Request PDUs which have been
  accepted and processed by the SFMP protocol entity.

  <Informative> This object has been deprecated along with SFMP.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.17"
 ::= { sfmpStatistics 17 }
```

### 6.3.18 Number of Incoming SFMP Get Responses

```
sfmpInGetResponses OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
```

```
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
  "<Definition> The total number of SFMP Get-Response PDUs which have been
  accepted and processed by the SFMP protocol entity.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.18"
 ::= { sfmpStatistics 18 }
```

### 6.3.19 Reserved

```
-- node 19 is reserved for traps to parallel SNMP, but it
-- does not apply to SFMP at present
```

### 6.3.20 Number of Outgoing SFMP Packets indicating a Too Big Error

```
sfmpOutTooBig OBJECT-TYPE
SYNTAX Counter32
UNITS "packets"
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
  "<Definition> The total number of SFMP PDUs which were generated by the
  SFMP protocol entity with a Message Type of Error and Error Number of
  tooBig.
  <Informative> This object has been deprecated along with SFMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.20"
 ::= { sfmpStatistics 20 }
```

### 6.3.21 Number of Outgoing SFMP Packets indicating a No Such Name Error

```
sfmpOutNoSuchNames OBJECT-TYPE
SYNTAX Counter32
UNITS "packets"
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
  "<Definition> The total number of SFMP PDUs which were generated by the
  SFMP protocol entity with a Message Type of Error and Error Number of
  noSuchname.
  <Informative> This object has been deprecated along with SFMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.21"
 ::= { sfmpStatistics 21 }
```

### 6.3.22 Number of Outgoing SFMP Packets indicating a Bad Value Error

```
sfmpOutBadValues OBJECT-TYPE
SYNTAX Counter32
UNITS "packets"
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
  "<Definition> The total number of SFMP PDUs which were generated by the
  SFMP protocol entity with a Message Type of Error and Error Number of
  badValue.
  <Informative> This object has been deprecated along with SFMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.22"
```

```
::= { sfmpStatistics 22 }
```

### 6.3.23 Number of Outgoing SFMP Packets indicating a Read-Only Error

```
sfmpOutReadOnly OBJECT-TYPE  
SYNTAX Counter32  
UNITS "packets"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
    "<Definition> The total number of SFMP PDUs which were generated by the  
    SFMP protocol entity with a Message Type of Error and Error Number of  
    readOnly.  
  
    <Informative> This object has been deprecated along with SFMP.  
  
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.23"
```

```
::= { sfmpStatistics 23 }
```

### 6.3.24 Number of Outgoing SFMP Packets indicating a General Error

```
sfmpOutGenError OBJECT-TYPE  
SYNTAX Counter32  
UNITS "packets"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
    "<Definition> The total number of SFMP PDUs which were generated by the  
    SFMP protocol entity with a Message Type of Error and Error Number of  
    genErr.  
  
    <Informative> This object has been deprecated along with SFMP.  
  
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.24"
```

```
::= { sfmpStatistics 24 }
```

### 6.3.25 Number of Outgoing SFMP Get Requests

```
sfmpOutGetRequests OBJECT-TYPE  
SYNTAX Counter32  
UNITS "packets"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
    "<Definition> The total number of SFMP PDU's with a Message Type of Get-  
    Request, which have been generated by the SFMP protocol entity.  
  
    <Informative> This object has been deprecated along with SFMP.  
  
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.25"
```

```
::= { sfmpStatistics 25 }
```

### 6.3.26 Reserved

```
-- node 26 is reserved for out get nexts to parallel SNMP,  
-- but it does not apply to SFMP
```

### 6.3.27 Number of Outgoing SFMP Set Requests

```
sfmpOutSetRequests OBJECT-TYPE  
SYNTAX Counter32  
UNITS "packets"  
MAX-ACCESS read-only  
STATUS deprecated
```

DESCRIPTION

"<Definition> The total number of SFMP PDU's with a Message Type of Set-Request, which have been generated by the SFMP protocol entity.

<Informative> This object has been deprecated along with SFMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.27"

::= { sfmpStatistics 27 }

**6.3.28 Number of Outgoing SFMP Get Responses**

sfmpOutGetResponses OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of SFMP PDU's with a Message Type of Get-Response, which have been generated by the SFMP protocol entity.

<Informative> This object has been deprecated along with SFMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.28"

::= { sfmpStatistics 28 }

**6.3.29 Number of Outgoing SFMP Trap Messages**

sfmpOutTrapMessages OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of SFMP PDUs with a message type of Trap that have been generated by the SFMP protocol entity.

<Informative> This object has been deprecated along with SFMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.29"

::= { sfmpStatistics 29 }

**6.3.30 Reserved**

-- node 30 is reserved for enable authentication traps to parallel  
-- SNMP, but it does not apply to SFMP

**6.3.31 Number of Incoming SFMP Set Requests - No Replies**

sfmpInSetRequestsNoReply OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of SFMP Set-Request No Reply PDUs which have been accepted and processed by the SFMP protocol entity.

<Informative> This object has been deprecated along with SFMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.31"

::= { sfmpStatistics 31 }

**6.3.32 Number of Incoming SFMP Set Responses**

sfmpInSetResponses OBJECT-TYPE

```
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP Set-Response PDUs which have been
  accepted and processed by the SFMP protocol entity.

  <Informative> This object has been deprecated along with SFMP.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.32"
 ::= { sfmpStatistics 32 }
```

### 6.3.33 Number of Incoming SFMP Error Responses

```
sfmpInErrorResponses OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP Error-Response PDUs which have been
  accepted and processed by the SFMP protocol entity.

  <Informative> This object has been deprecated along with SFMP.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.33"
 ::= { sfmpStatistics 33 }
```

### 6.3.34 Number of Outgoing SFMP Set Requests - No Replies

```
sfmpOutSetRequestsNoReply OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP PDU's with a Message Type of Set-
  Request-No-Reply, which have been generated by the SFMP protocol entity.

  <Informative> This object has been deprecated along with SFMP.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.34"
 ::= { sfmpStatistics 34 }
```

### 6.3.35 Number of Outgoing SFMP Set Responses

```
sfmpOutSetResponses OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of SFMP PDU's with a Message Type of Set-
  Response, which have been generated by the SFMP protocol entity.

  <Informative> This object has been deprecated along with SFMP.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.35"
 ::= { sfmpStatistics 35 }
```



### 6.3.36 Number of Outgoing SFMP Error Responses

```
sfpOutErrorResponses OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
    "<Definition> The total number of SFMP PDU's with a Message Type of Error-
    Response, which have been generated by the SFMP protocol entity.
    <Informative> This object has been deprecated along with SFMP.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.1.36"
 ::= { sfpStatistics 36 }
END
```

## Section 7 DEPRECATED DYNAMIC OBJECT MIB

The text provided from Section 7.1 through the end of Section 7 (except the headings) constitutes the standard NTCIP1201-DynObjMgmt MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 7.1 HEADER

```
NTCIP1201-DynObjMgmt DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, Integer32, zeroDotZero
FROM SNMPv2-SMI
    -- RFC 2578

    TEXTUAL-CONVENTION
    FROM SNMPv2-TC
    -- RFC 2579

    protocols, NtcipOwnerString
    FROM NTCIP8004-Transportation;

dynObjMgmt MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of the
    NTCIP1103v0352-STMP MIB, which was previously defined in NTCIP 1103. This
    node contains management information related to dynamic objects. This node
    was deprecated in 2021 as a part of the NTCIP 8004v03 update because
    dynamic objects are a specific feature of the Simple Transportation
    Management Protocol (STMP), which was itself deprecated because it was
    deemed to provide insufficient security for modern networks.
    *** All objects in this MIB have been deprecated. ***

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.3"
REVISION "20230701145436Z"
DESCRIPTION
    "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
    "NTCIP 1103 v03 - No change."
REVISION "200903310000Z"
DESCRIPTION
    "NTCIP 1103 v02 - ."
REVISION "199609270000Z"
DESCRIPTION
    "NTCIP 1101 v01: Original version of these objects."
 ::= { protocols 3 }
```

## 7.2 TEXTUAL CONVENTIONS

```
ConfigEntryStatus ::= TEXTUAL-CONVENTION
STATUS      deprecated
DESCRIPTION
  "See Clause 5.2.4.1 of NTCIP 1103 v03 for the complete definition of this
  type."
SYNTAX      INTEGER { valid (1),
                     underCreation (2),
                     invalid (3) }
EntryStatus ::= TEXTUAL-CONVENTION
STATUS      deprecated
DESCRIPTION
  "See Annex E of NTCIP 1103 v03 for a complete definition of this Type."
SYNTAX      INTEGER { valid (1),
                     createRequest (2),
                     underCreation (3),
                     invalid (4) }
```

## 7.3 OBJECT IDENTITIES

### 7.3.1 Data Node

```
dynObjData OBJECT-IDENTITY
STATUS      deprecated
DESCRIPTION
  "<Definition> A node containing SNMP object definitions of the dynamic
  objects.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2"
::= { dynObjMgmt 2 }
```

## 7.4 DYNAMIC OBJECT DEFINITION

### 7.4.1 Maximum Dynamic Object Table Entries

```
dynObjDefTableMaxEntries OBJECT-TYPE
SYNTAX      Integer32 (1..255)
UNITS       "entries"
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> This object specifies the maximum number of rows that may be
  implemented in the Dynamic Object Definition table.
  <Informative> This object has been deprecated along with STMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.4"
::= { dynObjMgmt 4 }
```

### 7.4.2 Dynamic Object Definition Table

```
dynObjDef OBJECT-TYPE
SYNTAX      SEQUENCE OF DynObjEntry
MAX-ACCESS not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition> A list of objects to be included in dynamic objects
  <Table Type> static"
```

```
<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldTable
<Informative> This object has been deprecated along with STMP.
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.1"
 ::= { dynObjMgmt 1 }

dynObjEntry OBJECT-TYPE
SYNTAX      DynObjEntry
MAX-ACCESS not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition> A list of OBJECT IDENTIFIERS that make up a dynamic object
  <Supplanted by> ISO26048-1-DynObj.fdDynObjFieldEntry
  <Informative> This object has been deprecated along with STMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.1.1"
INDEX       { dynObjNumber, dynObjIndex }
 ::= { dynObjDef 1 }

DynObjEntry ::= SEQUENCE {
    dynObjNumber      Integer32,
    dynObjIndex       Integer32,
    dynObjVariable    OBJECT IDENTIFIER,
    dynObjOwner       NtcipOwnerString, -- previously deprecated
    dynObjStatus      EntryStatus }      -- previously deprecated}
```

#### 7.4.2.1 Dynamic Object Number

```
dynObjNumber OBJECT-TYPE
SYNTAX      Integer32 ( 1..13)
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The dynamic object number that this entry is to be associated
  with.
  <Supplanted by> ISO26048-1-DynObj.fdDynObjIndex
  <Informative> This object has been deprecated along with STMP.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.1.1.1"
 ::= { dynObjEntry 1 }
```

#### 7.4.2.2 Dynamic Object Index

```
dynObjIndex OBJECT-TYPE
SYNTAX      Integer32 (1..255)
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> An index that uniquely identifies an entry in the dynamic
  object table. Each entry defines an object that is to be associated with a
  dynamic object number. The dynObjIndex determines the order in which
  objects are transmitted for the associated dynamic object. The lower
  dynObjIndex numbers are transmitted before larger numbers for entries
  within the same dynamic object.
  <Supplanted by> ISO26048-1-DynObj.fdDynObjFieldIndex
```

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.1.1.2"

::= { dynObjEntry 2 }

#### 7.4.2.3 Dynamic Object Variable

dynObjVariable OBJECT-TYPE

SYNTAX OBJECT IDENTIFIER

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> The complete object identifier of the particular variable to be included in the specified dynamic object number. When defining dynamic objects, the maximum size of all the objects included in a dynamic object shall not exceed the maximum packet size of the communications network. When set to reference a columnar object, an agent may wish to only validate the prefix portion of the object identifier. The suffix or index portion of an object identifier need not be instantiated or exist at the time a dynObjVariable is defined.

This object shall not reference any of the objects identified in NTCIP 1103 Clause 8.2.

This object may not be modified unless the associated dynObjConfigStatus object is equal to underCreation.

<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldObject

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.1.1.3"

DEFVAL { zeroDotZero }

::= { dynObjEntry 3 }

#### 7.4.2.4 Dynamic Object Owner

dynObjOwner OBJECT-TYPE

SYNTAX NtcipOwnerString

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> This object has been replaced with dynObjConfigOwner. The entity that configured this entry and is therefore using the resources assigned to it. This object may not be modified if the associated dynObjStatus object is equal to valid(1).

<Informative> This object was deprecated in NTCIP 1103 v02.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.1.1.4"

::= { dynObjEntry 4 }

#### 7.4.2.5 Dynamic Object Status

dynObjStatus OBJECT-TYPE

SYNTAX EntryStatus

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> This object has been replaced with dynObjConfigStatus. The status of this dynamic object definition entry. See description of EntryStatus above for restrictions on accesses.

<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldRowStatus

<Informative> This object was deprecated in NTCIP 1103 v02.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.1.1.5"

::= { dynObjEntry 5 }

## 7.5 DYNAMIC OBJECT INFORMATION

### 7.5.1 Dynamic Object Data

#### 7.5.1.1 Dynamic Object 1

dynObj1 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 1. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent.

<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.1"

::= { dynObjData 1 }

#### 7.5.1.2 Dynamic Object 2

dynObj2 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 2. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent

<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.2"

::= { dynObjData 2 }

#### 7.5.1.3 Dynamic Object 3

dynObj3 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 3. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little

advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.3"

::= { dynObjData 3 }

#### 7.5.1.4 Dynamic Object 4

dynObj4 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 4. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.4"

::= { dynObjData 4 }

#### 7.5.1.5 Dynamic Object 5

dynObj5 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 5. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.5"

::= { dynObjData 5 }

#### 7.5.1.6 Dynamic Object 6

dynObj6 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 6. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little

advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.6"

::= { dynObjData 6 }

#### 7.5.1.7 Dynamic Object 7

dynObj7 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 7. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.7"

::= { dynObjData 7 }

#### 7.5.1.8 Dynamic Object 8

dynObj8 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 8. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.8"

::= { dynObjData 8 }

#### 7.5.1.9 Dynamic Object 9

dynObj9 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 9. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little



advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.9"

::= { dynObjData 9 }

#### 7.5.1.10 Dynamic Object 10

dynObj10 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 10. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.10"

::= { dynObjData 10 }

#### 7.5.1.11 Dynamic Object 11

dynObj11 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 11. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue  
<Informative> This object has been deprecated along with STMP.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.11"

::= { dynObjData 11 }

#### 7.5.1.12 Dynamic Object 12

dynObj12 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 12. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little

advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent

<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.12"

::= { dynObjData 12 }

### 7.5.1.13 Dynamic Object 13

dynObj13 OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> The value of this object is determined by the dynObjDef entries with dynObjNumber equal to 13. Packed Encoding Rules are utilized to encode the objects for transmission. This object is intended for use with the Simple Transportation Management Protocol, and provides little advantage if used with SNMP. If no objects are defined for this dynamic object number, then an error of noSuchName shall be returned by the agent

<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue and fdDynObjNewValue

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.2.13"

::= { dynObjData 13 }

## 7.5.2 Dynamic Object Configuration

dynObjConfigTable OBJECT-TYPE  
SYNTAX SEQUENCE OF DynObjConfigEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION

"<Definition> A table consisting of an owner and status for each of the 13 dynamic object definitions.

<Table Type> static

<Supplanted by> ISO26048-1-DynObj.fdDynObjTable

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.3"

::= { dynObjMgmt 3 }

dynObjConfigEntry OBJECT-TYPE  
SYNTAX DynObjConfigEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION

"<Definition> A table consisting of an owner and status for each of the 13 dynamic object definitions.

<Supplanted by> ISO26048-1-DynObj.fdDynObjEntry

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.3.1"

```
INDEX      { dynObjNumber }  
::= { dynObjConfigTable 1 }
```

```
DynObjConfigEntry ::= SEQUENCE {  
    dynObjConfigOwner  NtcipOwnerString,  
    dynObjConfigStatus ConfigEntryStatus}
```

#### 7.5.2.1 Dynamic Object Configuration Owner

dynObjConfigOwner OBJECT-TYPE

SYNTAX NtcipOwnerString

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> The entity that configured the associated dynamic object.  
This object may not be modified unless the associated dynObjConfigStatus  
object is equal to underCreation.

<Supplanted by> ISO26048-1-Owner.fdOwnerIndex

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.3.1.1"

```
DEFVAL { "" }  
::= { dynObjConfigEntry 1 }
```

#### 7.5.2.2 Dynamic Object Configuration Status

dynObjConfigStatus OBJECT-TYPE

SYNTAX ConfigEntryStatus

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> Indicates the state of the associated dynamic object.  
Depending on the validity checks that are performed on the dynamic object  
definition, a set request may or may not be honored. See Clause 5.2.4.1 for  
a complete description.

<Supplanted by> ISO26048-1-DynObj.fdDynObjRowStatus

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.3.3.1.2"

```
::= { dynObjConfigEntry 2 }  
END
```

## Section 8 DEPRECATED STMP MIB

The text provided from Section 8.1 through the end of Section 8 (except the headings) constitutes the standard NTCIP1201-Stmp MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 8.1 HEADER

```
NTCIP1201-Stmp DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, Counter32
FROM SNMPv2-SMI
    -- RFC 2578

    application
        FROM NTCIP8004-Transportation;

stmp MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of the
    NTCIP1103v0352-STMP-Stats MIB, which was defined in NTCIP 1103. This MIB
    defines objects related to communication statistics for the Simple
    Transportation Management Protocol (STMP).
    This MIB was deprecated in NTCIP 1201 v04 because the Simple Transportation
    Management Protocol (STMP) was deprecated because it was deemed to provide
    insufficient security for modern networks.
    *** All objects in this MIB have been deprecated. ***

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3"
REVISION "20230701145436Z"
DESCRIPTION
    "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
    "NTCIP 1103 v03 - No change."
REVISION "200903310000Z"
DESCRIPTION
    "NTCIP 1103 v02 - Changed name to snmpMaxPacketSize. Separated SNMP object
    into its own MIB."
REVISION "200409270000Z"
DESCRIPTION
    "NTCIP 1103 v01: Original version with object named snmp-maxPacketSize."
::= { application 3 }
```

## 8.2 OBJECT IDENTITIES

### 8.2.1 STMP Statistics Node

```
stmpStatistics OBJECT-IDENTITY
STATUS      deprecated
DESCRIPTION
  "<Definition> This node contains communication statistics for the Simple
  Transportation Management Protocol (STMP).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1"
 ::= { stmp 1 }
```

## 8.3 OBJECTS

### 8.3.1 Number of Incoming STMP Packets

```
stmpInPkts OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of Messages delivered to the STMP entity for
  processing.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.1"
 ::= { stmpStatistics 1 }
```

### 8.3.2 Number of Outgoing STMP Packets

```
stmpOutPkts OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDU's which were generated by the
  STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.2"
 ::= { stmpStatistics 2 }
```

### 8.3.3 Reserved

```
-- node 3 is reserved for bad version to parallel SNMP,
-- but it does not apply to STMP
```

### 8.3.4 Reserved

-- node 4 is reserved for bad community name to parallel  
-- SNMP, but it does not apply to STMP

### 8.3.5 Reserved

-- node 5 is reserved for bad community use to parallel  
-- SNMP, but it does not apply to STMP

### 8.3.6 Number of Incoming STMP Packets with Parsing Errors

```
stmpInParseErrs OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of OER errors encountered by the STMP
  protocol entity when decoding received STMP Messages.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.6"
 ::= { stmpStatistics 6 }
```

### 8.3.7 Reserved

-- node 7 is reserved for bad types to parallel SNMP, but  
-- it does not apply to STMP

### 8.3.8 Number of Incoming STMP Packets indicating a Too Big Error

```
stmpInTooBig OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDUs which were delivered to the
  STMP protocol entity with a Message Type of Error and Error Number of
  tooBig.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.8"
 ::= { stmpStatistics 8 }
```

### 8.3.9 Number of Incoming STMP Packets indicating a No Such Name Error

```
stmpInNoSuchNames OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
```

DESCRIPTION

"<Definition> The total number of STMP PDUs which were delivered to the STMP protocol entity with a Message Type of Error and Error Number of noSuchName.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.9"

::= { stmpStatistics 9 }

**8.3.10 Number of Incoming STMP Packets indicating a Bad Value Error**

stmpInBadValues OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP PDUs which were delivered to the STMP protocol entity with a Message Type of Error and Error Number of badValue.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.10"

::= { stmpStatistics 10 }

**8.3.11 Number of Incoming STMP Packets indicating a Read-Only Error**

stmpInReadOnlys OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP PDUs which were delivered to the STMP protocol entity with a Message Type of Error and Error Number of readOnly.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.11"

::= { stmpStatistics 11 }

**8.3.12 Number of Incoming STMP Packets indicating a General Error**

stmpInGenErrs OBJECT-TYPE

```
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDUs which were delivered to the
  STMP protocol entity with a Message Type of Error and Error Number of
  genError.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.12"
 ::= { stmpStatistics 12 }
```

### 8.3.13 Reserved

```
-- node 13 is reserved for total request vars to parallel
-- SNMP, but it does not apply to STMP
```

### 8.3.14 Reserved

```
-- node 14 is reserved for total set vars to parallel SNMP,
-- but it does not apply to STMP
```

### 8.3.15 Number of Incoming STMP Get Requests

```
stmpInGetRequests OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP Get-Request PDUs which have been
  accepted and processed by the STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.15"
 ::= { stmpStatistics 15 }
```

### 8.3.16 Number of Incoming STMP Get Next Requests

```
stmpInGetNexts OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP Get-Next PDUs which have been
  accepted and processed by the STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
```



syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.16"

::= { stmpStatistics 16 }

### 8.3.17 Number of Incoming STMP Set Requests

stmpInSetRequests OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP Set-Request PDUs which have been accepted and processed by the STMP protocol entity.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.17"

::= { stmpStatistics 17 }

### 8.3.18 Number of Incoming STMP Get Responses

stmpInGetResponses OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP Get-Response PDUs which have been accepted and processed by the STMP protocol entity.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.18"

::= { stmpStatistics 18 }

### 8.3.19 Reserved

-- node 19 is reserved for in trap responses to parallel

-- SNMP, but it does not apply to STMP

### 8.3.20 Number of Outgoing STMP Packets indicating a Too Big Error

stmpOutTooBigs OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP PDUs which were generated by the STMP protocol entity with a Message Type of Error and Error Number of tooBig.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.20"

::= { stmpStatistics 20 }

### 8.3.21 Number of Outgoing STMP Packets indicating a No Such Name Error

stmpOutNoSuchNames OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP PDUs which were generated by the STMP protocol entity with a Message Type of Error and Error Number of noSuchName.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.21"

::= { stmpStatistics 21 }

### 8.3.22 Number of Outgoing STMP Packets indicating a Bad Value Error

stmpOutBadValues OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP PDUs which were generated by the STMP protocol entity with a Message Type of Error and Error Number of badValue.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.22"

::= { stmpStatistics 22 }

### 8.3.23 Number of Outgoing STMP Packets indicating a Read-Only Error

stmpOutReadOnly OBJECT-TYPE

SYNTAX Counter32

```
UNITS      "packets"
MAX-ACCESS read-only
STATUS     deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDUs which were generated by the
  STMP protocol entity with a Message Type of Error and Error Number of
  readOnly.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.23"
 ::= { stmpStatistics 23 }
```

### 8.3.24 Number of Outgoing STMP Packets indicating a General Error

```
stmpOutGenError OBJECT-TYPE
SYNTAX      Counter32
UNITS      "packets"
MAX-ACCESS read-only
STATUS     deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDUs which were generated by the
  STMP protocol entity with a Message Type of Error and Error Number of
  genErr.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.24"
 ::= { stmpStatistics 24 }
```

### 8.3.25 Number of Outgoing STMP Get Requests

```
stmpOutGetRequests OBJECT-TYPE
SYNTAX      Counter32
UNITS      "packets"
MAX-ACCESS read-only
STATUS     deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDU's with a Message Type of Get-
  Request, which have been generated by the STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.25"
 ::= { stmpStatistics 25 }
```

### 8.3.26 Number of Outgoing STMP Get Next Requests

```
stmpOutGetNexts OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDU's with a Message Type of Get-
  Next, which have been generated by the STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.26"
 ::= { stmpStatistics 26 }
```

### 8.3.27 Number of Outgoing STMP Set Requests

```
stmpOutSetRequests OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDU's with a Message Type of Set-
  Request, which have been generated by the STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.27"
 ::= { stmpStatistics 27 }
```

### 8.3.28 Number of Outgoing STMP Get Responses

```
stmpOutGetResponses OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP PDU's with a Message Type of Get-
  Response, which have been generated by the STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.28"
 ::= { stmpStatistics 28 }
```

### 8.3.29 Reserved

-- node 29 is reserved for in trap responses to parallel  
-- SNMP, but it does not apply to STMP

### 8.3.30 Reserved

-- node 30 is reserved for enable authentication traps to parallel  
-- SNMP, but it does not apply to STMP

### 8.3.31 Number of Incoming STMP Set Request - No Replies

```
stmpInSetRequestsNoReply OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP Set-Request No Reply PDUs which have
  been accepted and processed by the STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.31"
 ::= { stmpStatistics 31 }
```

### 8.3.32 Number of Incoming STMP Set Responses

```
stmpInSetResponses OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP Set-Response PDUs which have been
  accepted and processed by the STMP protocol entity.

  <Informative> This object has been deprecated along with STMP. This object
  was originally defined with a syntax of Counter. This MIB updates the
  syntax to Counter32 to conform to SMIV2 conventions. There is no difference
  between the Counter and Counter32 syntaxes. Users should note, however,
  that neither a Counter nor a Counter32 is Required to initialize at or
  reset to zero (0).

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.32"
 ::= { stmpStatistics 32 }
```

### 8.3.33 Number of Incoming STMP Error Responses

```
stmpInErrorResponses OBJECT-TYPE
SYNTAX      Counter32
UNITS       "packets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The total number of STMP Error-Response PDUs which have been
  accepted and processed by the STMP protocol entity.
```

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.33"

::= { stmpStatistics 33 }

### 8.3.34 Number of Outgoing STMP Set Request - No Replies

stmpOutSetRequestsNoReply OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP PDU's with a Message Type of Set-Request-No-Reply, which have been generated by the STMP protocol entity.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.34"

::= { stmpStatistics 34 }

### 8.3.35 Number of Outgoing STMP Set Responses

stmpOutSetResponses OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP PDU's with a Message Type of Set-Response, which have been generated by the STMP protocol entity.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.35"

::= { stmpStatistics 35 }

### 8.3.36 Number of Outgoing STMP Error Responses

stmpOutErrorResponses OBJECT-TYPE

SYNTAX Counter32

UNITS "packets"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The total number of STMP PDU's with a Message Type of Error-Response, which have been generated by the STMP protocol entity.

<Informative> This object has been deprecated along with STMP. This object was originally defined with a syntax of Counter. This MIB updates the syntax to Counter32 to conform to SMIV2 conventions. There is no difference between the Counter and Counter32 syntaxes. Users should note, however, that neither a Counter nor a Counter32 is Required to initialize at or reset to zero (0).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.1.36"

```
::= { stmpStatistics 36 }  
END
```

## Section 9 DEPRECATED STMP CONFIGURATION MIB

The text provided from Section 9.1 through the end of Section 9 (except the headings) constitutes the standard NTCIP1201-ProfilesSTMP MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 9.1 HEADER

```
NTCIP1201-ProfilesSTMP DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, Integer32
FROM SNMPv2-SMI
    -- RFC 2578

profiles
    FROM NTCIP8004-Transportation;

profilesSTMP MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of the
    NTCIP1103v0352-STMP-Config MIB, which was defined in NTCIP 1103. This MIB
    defines objects related to the configuration of the Simple Transportation
    Management Protocol (STMP). This MIB was deprecated in NTCIP 1201 v04
    because the Simple Transportation Management Protocol (STMP) was deprecated
    because it was deemed to provide insufficient security for modern networks.
    *** All objects in this MIB have been deprecated. ***

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.2"
REVISION "20230701145436Z"
DESCRIPTION
    "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
    "NTCIP 1103 v03 - No change."
REVISION "200903310000Z"
DESCRIPTION
    "NTCIP 1103 v02 - Changed name to snmpMaxPacketSize. Separated SNMP object
    into its own MIB."
REVISION "200409270000Z"
DESCRIPTION
    "NTCIP 1103 v01: Original version with object named snmp-maxPacketSize."
 ::= { profiles 2 }
```



## 9.2 OBJECTS

### 9.2.1 Dynamic Object Persistence

dynamicObjectPersistence OBJECT-TYPE

SYNTAX Integer32 (0..65535)

UNITS "minutes"

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> The maximum power outage time in minutes that may occur before all STMP dynamic object definitions in a device shall be invalidated. If this object is set to zero then the existing dynamic object definitions shall be invalidated on device power up. If this object is set to its maximum value (65535), then the existing dynamic object definitions shall nominally persist for an infinite period (in practice this is limited by the non-volatile memory capabilities of the device). This object shall not be invalidated due to power outages of any duration. A device that supports STMP dynamic objects shall support this object.

<Supplanted by> ISO26048-1-DynObj.fdOwnerDynObjPersistence

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.2.2.1"

DEFVAL { 65535 }

::= { profilesSTMP 1 }

### 9.2.2 Dynamic Object Configuration ID

dynamicObjectTableConfigID OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> Specifies a relatively unique ID (e.g., this could be a counter, a check-sum, etc.) for the current values stored in the dynObjVariable and dynObjConfigOwner objects for all dynamic objects with a dynObjStatus of valid. This value shall be calculated on the change of any dynObjStatus to or from the valid state. This value reported by this object shall not change unless there has been a change in the data since the last request; however a genErr shall be returned if the unique ID value has not yet been updated. A management station will be able to detect any change in the configuration of dynamic objects by monitoring this value after it has established a baseline.

<Supplanted by> ISO26048-1-DynObj.fdOwnerDynObjConfigID

<Informative> This object has been deprecated along with STMP.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.2.2.2"

::= { profilesSTMP 2 }

END

## Section 10 DEPRECATED LOGICAL NAMES MIB

The text provided from Section 10.1 through the end of Section 10 (except the headings) constitutes the standard NTCIP1201-LogicalNames MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 10.1 HEADER

```
NTCIP1201-LogicalNames DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, Integer32, IpAddress
FROM SNMPv2-SMI
-- RFC 2578

application, NtcipRowStatusStatic
FROM NTCIP8004-Transportation;

logicalNames MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
"name: NTCIP Coordinator
email: ntcip@nema.org
postal: National Electrical Manufacturers Association
1300 North 17th Street, Suite 1752
Rosslyn, Virginia 22209-3801"
DESCRIPTION
"<Definition> This MIB defines the SMIV2 representation of the
NTCIP1103v0352-LogicalNames MIB, which was defined in NTCIP 1103. This MIB
defines various objects related to the mapping of logical device names to
network addresses. This MIB was deprecated in NTCIP 1201 v04 because the
SNMPv3 includes the definition of targets in the SNMP-TARGET-MIB, as
defined in RFC 3413.
*** All objects in this MIB have been deprecated. ***

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4"
REVISION "20230701145436Z"
DESCRIPTION
"NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
"NTCIP 1103 v03 - No change."
REVISION "200903310000Z"
DESCRIPTION
"NTCIP 1103 v02 - Separated into its own MIB."
REVISION "200409270000Z"
DESCRIPTION
"NTCIP 1103 v01: Original version."
 ::= { application 4 }
```

## 10.2 OBJECTS

### 10.2.1 Maximum Logical Name Translations

```
logicalNameTranslationTableMaxEntries OBJECT-TYPE
SYNTAX      Integer32 (1..255)
UNITS       "entries"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
    "<Definition> This object specifies the maximum number of rows that may be
    implemented in the logical name translation table.

    <Informative> This object was deprecated because the SNMPv3 includes the
    definition of targets in the SNMP-TARGET-MIB, as defined in RFC 3413.

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.1"
 ::= { logicalNames 1 }
```

### 10.2.2 Logical Name Translation Table

```
logicalNameTranslationTable OBJECT-TYPE
SYNTAX      SEQUENCE OF LogicalNameTranslationEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
    "<Definition> This table defines the logical names of the other network
    entities with which the device may communicate and maps these names to the
    network addresses of those devices.

    <Table Type> static

    <Supplanted by> SNMP-TARGET-MIB.SnmpTargetAddrTable (RFC3413)

    <Informative> This object was deprecated because the SNMPv3 includes the
    definition of targets in the SNMP-TARGET-MIB, as defined in RFC 3413.

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.2"
 ::= { logicalNames 2 }
```

```
logicalNameTranslationEntry OBJECT-TYPE
SYNTAX      LogicalNameTranslationEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
    "<Definition> This is one logical row of the logical name translation
    table.

    <Supplanted by> SNMP-TARGET-MIB.SnmpTargetAddrEntry (RFC3413)

    <Informative> This object was deprecated because the SNMPv3 includes the
    definition of targets in the SNMP-TARGET-MIB, as defined in RFC 3413.

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.2.1"
```

```
INDEX      { logicalNameTranslationIndex }
 ::= { logicalNameTranslationTable 1 }
```

```
LogicalNameTranslationEntry ::= SEQUENCE {
    logicalNameTranslationIndex      Integer32,
    logicalNameTranslationLogicalName OCTET STRING,
    logicalNameTranslationNetworkAddress IpAddress,
    logicalNameTranslationStatus      NtcipRowStatusStatic}
```

### 10.2.2.1 Index for the Logical Name Translation

logicalNameTranslationIndex OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object provides the index into the logical name table.  
<Supplanted by> SNMP-TARGET-MIB.SnmpTargetAddrName (RFC3413)  
<Informative> This object was deprecated because the SNMPv3 includes the definition of targets in the SNMP-TARGET-MIB, as defined in RFC 3413.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.2.1.1"  
 ::= { logicalNameTranslationEntry 1 }

### 10.2.2.2 Logical Name for the Logical Name Translation

logicalNameTranslationLogicalName OBJECT-TYPE  
SYNTAX OCTET STRING (SIZE (0..32))  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object defines the logical name of the network entity for which this row is defined.  
<Supplanted by> SNMP-TARGET-MIB.SnmpTargetAddrName (RFC3413)  
<Informative> This object was deprecated because the SNMPv3 includes the definition of targets in the SNMP-TARGET-MIB, as defined in RFC 3413.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.2.1.2"  
DEFVAL { "" }  
 ::= { logicalNameTranslationEntry 2 }

### 10.2.2.3 Network Address of the Logical Name Translation

logicalNameTranslationNetworkAddress OBJECT-TYPE  
SYNTAX IPAddress  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object defines the network address of the associated network entity for the given profile. If the transport profile is 'internet, the network address is the IP address of the entity stored as an IPAddress. If the transport profile is 't2, there is no physical network address, but the entity is assigned a dummy IP address in order to abstract the mapping to the ipNetToMediaTable defined in MIB-II.  
<Supplanted by> SNMP-TARGET-MIB.SnmpTargetAddrTAddress (RFC3413)  
<Informative> This object was deprecated because the SNMPv3 includes the definition of targets in the SNMP-TARGET-MIB, as defined in RFC 3413. This object was originally defined with the syntax of NetworkAddress; The NetworkAddress is not defined in SNMPv3 and is actually an alias of IPAddress (i.e., it was defined as CHOICE {internet IPAddress}, which in BER is encoded exactly the same as IPAddress.) The conversion to SMIV2 replaced this syntax with IPAddress.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.2.1.3"

```
DEFVAL      { '00000000'H }  
::= { logicalNameTranslationEntry 3 }
```

#### 10.2.2.4 Logical Name Translation Status

logicalNameTranslationStatus OBJECT-TYPE

SYNTAX NtcipRowStatusStatic

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> This object allows for the management of rows within the table.

<Supplanted by> SNMP-TARGET-MIB.SnmpTargetAddrRowStatus (RFC3413)

<Informative> This object was deprecated because the SNMPv3 includes the definition of targets in the SNMP-TARGET-MIB, as defined in RFC 3413.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.2.1.4"

```
DEFVAL      { invalid }
```

```
::= { logicalNameTranslationEntry 4 }
```

```
END
```

## Section 11 DEPRECATED REPORT MIB

The text provided from Section 11.1 through the end of Section 11 (except the headings) constitutes the standard NTCIP1201-GlobalReport MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 11.1 HEADER

```
NTCIP1201-GlobalReport DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, Integer32, Unsigned32,
    Opaque, zeroDotZero
    FROM SNMPv2-SMI
    -- RFC 2578
    VariablePointer
    FROM SNMPv2-TC
    -- RFC 2579
    global
    FROM NTCIP1201-Global;

globalReport MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
    "name: NTCIP Coordinator
    email: ntcip@nema.org
    postal: National Electrical Manufacturers Association
    1300 North 17th Street, Suite 1752
    Rosslyn, Virginia 22209-3801"
DESCRIPTION
    "<Definition> This MIB defines the SMIV2 representation of the
    NTCIP1103v0352-globalReport MIB, which was defined in NTCIP 1103. This MIB
    defines objects related to managing event information for the Purpose of
    logging data within the device. This MIB was deprecated in NTCIP 1201 v04
    due to security issues in the structure of the MIB. The objects have been
    replaced by objects in ISO 26048-1.
    *** All objects in this MIB have been deprecated. ***

    <Informative> The event class table is presented first to ease the
    readability of the standard; however, the node numbers assigned to this
    table reflect the original node numbering used in the original 1996
    specification to preserve backwards compatibility with existing systems.

    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4"
REVISION "20230701145436Z"
DESCRIPTION
    "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
    "NTCIP 1103 v03 - No change."
REVISION "200903310000Z"
DESCRIPTION
    "NTCIP 1103 v02 - Separated into its own MIB."
```

REVISION "200409270000Z"  
DESCRIPTION  
"NTCIP 1103 v01: Original version."  
::= { global 4 }

## 11.2 OBJECTS

### 11.2.1 Event Classes

maxEventClasses OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
UNITS "classes"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
"<Definition> The object defines the number of rows in the eventClassTable that this device supports. This is a static table.  
  
<Informative> The eventClassTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.5"  
::= { globalReport 5 }

### 11.2.2 Event Class Table

eventClassTable OBJECT-TYPE  
SYNTAX SEQUENCE OF EventClassEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
"<Definition>This table is used to configure event logging limits and log table maintenance.  
  
<Table Type> static  
  
<Supplanted by> ISO26048-1-Log.fdLogManagerTable (ISO 26048-1)  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6"  
::= { globalReport 6 }

eventClassEntry OBJECT-TYPE  
SYNTAX EventClassEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
"<Definition>This defines a row in the Event Class Table  
  
<Supplanted by> ISO26048-1-Log.fdLogManagerEntry (ISO 26048-1)  
  
<Informative> The replacement table adds an index (fdLogManagerOwner) to allow administrators to limit access rights to groups of rows within the table and to allow users to define new rows without worrying if they are overwriting rows managed by other users. The replacement table also adds the following columns: a size limit (in octets) in addition to a limit on the number of entries, a separation of the time into a date and time to parallel the new way to represent time, parameters to configure the type of memory to use to store the log and the log configuration, a counter of the number of events bumped, and a row status. It is missing the number of rows currently in the log.

```
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6.1"
INDEX      { eventClassNumber }
::= { eventClassTable 1 }

EventClassEntry ::= SEQUENCE {
    eventClassNumber      Integer32,
    eventClassLimit       Integer32,
    eventClassClearTime   Unsigned32,
    eventClassDescription OCTET STRING,
    eventClassNumRowsInLog Integer32,
    eventClassNumEvents   Integer32}

```

### 11.2.2.1 Event Class Number Parameter

```
eventClassNumber OBJECT-TYPE
SYNTAX      Integer32 (1..255)
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
    "<Definition>This is a class value that is to be configured.
    <Supplanted by> ISO26048-1-Log.fdLogManagerName (ISO 26048-1)
    <Informative> The replacement object is a secondary index that uses an
    SnmpAdminString syntax; the primary index is ISO26048-1-
    Log.fdLogManagerOwner which also uses an SnmpAdminSyntax.
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6.1.1"
::= { eventClassEntry 1 }

```

### 11.2.2.2 Event Class Limit Parameter

```
eventClassLimit OBJECT-TYPE
SYNTAX      Integer32 (0..255)
UNITS       "events"
MAX-ACCESS read-write
STATUS      deprecated
DESCRIPTION
    "<Definition>This object specifies the maximum number of events of the
    associated class to store in the log. Once the limit is reached, the oldest
    entry of the matching class will be overwritten by any new entry of the
    same class. If the value of this object is set to a number smaller than the
    current number of rows within this class in the eventLogTable, then the
    oldest entries shall be lost/deleted. The sum of all event class limits
    shall not exceed the maxEventLogSize object; if a SET operation to this
    object causes the sum of eventClassLimit objects to exceed maxEventLogSize,
    then the agent shall respond with a genErr.
    The event cannot be logged if the eventClass has an eventClassLimit of zero
    (0). <Unit>Event
    <Supplanted by> ISO26048-1-Log.fdLogManagerSizeLimit & LOG-
    MIB.fdLogManagerEntryLimit (ISO 26048-1)
    <Informative> The replacement objects allow limits to be placed on the
    number of entries or the total size. The replacement object for the number
    of entries has a syntax of Unsigned32.
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6.1.2"
::= { eventClassEntry 2 }

```



### 11.2.2.3 Event Class Clear Time Parameter

eventClassClearTime OBJECT-TYPE  
SYNTAX Unsigned32  
UNITS "seconds"  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object is used to clear multiple event log entries from the eventLogTable. All events of this class that have an eventLogTime equal to or less than this object shall be cleared from the eventLogTable. If this object has a value greater than the current value of globalTime, it shall prevent the logging of any events of this class.  
<Supplanted by> ISO26048-1-Log.fidLogManagerClearDate & LOG-MIB.fidLogManagerClearTime (ISO 26048-1)  
<Informative> This SMIV2 representation of the original SMIV1 object uses an Unsigned32 syntax rather than the original Counter syntax because the definition of the object does not meet the semantics defined for a Counter object. As a result, this object will have a different value in the 'type' field in the BER encoding when transmitted using SNMPv3 (and using SMIV2) than when transmitted using SNMPv1 (and using the SMIV1 definition). The superseding objects use the extended time format that includes a date object (good to the year 65535) and a time object (with millisecond resolution).  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6.1.3"  
DEFVAL { 0 }  
 ::= { eventClassEntry 3 }

### 11.2.2.4 Event Class Description Parameter

eventClassDescription OBJECT-TYPE  
SYNTAX OCTET STRING  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object specifies a description of the class in ASCII characters.  
<Supplanted by> ISO26048-1-Log.fidLogManagerDescription (ISO 26048-1)  
<Informative> The replacement object specifies an SnmpAdminString Syntax, which allows management systems to automatically recognize the entry as text in any language.  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6.1.4"  
 ::= { eventClassEntry 4 }

### 11.2.2.5 Event Class Number of Rows in Event Log Table Parameter

eventClassNumRowsInLog OBJECT-TYPE  
SYNTAX Integer32 (0..255)  
UNITS "events"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition> The number of rows for this class that currently exist in the eventLogTable.

<Informative> The replacement table does not include a parallel object for this object.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6.1.5"

::= { eventClassEntry 5 }

#### 11.2.2.6 Class Event Log Counter Parameter

eventClassNumEvents OBJECT-TYPE

SYNTAX Integer32 (0..65535)

UNITS "events"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> This object is a counter that gets incremented every time an event occurs for this class; it shall initialize to zero at power up. The value shall roll over each time it exceeds the maximum of 65535.

<Supplanted by> ISO26048-1-Log.fLogManagerEventsLogged (ISO 26048-1)

<Informative> The replacement object has a syntax of Counter32.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6.1.6"

::= { eventClassEntry 6 }

#### 11.2.3 Maximum Event Log Configurations Parameter

maxEventLogConfigs OBJECT-TYPE

SYNTAX Integer32 (1..65535)

UNITS "event types"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The number of rows that exist in the static eventLogConfig table for this device.

<Informative> The eventLogConfigTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.1"

::= { globalReport 1 }

#### 11.2.4 Event Log Configuration Table

eventLogConfigTable OBJECT-TYPE

SYNTAX SEQUENCE OF EventLogConfigEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition>A table containing Event Log Configuration information. The number of rows in this table is equal to the maxEventLogConfigs object. This table defines the parameters that the device will monitor to create an event.

<Table Type> static

<Supplanted by> ISO26048-1-Log.fLogFactoryTable (ISO 26048-1) & ISO26048-1-CondTrigger.fLogCondTriggerTable (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2"

::= { globalReport 2 }

```
eventLogConfigEntry OBJECT-TYPE
SYNTAX      EventLogConfigEntry
MAX-ACCESS not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition>This object defines an entry in the event log configuration
  table.

  <Supplanted by> ISO26048-1-Log.fLogFactoryEntry (ISO 26048-1) & ISO26048-
  1-CondTrigger.fCondTriggerEntry (ISO 26048-1)

  <Informative> The replacement tables have dual indicies consisting of an
  owner and a name. The trigger table manages the definition of the trigger
  with the following changes:
  - A textual description is added
  - Samples (compare value) can be either the current value (as with this
  node, or a delta (i.e., how fast a value is changing)
  - The concept of Opaque is not supported in SNMPv3, so the replacement
  table includes a 'ValueOctet' object that is used with the bitwise operator
  - A wildcard column that allows defining the same condition on multiple
  comparison OIDs (e.g., all rows of a table)
  - Columns to define the target and context of the comparison object; in
  other words, the comparison can be performed by a proxy agent or can
  reference another device to get the object value to compare against
  - A frequency object that allows the configuration to control how
  frequently a comparison is made
  - A truthDuration object that allows the configuration to require the
  evaluation to be true for some length of time prior to firing the trigger.
  - Startup objects that define whether the triggers startup in a fired or
  unfired state (for hysteresis, there are two startups)
  - A pointer to the action table that identifies the action(s) to be
  performed.
  - An error message object that allows a device to report configuration
  errors.
  - Counters for the number of times the trigger has fired, had Evaluation
  errors, and activation errors.
  - An indication of the type of storage to use
  - A RowStatus object.
  The log factory table manages the definition of the information to be
  recorded with the following changes:
  - An object context, which allows a proxy/hrbrid agent to capture
  information from another context
  - A StorageType object
  - A RowStatus object

  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1"

INDEX      { eventConfigID }
 ::= { eventLogConfigTable 1 }

EventLogConfigEntry ::= SEQUENCE {
    eventConfigID      Integer32,
    eventConfigClass   Integer32,
    eventConfigMode    INTEGER,
    eventConfigCompareValue Integer32,
    eventConfigCompareValue2 Integer32,
    eventConfigCompareOID VariablePointer,
    eventConfigLogOID  VariablePointer,
    eventConfigAction  INTEGER,
```

eventConfigStatus INTEGER}

#### 11.2.4.1 Event Log Configuration ID Parameter

eventConfigID OBJECT-TYPE  
SYNTAX Integer32 (1..65535)  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object contains the row number which is used to identify the event associated with this row in the eventLogConfigTable. The number of event IDs shall not exceed the value indicated in the maxEventLogConfigs object.  
  
<Supplanted by> ISO26048-1-Log.fidLogFactory  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.1"  
  
 ::= { eventLogConfigEntry 1 }

#### 11.2.4.2 Event Log Configuration Class Parameter

eventConfigClass OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object contains the class value to assign to the event associated with this row in the event configuration table. This value is used in the event log table to organize various events defined in this table into logical groupings. This value shall not exceed the maxEventClasses object value.  
  
<Supplanted by> ISO26048-1-Log.fidLogEventFactoryLogName (ISO 26048-1)  
  
<Informative> The event cannot be logged if the EventClass has an eventClassLimit of zero (0).  
The replacement object specifies an SnmpAdminString  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.2"  
  
DEFVAL { 1 }  
 ::= { eventLogConfigEntry 2 }

#### 11.2.4.3 Event Log Configuration Mode Parameter

eventConfigMode OBJECT-TYPE  
SYNTAX INTEGER { other (1),  
 onChange (2),  
 greaterThanValue (3),  
 smallerThanValue (4),  
 hysteresisBound (5),  
 periodic (6),  
 andedWithValue (7) }  
  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object specifies the mode of operation for this event.  
  
<Format> The modes are defined as follows:

Value	Description
-------	-------------

other	the event mode of operation is not described in this standard, refer to the device manual.
onChange	create a log entry when the object value referenced by eventConfigCompareOID changes. The values of eventConfigCompareValue and eventConfigCompareValue2 are ignored in this mode.
greaterThanValue	create a log entry when the object value referenced by eventConfigCompareOID becomes greater than the value of eventConfigCompareValue for the time (tenth seconds) defined by eventConfigCompareValue2 (zero means immediate logging).
smallerThanValue	create a log entry when the object value referenced by eventConfigCompareOID becomes less than the value of eventConfigCompareValue for the time (tenth seconds) defined by eventConfigCompareValue2 (zero means immediate logging).
hysteresisBound	create a log entry when the object value referenced by eventConfigCompareOID becomes less than or greater than the bound values. The lowerbound value is the lower value of eventConfigCompareValue and eventConfigCompareValue2; the upperbound value is the higher value of the two values.

When the object value becomes greater than the upper bound value, subsequent logging of upperbound conditions shall not occur until the object value becomes less than the lower bound value.

When the object value becomes less than the lower bound value, subsequent logging of lowerbound conditions shall not occur until the object value becomes greater than the upper bound value.

Periodic create a log entry every x seconds, where x is defined by the value stored in eventConfigCompareValue. The values stored in eventConfigCompareValue2 and eventConfigCompareOID are ignored in this mode.

andedWithValue create a log entry when the object value referenced by eventConfigCompareOID ANDED with the value of eventConfigCompareValue is NOT equal to zero for the time (tenth seconds) defined by eventConfigCompareValue2 (zero means immediate logging). This allows monitoring of a specific bit; the condition becomes true anytime that any one of the selected bits become true.

<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerMode (ISO 26048-1)

<Informative> The replacement object adds the following modes: - equal - not equal - on creation (i.e., creation of a new row in a table) - on deletion - separate modes for octet string and integer bitwise operations

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.3"

DEFVAL { onChange }  
::= { eventLogConfigEntry 3 }

#### 11.2.4.4 Event Log Configuration Compare Value Parameter

eventConfigCompareValue OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object contains the comparison value to use with eventConfigMode values (greaterThanValue, smallerThanValue, hysteresisBound). No value within this object is necessary when the eventConfigMode-object has the value onChange (2).

<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerValue & fdCondTriggerValueOctet (ISO 26048-1)

<Informative> The interger-based replacement object conforms to SNMPv3 rules and does not allow the specification of a 64-bit value and requires

integers to be defined as either signed or unsigned.  
eventConfigCompareValue is perhaps ambiguous about this, although SNMPv1 conventions dictate the same functionality. The result is that the current design does not allow for comparing large unsigned integer values.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.4"

DEFVAL { 0 }  
::= { eventLogConfigEntry 4 }

#### 11.2.4.5 Event Log Configuration Compare Value 2 Parameter

eventConfigCompareValue2 OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>If the eventConfigMode is set to hysteresisBound, this object specifies the second comparison value for the hysteresis. If the eventConfigMode is set to greaterThanValue, smallerThanValue, or andedWithValue, this object specifies the time (in tenths of seconds, +1 tenth / -0 tenths) for which the samples used for comparison shall be true prior to the event condition becoming true. If the eventConfigMode is set to onChange or periodic, the value of this object shall be ignored. The amount of time the condition shall be true is measured in tenths of a second. The accuracy of this timer is limited to +1 tenth of a second and:0 tenths of a second. If the event is true for at least the time shown in this parameter +1 tenth of a second, the condition shall trigger a log entry. It is recognized that some designs only sample the condition periodically, in which case the condition shall be true for at least the time indicated by this object before the event becomes true and the event shall always become true if the condition is true for a duration equal to the value shown in this object plus 1 tenth of a second.

<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerValue2 (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.5"

DEFVAL { 0 }  
::= { eventLogConfigEntry 5 }

#### 11.2.4.6 Event Log Configuration Compare Object Identifier Parameter

eventConfigCompareOID OBJECT-TYPE

SYNTAX VariablePointer

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> This object contains the object identifier which references the value against which the comparison is made. If the eventConfigMode is set to periodic, the value of this object shall be ignored. If the eventConfigMode is set to greaterThanValue, smallerThanValue or hysteresisBound, this object shall reference an object whose SYNTAX resolves to a ranged or unranged INTEGER. As with all other objects that are sub-ranged by a given implementation, an agent should return a badValue error if it receives a set command indicating a OID which is not supported by the implementation or which is not zeroDotZero.

<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerObject (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.6"

DEFVAL { zeroDotZero }

```
::= { eventLogConfigEntry 6 }
```

#### 11.2.4.7 Event Log Configuration Log Object Identifier Parameter

```
eventConfigLogOID OBJECT-TYPE  
SYNTAX      VariablePointer  
MAX-ACCESS  read-write  
STATUS      deprecated  
DESCRIPTION
```

"<Definition>This object contains the object identifier which indicates what value to log when a condition or event occurs (e.g., log the phase display when the watchdog alarm status changes). As with all other objects that are sub-ranged by a given implementation, an agent should return a badValue error if it receives a set command indicating a value which is not supported by the implementation. The valid value range of this object shall not include any values, other than zeroDotZero, that do not correspond to objects that may exist within the agent, although it may be further restricted.

The valid value range of this object shall not include objects under the following nodes: Security - { nema transportation devices global security }  
CHAP - { nema transportation protocols layers chap }

<Supplanted by> ISO26048-1-Log.fidLogEventFactoryObjectID (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.7"

```
DEFVAL      { zeroDotZero }  
::= { eventLogConfigEntry 7 }
```

#### 11.2.4.8 Event Log Configuration Action Parameter

```
eventConfigAction OBJECT-TYPE  
SYNTAX      INTEGER { other (1),  
                    disabled (2),  
                    log (3) }  
MAX-ACCESS  read-write  
STATUS      deprecated  
DESCRIPTION
```

"<Definition>The value of this object indicates what action shall take place when this event occurs.

<Format>

Other - indicates that the action is other than defined in this standard. This value exists in order to support proprietary event logging mechanisms configured by other means not specified in this standard. If this value is used in a SET request, the agent shall respond with a badValueError.  
Disabled - no event log entry shall be generated or recorded due to this event. In an agent complying with NTCIP 1103 v03 or later, this event shall not be used to trigger NTCIP traps, nor to construct NTCIP trap messages.  
Log - an event log entry shall be generated when this event occurs. In an agent complying with NTCIP 1103 v03 and later, this may trigger an NTCIP trap (see the eventConfigID index element of trapTable). If eventConfigClass refers to an eventClassTable row having eventClassLimit = 0, the log entry's eventLogValue shall be used to construct any necessary trap messages implied by the associated trapTable rows, but the log entry shall then be discarded and not added to the eventLogTable. If the eventClassLimit is greater than zero, the log entry shall be added to the eventLogTable, subject to the constraints imposed by the associated eventConfigClass.



<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerActionOwner, fdCondTriggerAction, fdCondTriggerActionOwner2, & fdCondTriggerAction2 (ISO 26048-1)

<Informative> The replacement object points to a table that can point to multiple actions. For example, a single trigger can result in both a log entry and a notification.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.8"

DEFVAL { disabled }  
 ::= { eventLogConfigEntry 8 }

#### 11.2.4.9 Event Log Configuration Status Parameter

eventConfigStatus OBJECT-TYPE  
SYNTAX INTEGER { other (1),  
 disabled (2),  
 log (3),  
 error (4) }

MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION

"<Definition>The value of this object indicates the current status of the configured event. Upon setting any object in this row of the eventLogConfigTable, the agent will determine if the setting is valid and will set this object to one of the following states:  
other indicates that the action is successfully set to a mode other than that defined in this standard  
disabled indicates that the action is set to disabled  
log indicates that the action is successfully set to the log state after passing consistency checks.  
error indicates that the requested action could not be implemented due to a consistency check

<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerRowStatus (ISO 26048-1) & ISO26048-1-Log.fdLogEventFactoryRowStatus (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.9"

::= { eventLogConfigEntry 9 }

#### 11.2.5 Maximum Event Log Size Parameter

maxEventLogSize OBJECT-TYPE  
SYNTAX Integer32 (1..65535)  
UNITS "events"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION

"<Definition>The maximum, fixed number of rows that can be utilized within the eventLogTable.

<Supplanted by> ISO26048-1-Log.fdLogsGlobalEntryLimit and fdLogsGlobalSizeLimit (ISO 26048-1)

<Informative> The replacement objects are for all managers.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.3"

::= { globalReport 3 }

#### 11.2.6 Event Log Table

eventLogTable OBJECT-TYPE

```
SYNTAX      SEQUENCE OF EventLogEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition>A table containing Event History data collected. A request for
  an object from a row that has not been instantiated or has been cleared
  shall return a noSuchName error.

  <Table Type> dynamic

  <Supplanted by> ISO26048-1-Log.fdLogTable (ISO 26048-1)

  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4"
 ::= { globalReport 4 }

eventLogEntry OBJECT-TYPE
SYNTAX      EventLogEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition>This object defines an entry in the event log Table.

  <Supplanted by> ISO26048-1-Log.fdLogEntry (ISO 26048-1)

  <Informative> EventLogTable was modified in NTCIP 1103 v03 to add an entry
  eventLogTimeMilliseconds Integer, which did not exist in NTCIP 1103 v02.
  The replacement table precedes the class/manager and number indicies With
  an owner so that access to information in the table can be controlled with
  proper configuration of the SNMP agent. The replacement table also records
  both the date and time of the event and the date and time that the event
  was placed into the table (i.e., so that if there is latency, a command to
  clear the table does not inadvertently delete rows that had occurred but
  had not been entered into the table.

  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1"
INDEX       { eventLogClass, eventLogNumber }
 ::= { eventLogTable 1 }

EventLogEntry ::= SEQUENCE {
    eventLogClass      Integer32,
    eventLogNumber     Integer32,
    eventLogID         Integer32,
    eventLogTime       Unsigned32,
    eventLogValue      Opaque,
    eventLogTimeMilliseconds Integer32}

```

#### 11.2.6.1 Event Log Class Parameter

```
eventLogClass OBJECT-TYPE
SYNTAX      Integer32 (1..255)
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition>This object contains the class of the associated event as
  defined in the eventLogConfig Table.

  <Supplanted by> ISO26048-1-Log.fdLogManagerOwner & fdLogManagerName (ISO
  26048-1)

  <Informative> The replacement objects are SnmpAdminStrings

```

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1.1"

::= { eventLogEntry 1 }

#### 11.2.6.2 Event Log Number Parameter

eventLogNumber OBJECT-TYPE

SYNTAX Integer32 (1..255)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The event number within this class for this event. Event numbers shall be assigned starting at 1 and shall increase to the value specified by the associated eventClassLimit for the class associated with the rows. Events shall maintain a chronological ordering in the table with the oldest event of a class occupying the row with eventNumber = 1, and subsequent events filling subsequent rows. This ordering shall be maintained for those rows still remaining when events are cleared.

<Supplanted by> ISO26048-1-Log.fdLogIndex (ISO 26048-1)

<Informative> The replacement object is an Unsigned32

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1.2"

::= { eventLogEntry 2 }

#### 11.2.6.3 Event Log ID Parameter

eventLogID OBJECT-TYPE

SYNTAX Integer32 (1..65535)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>This object contains the event configuration ID (from the eventLogConfigTable) that caused this table entry. It indicates the row in the eventLogConfig table responsible for this event entry.

<Supplanted by> ISO26048-1-Log.fdLogFactoryName (ISO 26048-1)

<Informative> The replacement object is an SnmpAdminString

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1.3"

::= { eventLogEntry 3 }

#### 11.2.6.4 Event Log Time Parameter

eventLogTime OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The time that the event was detected. If the device supports the globalTime object, the value shall reflect the value of globalTime when the event occurred, otherwise this shall be the time in seconds since the device powered up. The event shall be detected and timestamped within one second from the event becoming true. The event shall be logged in the table within five seconds of the event being detected. These timing resolutions may be modified by a device profile.

<Supplanted by> ISO26048-1-Log.fdLogEventDate, fdLogEventTime, fdLogDate, fdLogTime (ISO 26048-1)

<Informative> This SMIV2 representation of the original SMIV1 object uses an Unsigned32 syntax rather than the original Counter syntax because the definition of the object does not meet the semantics defined for a Counter object. As a result, this object will have a different value in the 'type' field in the BER encoding when transmitted using SNMPv3 (and using SMIV2) than when transmitted using SNMPv1 (and using the SMIV1 definition). The superseding objects adopt the new time format. They also distinguish between event time and log time.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1.4"

::= { eventLogEntry 4 }

#### 11.2.6.5 Event Log Value Parameter

eventLogValue OBJECT-TYPE  
SYNTAX Opaque  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION

"<Definition>The value of this object is set to the BER encoding of the value referenced by the eventConfigLogOID of the associated eventLogID when the event was logged. Its length is variable. The value shall not contain any padding characters either before or after the values. NOTE: Opaque objects are doubly wrapped. For SNMP operations, which use BER, this would be {type, length, {type, length, value}}. For example, a zero-length octet string, would be encoded in BER as 0x44 02 04 00. For STMP or SFMP operations, which use OER, this would be { length, {type, length, value}}. For example, the same example would be encoded in OER as 0x02 04 00.

<Supplanted by> ISO26048-1-Log.fdLogValue (ISO 26048-1)

<Informative> The replacement object uses an ITSOerString (i.e., the value is in OER and sent over SNMP with a BER wrapper as opposed to the Opaque which created a double BER wrapper). This means that the manager who requests the value must know the syntax that was encoded.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1.5"

::= { eventLogEntry 5 }

#### 11.2.6.6 Event Log Time Milliseconds Parameter

eventLogTimeMilliseconds OBJECT-TYPE  
SYNTAX Integer32 (0..999)  
UNITS "milliseconds"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION

"<Definition>The number of milliseconds after the beginning of the second indicated by the value of eventLogTime at which the event was detected. Devices that do not support sub-second event time resolution shall always set this object to zero. When implementing eventLogTimeMilliseconds, devices require a time source with millisecond-level resolution, such as GPS or TIA (International Atomic Time).

<Supplanted by> ISO26048-1-Log.fdLogEventTime, fdLogTime

<Informative> This data content is covered by the new time format used by ISO26048-1-Log.fdLogEventTime & fdLogTime (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1.6"

::= { eventLogEntry 6 }

### 11.2.7 Total Event Log Counter Parameter

```
numEvents OBJECT-TYPE
SYNTAX      Integer32 (0..65535)
UNITS       "events"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> This object is a counter that gets incremented every time an
  event occurs and shall initialize to zero at power up. The value shall roll
  over each time it exceeds the maximum of 65535.

  <Supplanted by> ISO26048-1-Log.fdLogsTotalLogged (ISO 26048-1)

  <Informative> The replacement object uses a syntax of Counter32, which does
  not require a zero-base.

  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.7"

 ::= { globalReport 7 }
```

### 11.2.8 Event Log Time Latency Parameter

```
eventTimeLatency OBJECT-TYPE
SYNTAX      Integer32 (0..1000)
UNITS       "milliseconds"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> This object indicates the maximum amount of time, in
  milliseconds, that may elapse between an event's occurrence and the time
  reported for that event entry in the eventLogTable. This is a global,
  constant value that reports the capability of the device with respect to
  event-reporting latency. It should account for all sources of latency,
  including both hardware and firmware delays. If eventTimeLatency has a
  value of L, this means that any event in the eventLogTable may actually
  have occurred up to L milliseconds prior to the time reported by the
  eventLogTime and eventLogTimeMilliseconds values associated with the event.
  A value of 0 indicates that the device reports accurate event times with
  millisecond resolution. A value of 1000 indicates that the device cannot
  accurately report sub-second event times.

  <Supplanted by> ISO26048-1-Log.fdLogsRecordingLatency (ISO 26048-1)

  <Informative> The replacement object places limits on the time between
  retrieving the value to log and the logging rather than what is described
  here.

  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.8"

 ::= { globalReport 8 }
END
```

## Section 12 DEPRECATED SECURITY MIB

The text provided from Section 12.1 through the end of Section 12 (except the headings) constitutes the standard NTCIP1201-Security MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 12.1 HEADER

```
NTCIP1201-Security DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Integer32, Gauge32
FROM SNMPv2-SMI
-- RFC 2578

global
FROM NTCIP1201-Global;

security MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
  "name: NTCIP Coordinator
  email: ntcip@nema.org
  postal: National Electrical Manufacturers Association
  1300 North 17th Street, Suite 1752
  Rosslyn, Virginia 22209-3801"
DESCRIPTION
  "<Definition> This MIB defines the SMIV2 representation of the
  NTCIP1103v0352-Security MIB, which was defined in NTCIP 1103. This MIB
  defines objects related to managing the configuration of community names
  and associated access rights for SNMPv1 packets.
  This MIB was deprecated in NTCIP 1201 v04 due to security limitations of
  SNMPv1. The objects contained in this MIB are no longer relevant as
  Security is now based on (D)TLS certificates combined with SNMP
  securityNames as defined in the Transport Security Model (RFC 5591) and the
  TLS Transport Model (RFC 6353). However, this MIB defines how to exchange
  this data to enable a manager to remotely manage these objects through a
  proxy agent.
  *** All objects in this MIB have been deprecated. ***"
  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5"
REVISION "20230701145436Z"
DESCRIPTION
  "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
  "NTCIP 1103 v03 - No change."
REVISION "200903310000Z"
DESCRIPTION
  "NTCIP 1103 v02 - Separated into its own MIB."
REVISION "200409270000Z"
DESCRIPTION
  "NTCIP 1103 v01: Original version."
```

```
::= { global 5 }
```

## 12.2 OBJECTS

### 12.2.1 Community Name Administrator Parameter

```
communityNameAdmin OBJECT-TYPE  
SYNTAX OCTET STRING (SIZE(8..16))  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION
```

"<Definition> This object is the community name that shall be used to specifically gain access to information under the security node. A message with this value in the community name field of an SNMP message has user read-write access to the security node objects and all other objects implemented in the device. The syntax is defined as an OCTET STRING and therefore any character can have a value of 0..255.

<Informative> This object has been deprecated along with the use of SNMPv1.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.1"

```
DEFVAL { "administrator" }  
::= { security 1 }
```

### 12.2.2 Maximum Community Names Parameter

```
communityNamesMax OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION
```

"<Definition> This object specifies the maximum number of rows that are implemented in the community name table.

<Informative> This object has been deprecated along with the use of SNMPv1.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.2"

```
::= { security 2 }
```

### 12.2.3 Community Names Table

```
communityNameTable OBJECT-TYPE  
SYNTAX SEQUENCE OF CommunityNameTableEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION
```

"<Definition> This table defines the community names that can appear in the community name field of the SNMP message and access privileges associated with that community name.

<Table Type> static

<Informative> This object has been deprecated along with the use of SNMPv1.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.3"

```
::= { security 3 }
```

```
communityNameTableEntry OBJECT-TYPE  
SYNTAX CommunityNameTableEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION
```

```
"<Definition> This is the row index of information in the community name
table.
<Informative> This object has been deprecated along with the use of SNMPv1.
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.3.1"
INDEX      { communityNameIndex }
::= { communityNameTable 1 }

CommunityNameTableEntry ::= SEQUENCE {
    communityNameIndex      Integer32,
    communityNameUser       OCTET STRING,
    communityNameAccessMask Gauge32}
```

### 12.2.3.1 Community Name Index Parameter

```
communityNameIndex OBJECT-TYPE
SYNTAX      Integer32 (1..255)
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
"<Definition> This object defines the row index into the
communityNameTable. This value shall not exceed the communityNamesMax
object value.
<Informative> This object has been deprecated along with the use of SNMPv1.
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.3.1.1"
::= { communityNameTableEntry 1 }
```

### 12.2.3.2 User Community Name Parameter

```
communityNameUser OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE(6..16))
MAX-ACCESS read-write
STATUS      deprecated
DESCRIPTION
"<Definition> This object defines a community name value that a security
administrator can assign user read-write access to information (other than
security) in a device. A message with this value in the community name
field of an SNMP/SFMP message has user access rights as defined in the
communityNameAccessMask. The syntax is defined as an OCTET STRING and
therefore any character can have a value of 0..255.
<Informative> This object has been deprecated along with the use of SNMPv1.
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.3.1.2"
DEFVAL      { "public" }
::= { communityNameTableEntry 2 }
```

### 12.2.3.3 User Community Name Mask Parameter

```
communityNameAccessMask OBJECT-TYPE
SYNTAX      Gauge32
MAX-ACCESS read-write
STATUS      deprecated
DESCRIPTION
"<Definition> This object defines a 32 bit mask that can be used to
associate 'write access' with a community name. A value of 0x00 00 00 00
grants the community name user read-only access and overrides any
individual object's read-write access clause. A value of 0xFF FF FF FF
```



grants the community name user read-write access and an individual object's read-write access clause applies. Values other than 0x00 00 00 00 and 0xFF FF FF FF are implementation specific and may limit viewing and/or accessing the information in a device.

<Informative> This object has been deprecated along with the use of SNMPv1.

<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.3.1.3"

```
DEFVAL      { 4294967295 }  
 ::= { communityNameTableEntry 3 }  
END
```

## Section 13 DEPRECATED TRAP MIB

The text provided from Section 13.1 through the end of Section 13 (except the headings) constitutes the standard NTCIP1201-NtcipTraps MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 13.1 HEADER

```
NTCIP1201-NtcipTraps DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
Integer32, Counter32, zeroDotZero
                                FROM SNMPv2-SMI
                                -- RFC 2578

ITSOerString
                                FROM ISO26048-1-FieldDevice
application, protocols, NtcipRowStatusStatic
                                FROM NTCIP8004-Transportation
eventConfigID
                                FROM NTCIP1201-GlobalReport;

ntcipTraps MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
  "name: NTCIP Coordinator
   email: ntcip@nema.org
   postal: National Electrical Manufacturers Association
           1300 North 17th Street, Suite 1752
           Rosslyn, Virginia 22209-3801"
DESCRIPTION
  "<Definition> This MIB defines the SMIV2 representation of the
  NTCIP1103v0352-Traps MIB, which was defined in NTCIP 1103. This MIB defines
  objects related to:
  (a) configuration of block and watch objects,
  (b) configuration and monitoring of traps
  This MIB was deprecated in NTCIP 1201 v04 due to security issues in the
  structure of the MIB. The objects have been replaced by objects in ISO
  26048-1.
  *** All objects in this MIB have been deprecated. ***

  <Informative> In addition to the objects mentioned below, the replacement
  feature includes objects to indicate: a) the encodings supported (BER
  and/or OER) b) whether there are limitations on the support for structures
  that allow new values (i.e., set operations) c) whether the supports one-
  step processing, two-step processing, or both

  The watchBlock and reportBlock features are replaced by the same fdDynObj
  feature defined in ISO 26048-1.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.4"
REVISION "20230701145436Z"
DESCRIPTION
  "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
```

REVISION "201612310000Z"  
DESCRIPTION  
"NTCIP 1103 v03 - No change."

REVISION "200903310000Z"  
DESCRIPTION  
"NTCIP 1103 v02 - Separated into its own MIB."

REVISION "200409270000Z"  
DESCRIPTION  
"NTCIP 1103 v01: Original version."

::= { protocols 4 }

## 13.2 OBJECT IDENTITIES

### 13.2.1 Watch Blocks

watchBlocks OBJECT-IDENTITY  
STATUS deprecated  
DESCRIPTION

"<Definition> Watch Blocks are OER encoded configurable read only blocks intended to be utilized for device status monitoring in the eventConfigCompareOID in the eventConfigTable. The intent is to be able to configure events to monitor a collection of NTCIP objects at the same time, and trigger the logging and/or transmission of a trap message.

<Informative> When a watch block is used for the eventConfigCompareOID, the eventConfigMode object is restricted to onChange (2) Any entry with an attempt to use any other mode shall be ignored at run time. Because there is no restriction on the order in which the entries are created, specifying a watch block that has not been configured does not generate an error. Likewise care should be taken to ensure that the configuration of the event table and the watch blocks (as well as the report blocks) are consistent and correct.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6"

::= { application 6 }

### 13.2.2 Report Blocks

reportBlocks OBJECT-IDENTITY  
STATUS deprecated  
DESCRIPTION

"<Definition> Report blocks are OER encoded configurable read only blocks intended to be utilized for device status and other parameters as the eventConfigLogOID in the eventConfigTable. Like the watch blocks, they can only be validated at run-time. Improperly configured report blocks shall be ignored.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7"

::= { application 7 }

### 13.2.3 Clear Objects

eventClearObjects OBJECT-IDENTITY  
STATUS deprecated  
DESCRIPTION

"<Definition> This node is an identifier used to group all objects for support of clearing the report node (events) and report objects.

```
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8"  
 ::= { application 8 }
```

### 13.2.4 NTCIP Trap Management

```
trapMgmt OBJECT-IDENTITY  
STATUS deprecated  
DESCRIPTION  
 "<Definition> This node defines information used to manage the generation  
 and issuance of traps.  
 <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1"  
 ::= { ntcipTraps 1 }
```

### 13.2.5 NTCIP Trap Data

```
ntcipTrapData OBJECT-IDENTITY  
STATUS deprecated  
DESCRIPTION  
 "<Definition> This node defines information to be reported by the trap  
 management feature.  
 <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.2"  
 ::= { ntcipTraps 2 }
```

### 13.2.6 NTCIP Trap Notifications

```
ntcipTrapNotifications OBJECT-IDENTITY  
STATUS deprecated  
DESCRIPTION  
 "<Definition> This node defines trap information to be reported by the trap  
 management feature.  
 <Informative> SMIV2 defines notifications, which can be sent as  
 unacknowledged 'traps' or acknowledged 'informs'. The SMIV2 NOTIFICATION-  
 TYPE macro registers traps on the naming tree and per RFC 4181, to provide  
 backward compatibility the OID of a notification should be the SNMPv1  
 'enterprise' followed by a node '0' followed by the number assigned to the  
 SMIV1 trap. This node provides the prefix to define such notifications.  
 <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.2.0"  
 ::= { ntcipTrapData 0 }
```

## 13.3 WATCH BLOCKS

### 13.3.1 Maximum Watch Objects

```
maxWatchObjects OBJECT-TYPE  
SYNTAX Integer32 (150..8192)  
UNITS "watch objects"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
 "<Definition>The number of rows that exist in the  
 watchObjectDefinitionTable for this device.  
 <Supplanted by> ISO26048-1-DynObj.fDynObjsMaxObjects (ISO 26048-1)
```

<Informative> The watchObjectDefinitionTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows; however, the replacement object allows implementations to impose a limit on the size of block objects and to inform the user of the limit.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.1"

::= { watchBlocks 1 }

### 13.3.2 Maximum Watch Blocks

maxWatchBlocks OBJECT-TYPE

SYNTAX Integer32 (1..50)

UNITS "watch blocks"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The number of rows that exist in the watchBlockTable for this device.

<Informative> The watchBlockTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.2"

::= { watchBlocks 2 }

### 13.3.3 Watch Object Definition Table

watchObjectDefinitionTable OBJECT-TYPE

SYNTAX SEQUENCE OF WatchObjectDefinitionEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition>A table containing Watch Object definition information. The number of rows in this table is equal to the maxWatchObjects object.

<Table Type> static

<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldTable (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.3"

::= { watchBlocks 3 }

watchObjectDefinitionEntry OBJECT-TYPE

SYNTAX WatchObjectDefinitionEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition>This object defines an entry in the Watch Object Definition table.

<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldEntry (ISO 26048-1)

<Informative> The replacement table has a three-part index consisting of a group owner (which provides a level of protection from other users from changing the definition without authorization), a group name (which is functionally equivalent to watchBlock), and field index (which is functionally equivalent to watchID).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.3.1"

INDEX { watchID }

```
::= { watchObjectDefinitionTable 1 }
```

```
WatchObjectDefinitionEntry ::= SEQUENCE {  
    watchID      Integer32,  
    watchStatus  NtcipRowStatusStatic,  
    watchBlock   Integer32,  
    watchOID     OBJECT IDENTIFIER}
```

### 13.3.3.1 Watch Identification Parameter

```
watchID OBJECT-TYPE  
SYNTAX      Integer32 (1..255)  
MAX-ACCESS  read-only  
STATUS      deprecated  
DESCRIPTION  
    "<Definition>This object contains the row number which is used to identify  
    the object associated with this row in the watchObjectDefinitionTable. This  
    value shall not exceed the value indicated by the maxWatchObjects object.  
    <Supplanted by> ISO26048-1-DynObj.fdDynObjFieldIndex  
    <Informative> The replacement object is the third part of a three-part  
    index and is defined as an Unsigned32.  
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.3.1.1"
```

```
::= { watchObjectDefinitionEntry 1 }
```

### 13.3.3.2 Watch Status Parameter

```
watchStatus OBJECT-TYPE  
SYNTAX      NtcipRowStatusStatic  
MAX-ACCESS  read-write  
STATUS      deprecated  
DESCRIPTION  
    "<Definition>The value of this object indicates the current status of the  
    this row in the table.  
    <Informative> The replacement table does not have a RowStatus object; when  
    changes are necessary, the entire object needs to be cleared and redefined.  
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.3.1.2"
```

```
DEFVAL      { invalid }  
::= { watchObjectDefinitionEntry 2 }
```

### 13.3.3.3 Watch Block Parameter

```
watchBlock OBJECT-TYPE  
SYNTAX      Integer32 (1..255)  
MAX-ACCESS  read-write  
STATUS      deprecated  
DESCRIPTION  
    "<Definition>This object contains the block number to assign to the watch  
    object associated with this row in the watch object definition table. This  
    value shall not exceed the value indicated by the maxWatchBlocks object.  
    <Supplanted by> ISO26048-1-DynObj.fdDynObjName (ISO 26048-1)  
    <Informative> The replacement object is the second part of a three- part  
    index and is defined as an SnmpAdminString (SIZE(0..32))  
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.3.1.3"
```

```
DEFVAL      { 1 }
```

```
::= { watchObjectDefinitionEntry 3 }
```

#### 13.3.3.4 Watch Object Identifier Parameter

watchOID OBJECT-TYPE

SYNTAX OBJECT IDENTIFIER

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object contains the object identifier of the object to watch.

The following objects shall NOT be assigned to any watchOID:

- All objects under the security node (Annex B) { nema transportation devices global security }
- All objects under the dynObjMgmt node (Annex A) {nema transportation protocols dynObjMgmt}
- All objects under the chap node (Annex B of NTCIP 2301) { nema transportation protocols layers chap }
- Any objects so identified by various device standards
- Any objects whose SYNTAX does NOT resolve to a ranged or unranged INTEGER.
- Any other report object or watch object
- Any objects that the agent/device does not support.

An agent should return a badValue error if it receives a SET command for any of the above.

<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldObject (ISO 26048-1)

<Informative> The replacement object does not define any restrictions currently.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.3.1.4"

DEFVAL { zeroDotZero }

```
::= { watchObjectDefinitionEntry 4 }
```

#### 13.3.4 Watch Block Table

watchBlockTable OBJECT-TYPE

SYNTAX SEQUENCE OF WatchBlockEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> A table containing the Watch Blocks defined in the Watch Object Definition table. The number of rows in this table is equal to the value of the maxWatchBlocks object.

<Table Type> static

<Supplanted by> ISO26048-1-DynObj.fdDynObjTable (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.4"

```
::= { watchBlocks 4 }
```

watchBlockEntry OBJECT-TYPE

SYNTAX WatchBlockEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition>This defines a row in the watchBlockTable.

<Supplanted by> ISO26048-1-DynObj.fdDynObjEntry (ISO 26048-1)

<Informative> The replacement table has a dual index consisting of an owner and a group name. The table has the following changes:

- the encoding can be selected (BER or OER)
- the ObjectGroup can be configured for retrieval as either a one- step or two-step process. In other words, one-step groups can be retrieved directly; two-step groups have to be refreshed in one command and retrieved in a second command.
- there is a refresh command and a refresh date/time associated with the two-step process. There is also an object that provides an estimated duration for generating the result (which can be as simple as a duration multiplied by the number of objects)
- there is a 'new value' object that allows setting the referenced objects (i.e., similar to a dynObj set command)
- there are 'last error' and 'last error index' objects to provide insights into any issues that arise
- there is a 'clear' object that allows clearing the definition of all fields defined for the object group
- Added an indication of the type of storage to use
- the table is dynamic with a RowStatus object.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.4.1"

```
INDEX      { watchBlockNumber }  
 ::= { watchBlockTable 1 }
```

```
WatchBlockEntry ::= SEQUENCE {  
    watchBlockNumber      Integer32,  
    watchBlockStatus      NtcipRowStatusStatic,  
    watchBlockDescription OCTET STRING,  
    watchBlockValue       ITSOerString}
```

#### 13.3.4.1 Watch Block Number

watchBlockNumber OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
MAX-ACCESS read-only  
STATUS deprecated

##### DESCRIPTION

"<Definition>The block number for this row in the table. This value shall not exceed the value indicated by the maxWatchBlocks object.

<Supplanted by> ISO26048-1-DynObj.fdDynObjName (ISO 26048-1)

<Informative> The replacement object is the second part of a two-part index and is defined as an SnmpAdminString.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.4.1.1"

```
::= { watchBlockEntry 1 }
```

#### 13.3.4.2 Watch Block Status

watchBlockStatus OBJECT-TYPE  
SYNTAX NtcipRowStatusStatic  
MAX-ACCESS read-write  
STATUS deprecated

##### DESCRIPTION

"<Definition>The value of this object indicates the current status of this row in the table.

<Supplanted by> ISO26048-1-DynObj.fdDynObjRowStatus (ISO 26048-1)



<Informative> The replacement object uses RowStatus to support a dynamic table

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.4.1.2"

DEFVAL { invalid }  
::= { watchBlockEntry 2 }

#### 13.3.4.3 Watch Block Description

watchBlockDescription OBJECT-TYPE  
SYNTAX OCTET STRING (SIZE(0..20))  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> This object may be used to define a description of this watch block.

<Supplanted by> ISO26048-1-DynObj.fdDynObjDescription (ISO 26048-1)

<Informative> The replacement object is defined as an SnmpAdminString to support any language with automatic display as text.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.4.1.3"

DEFVAL { "" }  
::= { watchBlockEntry 3 }

#### 13.3.4.4 Watch Block Value

watchBlockValue OBJECT-TYPE  
SYNTAX ITSOerString  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION

"<Definition> An OER encoded string of all object values defined in watchObjectDefinitionTable, pointed at by watchOID (in watchID order) where the watchBlock IS watchBlockNumber AND the watchStatus IS available.

<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue (ISO 26048-1)

<Informative> The original syntax for this object was OerString as defined in NTCIP 8004 v02. NTCIP 8004 v03 recommends the use of ITSOerString in SMIV2 modules as it is formally defined as a textual convention in ISO 26048-1. Both OerString and ITSOerString resolve to OCTET STRING, so while the written syntax has changed, there is no impact on how it is encoded. The replacement object, fdObjectGroupCurrentValue, is defined as an OCTET STRING because it can represent either an OER-string or a BER- string depending on the value of fdObjectGroupEncoding.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.6.4.1.4"

DEFVAL { 'H' }  
::= { watchBlockEntry 4 }

### 13.4 REPORT BLOCKS

#### 13.4.1 Maximum Report Objects

maxReportObjects OBJECT-TYPE  
SYNTAX Integer32 (150..8192)  
UNITS "report objects"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION

"<Definition>The number of rows that exist in the reportObjectDefinitionTable for this device.  
<Supplanted by> ISO26048-1-DynObj.fdDynObjsMaxObjects (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.1"

::= { reportBlocks 1 }

### 13.4.2 Maximum Report Blocks

maxReportBlocks OBJECT-TYPE  
SYNTAX Integer32 (1..50)  
UNITS "report blocks"  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
"<Definition>The number of rows that exist in the reportBlockTable for this device.  
<Informative> The watchBlockTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.2"

::= { reportBlocks 2 }

### 13.4.3 Report Object Configuration Table

reportObjectDefinitionTable OBJECT-TYPE  
SYNTAX SEQUENCE OF ReportObjectDefinitionEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
"<Definition>A table containing Report Object definition information. The number of rows in this table is equal to the maxReportObjects object.  
<Table Type> static  
<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldTable (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3"

::= { reportBlocks 3 }

reportObjectDefinitionEntry OBJECT-TYPE  
SYNTAX ReportObjectDefinitionEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
"<Definition>This object defines an entry in the Report Object Definition table.  
<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldEntry (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3.1"

INDEX { reportID }

::= { reportObjectDefinitionTable 1 }

ReportObjectDefinitionEntry ::= SEQUENCE {  
reportID Integer32,  
reportStatus NtcipRowStatusStatic,  
reportBlock Integer32,  
reportOID OBJECT IDENTIFIER}

#### 13.4.3.1 Report Identification Parameter

reportID OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object contains the row number which is used to identify the objects associated with this row in the reportObjectDefinitionTable. This value shall not exceed the value indicated by the maxReportObjects object.  
  
<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldIndex (ISO 26048-1)  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3.1.1"  
 ::= { reportObjectDefinitionEntry 1 }

#### 13.4.3.2 Report Status Parameter

reportStatus OBJECT-TYPE  
SYNTAX NtcipRowStatusStatic  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> The value of this object indicates the current status of the this row in the table.  
  
<Informative> The replacement table does not have a RowStatus object; when changes are necessary, the entire object needs to be cleared and redefined.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3.1.2"  
 DEFVAL { invalid }  
 ::= { reportObjectDefinitionEntry 2 }

#### 13.4.3.3 Report Block Parameter

reportBlock OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object contains the block number to assign to the log object associated with this row in the reportObjectDefinitionTable. This value shall not exceed the value indicated by the maxReportBlocks object.  
  
<Supplanted by> ISO26048-1-DynObj.fdDynObjName (ISO 26048-1)  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3.1.3"  
 DEFVAL { 1 }  
 ::= { reportObjectDefinitionEntry 3 }

#### 13.4.3.4 Report Object Identifier Parameter

reportOID OBJECT-TYPE  
SYNTAX OBJECT IDENTIFIER  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object contains the object identifier of the object to log. The following objects shall NOT be assigned to any reportOID:  
All objects under the security node (Annex B) { nema transportation devices

global security }  
All objects under the dynObjMgmt node (Annex A) {nema transportation protocols dynObjMgmt}  
All objects under the chap node (Annex B of NTCIP 2301) {nema transportation protocols layers chap }  
Any other report object or watch object  
Any objects so identified by various device standards  
Any objects that the agent/device does not support.

An agent should return a badValue error if it receives a SET command for any of the above.

<Supplanted by> ISO26048-1-DynObj.fdDynObjFieldObject (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3.1.4"

DEFVAL { zeroDotZero }  
::= { reportObjectDefinitionEntry 4 }

#### 13.4.4 Report Block Table

reportBlockTable OBJECT-TYPE  
SYNTAX SEQUENCE OF ReportBlockEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition> A table containing the Report blocks defined in the reportObjectDefinitionTable. The number of rows in this table is equal to the value of the maxReportBlocks object.  
<Table Type> static  
<Supplanted by> ISO26048-1-DynObj.fdDynObjTable (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4"  
::= { reportBlocks 4 }

reportBlockEntry OBJECT-TYPE  
SYNTAX ReportBlockEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition> This defines a row in the reportBlockTable.  
<Supplanted by> ISO26048-1-DynObj.fdDynObjEntry (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1"  
INDEX { reportBlockNumber }  
::= { reportBlockTable 1 }

ReportBlockEntry ::= SEQUENCE {  
reportBlockNumber Integer32,  
reportBlockStatus NtcipRowStatusStatic,  
reportBlockDescription OCTET STRING,  
reportBlockValue ITSOerString}

##### 13.4.4.1 Report block Number

reportBlockNumber OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
MAX-ACCESS read-only

STATUS deprecated  
DESCRIPTION  
" <Definition>The block number for this row in the table. This value shall not exceed the value indicated by the maxReportBlocks object.  
<Supplanted by> ISO26048-1-DynObj.fdDynObjName (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1.1"  
::= { reportBlockEntry 1 }

#### 13.4.4.2 Report Block Status

reportBlockStatus OBJECT-TYPE  
SYNTAX NtcipRowStatusStatic  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition>The value of this object indicates the current status of this row in the table.  
<Supplanted by> ISO26048-1-DynObj.fdDynObjRowStatus (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1.2"

DEFVAL { invalid }  
::= { reportBlockEntry 2 }

#### 13.4.4.3 Report Block Description

reportBlockDescription OBJECT-TYPE  
SYNTAX OCTET STRING (SIZE(0..20))  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object may be used to define a description of this report block.  
<Supplanted by> ISO26048-1-DynObj.fdDynObjDescription (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1.3"

DEFVAL { "" }  
::= { reportBlockEntry 3 }

#### 13.4.4.4 Report Block Value

reportBlockValue OBJECT-TYPE  
SYNTAX ITSoerString  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition> An OER encoded string of all object values defined in reportObjectDefinitionTable, pointed at by reportOID (in reportID order) where the reportBlock IS reportBlockNumber AND the reportStatus IS available.  
<Supplanted by> ISO26048-1-DynObj.fdDynObjCurrentValue (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1.4"

DEFVAL { 'H' }  
::= { reportBlockEntry 4 }

## 13.5 TRAP MANAGEMENT

### 13.5.1 Trap Control

trapControl OBJECT-TYPE  
SYNTAX Integer32 (0..1)  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> The possible values are:  
0 - disable NTCIP traps  
1 - enable NTCIP traps  
The other values are reserved.  
  
<Supplanted by> ISO26048-1-Notification.fdNotificationsEnabled (ISO 26048-1)  
  
<Informative> The replacement object is defined as a TruthValue (RFC 2579), which is an enumeration where 1 = 'true' (in this case, enabled) and 2 = 'false' (in this case, disabled).  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.1"  
  
DEFVAL { 0 }  
 ::= { trapMgmt 1 }

### 13.5.2 Trap Data

trapData OBJECT-TYPE  
SYNTAX ITSOerString  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition> It contains an octet string (event notification) with octet trap sequence number (trapMgmtSeqNum) and octet trap manager index (trapMgmtManagerIndex), followed by one or more OER encoded sequences of ventide (eventConfigID), eventTime (globalTime of the occurrence of the event), eventLogTimeMilliseconds (fractional second of the occurrence of the event), and reported data (as pointed to by eventConfigLogOID). For aggregated trap messages (ackTrapChain and noackTrapChain) the trapData contains the octet trap sequence number (trapMgmtSeqNum) and octet trap manager index (trapMgmtManagerIndex), followed by from 1 to trapMaxAggregationSize triplets. The sequence to which OER encoding is applied is formally defined by the TrapDataStructure (see Section 6.4.1).  
  
<Supplanted by> ISO26048-1-Notification.fdNotificationData (ISO 26048-1)  
  
<Informative> The original syntax for this object was OerString as defined in NTCIP 8004 v02. NTCIP 8004 v03 recommends the use of ITSOerString in SMIV2 modules as it is formally defined as a textual convention in ISO 26048-1. Both OerString and ITSOerString resolve to OCTET STRING, so while the written syntax has changed, there is no impact on how it is encoded. The replacement object switches the order of the initial index and sequence number. It also uses a ITSDailyTimeStamp, which indicates time of day to the millisecond. It also includes a latency of the data and each data field is presented as a CHOICE of either the data value or the PDU error that the get operation generated.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.2"  
  
 ::= { trapMgmt 2 }

### 13.5.3 Trap Management Maximum Entries

```
trapMgmtMaxEntries OBJECT-TYPE
SYNTAX      Integer32 (1..255)
UNITS       "entries"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> The maximum number of entries in the trapMgmtTable.
  <Informative> The trapMgmtTable has been replaced with a dynamic table,
  which does not require an object indicating the maximum number of rows.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.3"
 ::= { trapMgmt 3 }
```

### 13.5.4 Trap Maximum Aggregation Events

```
trapMaxAggregationEvents OBJECT-TYPE
SYNTAX      Integer32 (1..255)
UNITS       "events"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> This object defines the maximum number of trap-events which
  can be aggregated.
  <Informative> The object does not have a parallel in ISO 26048-1.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.4"
 ::= { trapMgmt 4 }
```

### 13.5.5 Trap Maximum Aggregation Size

```
trapMaxAggregationSize OBJECT-TYPE
SYNTAX      Integer32 (1..1023)
UNITS       "octets"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> This object defines the maximum size (in bytes) of the
  aggregation chains that can be created during the aggregation process.
  <Supplanted by> ISO26048-1-Notification.fdNotificationsMaxSize (ISO 26048-
  1)
  <Informative> The replacement object is defined as an Unsigned32.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.5"
 ::= { trapMgmt 5 }
```

### 13.5.6 Trap Management Table

```
trapMgmtTable OBJECT-TYPE
SYNTAX      SEQUENCE OF TrapMgmtEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition> The table contains the list of management stations and their
  parameters where the agent traps are to be sent.
  <Table Type> static
  <Supplanted by> ISO26048-1-Notification.fdNotifyChannelTable (ISO 26048-1)
```

```
<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6"
 ::= { trapMgmt 6 }

trapMgmtEntry OBJECT-TYPE
SYNTAX      TrapMgmtEntry
MAX-ACCESS not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition> This defines a row in the trapMgmtTable.
  <Supplanted by> ISO26048-1-Notification.fdnotifyChannelEntry (ISO 26048-1)
  <Informative> The replacement table has a dual-index of an owner and name,
  both SnmpAdminStrings; it is separately assigned an ID, which more closely
  relates to the trapMgmtManagerIndex. In addition to the changes described
  below, the replacement table adds the following columns:
  - A command to clear the current channel queue
  - A StorageType for the row
  - A RowStatus for the dynamic table
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1"

INDEX      { trapMgmtManagerIndex }
 ::= { trapMgmtTable 1 }

TrapMgmtEntry ::= SEQUENCE {
  trapMgmtManagerIndex      Integer32,
  trapMgmtManagerPointer    Integer32,
  trapMgmtCommunityNamePointer Integer32,
  trapMgmtApplicationProtocol INTEGER,
  trapMgmtTransportProtocol INTEGER,
  trapMgmtPortNum           Integer32,
  trapMgmtMaxRetries        Integer32,
  trapMgmtRepeatInterval    Integer32,
  trapMgmtDelta              Integer32,
  trapMgmtQueueDepth        Integer32,
  trapMgmtLinkStateStatus   INTEGER,
  trapMgmtAntiStreamRate    Integer32,
  trapMgmtErrStatus         INTEGER,
  trapMgmtLostTraps         Counter32,
  trapMgmtRowStatus         NtcipRowStatusStatic,
  trapMgmtSeqNum            Integer32,
  trapMgmtSeqNumAck         Integer32}

```

### 13.5.6.1 Trap Manager Index

```
trapMgmtManagerIndex OBJECT-TYPE
SYNTAX      Integer32 (1..255)
MAX-ACCESS read-only
STATUS      deprecated
DESCRIPTION
  "<Definition> This object provides the index into the trapMgmtTable. This
  value shall not exceed the trapMgmtMaxEntries object value.
  <Supplanted by> ISO26048-1-Notification.fdnotifyChannelID (ISO 26048-1)
  <Informative> The replacement object is defined as an ITSUnsigned16.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.1"

```



```
::= { trapMgmtEntry 1 }
```

### 13.5.6.2 Trap Logical Name Translation Entry Pointer

trapMgmtManagerPointer OBJECT-TYPE

SYNTAX Integer32 (1..255)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> For UDP/IP stacks (trapMgmtTransportProtocol = 3), this value is equal to the logicalNameTranslationIndex for the logical name translation table entry where logicalNameTranslationName holds the logical name and logicalNameTranslationNetworkAddress holds the IP address of the destination management station for ntcip traps. Otherwise it is not used. This value shall not exceed the logicalNameTranslationTableMaxEntries object value.

<Supplanted by> ISO26048-1-Notification.fdNotifyChannelTarget (ISO 26048-1)

<Informative> The replacement object identifies an SNMP Target by its snmpTargetAddrName, as defined in RFC 3413. The snmpTargetAddrTable indicates the transport domain and address for the target.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.2"

DEFVAL { 1 }

```
::= { trapMgmtEntry 2 }
```

### 13.5.6.3 Trap Community Name Entry Pointer

trapMgmtCommunityNamePointer OBJECT-TYPE

SYNTAX Integer32 (1..255)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> This value is equal to the communityNameIndex for the community name table entry where communityNameUser holds the community name for ntcip traps sent to the destination management station. This value shall not exceed communityNamesMax object value.

<Supplanted by> SNMP-TARGET-MIB.snmpTargetParamsSecurityModel, snmpTargetParamsSecurityName, & snmpTargetParamsSecurityLevel (RFC 3413)

<Informative> In the replacement design, fdNotifyChannelTarget points to a row in the snmpTargetAddrTable, which points to a row in the snmpTargetParamsTable, which contains the security information to be used to communicate with the Target.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.3"

DEFVAL { 1 }

```
::= { trapMgmtEntry 3 }
```

### 13.5.6.4 Trap Application Layer Protocol

trapMgmtApplicationProtocol OBJECT-TYPE

SYNTAX INTEGER { other (1),  
snmp (2),  
sfmp (3) }

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> This object identifies the application layer protocol to use for TMP (Transportation Management Protocol) traps. The possible values

are:  
1 - other : not defined in this standard  
2 - snmp : use SNMPv1 Trap  
3 - sfmp : use SFMP Trap

<Supplanted by> SNMP-TARGET-MIB.snmpTargetParamsMPModel (RFC 3413)

<Informative> In the replacement design, fdNotifyChannelTarget points to a row in the snmpTargetAddrTable, which points to a row in the snmpTargetParamsTable, which contains snmpTargetParamsMPModel, which indicates a message processing model (i.e., SNMPv1 vs. SNMPv3). If the NTCIP (or NEMA) community desired, it could define its own processing model (e.g., something similar to SFMP with security) and assign it a value of (1206 \* 256) + id, where id is between 0 and 255 (see RFC 3411).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.4"

DEFVAL { snmp }  
::= { trapMgmtEntry 4 }

### 13.5.6.5 Trap Transport Layer Protocol

trapMgmtTransportProtocol OBJECT-TYPE  
SYNTAX INTEGER { other (1),  
t2 (2),  
udp (3) }

MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object identifies the transport profile to use for TMP traps. The possible values are:  
1 - other : not defined in standard  
2 - t2 : use T2 encapsulation to omit the port number  
3 - udp : use UDP/IP stack

<Supplanted by> SNMP-TARGET-MIB.snmpTargetAddrTDomain (RFC 3413)

<Informative> In the replacement design, fdNotifyChannelTarget points to a row in the snmpTargetAddrTable, which contains snmpTargetAddrTDomain, which identifies the transport domain to be used; registered domains can be found at <https://www.iana.org/assignments/snmp-number-spaces/snmp-number-spaces.xhtml>. For example, the snmpTLSTCPDomain is 1.3.6.1.6.1.8 and snmpDTLSUDPDDomain is 1.3.6.1.6.1.9.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.5"

DEFVAL { udp }  
::= { trapMgmtEntry 5 }

### 13.5.6.6 Trap Port Number

trapMgmtPortNum OBJECT-TYPE  
SYNTAX Integer32 (0..65535)  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> Port of the destination management station (e.g. 162 - default SNMP Trap port).

<Supplanted by> SNMP-TARGET-MIB.snmpTargetAddrTAddress (RFC 3413)

<Informative> In the replacement design, fdNotifyChannelTarget points to a row in the snmpTargetAddrTable, which contains snmpTargetAddrTAddress, which is defined in a format defined by the snmpTargetAddrTDomain. For

snmpTLSTCPDomain and snmpDTLSUDPDomain, the Taddress includes the port number; other protocols support similar values.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.6"

DEFVAL { 162 }  
::= { trapMgmtEntry 6 }

#### 13.5.6.7 Trap Maximum Retransmission Retries

trapMgmtMaxRetries OBJECT-TYPE

SYNTAX Integer32 (0..255)

UNITS "tries"

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> The maximum number of times an agent attempts to retransmit a trap before transitioning to the error state.

<Supplanted by> SNMP-TARGET-MIB.snmpTargetAddrRetryCount (RFC 3413)

<Informative> A value of one indicates that the agent attempts a maximum of two transmissions.

In the replacement design, fdNotifyChannelTarget points to a row in the snmpTargetAddrTable, which contains snmpTargetAddrRetryCount, which is an equivalent object.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.7"

DEFVAL { 0 }  
::= { trapMgmtEntry 7 }

#### 13.5.6.8 Trap Repeat Interval

trapMgmtRepeatInterval OBJECT-TYPE

SYNTAX Integer32 (0..255)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> The minimum number of seconds to wait before retransmitting a trap that has not been acknowledged. A value of zero (0) indicates an immediate retransmission of the trap.

<Supplanted by> SNMP-TARGET-MIB.snmpTargetAddrTimeout (RFC 3413)

<Informative> In the replacement design, fdNotifyChannelTarget points to a row in the snmpTargetAddrTable, which contains snmpTargetAddrTimeout, which indicates the timeout in hundredths of a second (up to 24 million seconds, which is almost a year).

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.8"

DEFVAL { 60 }  
::= { trapMgmtEntry 8 }

#### 13.5.6.9 Trap Repeat Interval Timeout Delta

trapMgmtDelta OBJECT-TYPE

SYNTAX Integer32 (0..255)

UNITS "seconds"

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> A number of seconds to be added to the total timeout for the next trap retransmission.

<Informative> In the replacement design, there is no equivalent to this object because as the definition of the timeout parameter indicates that it is implementation dependent.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.9"

DEFVAL { 60 }  
 ::= { trapMgmtEntry 9 }

#### 13.5.6.10 Trap Maximum Number of Queued Traps

trapMgmtQueueDepth OBJECT-TYPE

SYNTAX Integer32 (0..50)

UNITS "traps"

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> The maximum number of traps that can be queued for the Management station. Setting this value to zero flushes and disables the queue, and prevents any queueable traps from being sent.

<Supplanted by> ISO26048-1-Notification.fDNotifyChannelQueueDepth (ISO 26048-1)

<Informative> The replacement object has a syntax of Unsigned32.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.10"

DEFVAL { 1 }  
 ::= { trapMgmtEntry 10 }

#### 13.5.6.11 Trap State of Communications Link

trapMgmtLinkStateStatus OBJECT-TYPE

SYNTAX INTEGER { other (1),  
 ready (2),  
 pending (3),  
 error (4) }

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> This object contains the current link state of the manager registered in this row.

<Format> The states are defined as follows:

other (1) - not defined in this standard;

ready (2) - any trap can be sent to the manager (initial condition after power-on); if an ACK trap appears in the queue the agent sends the trap message to the manager, starts timer and internal retry counter, and sets the state to pending;

pending (3) - waiting for the manager to acknowledge the last ACK trap; NOACK and forced mode traps can be transmitted to the manager; if after all retries and timeouts the management station did not acknowledge an ACK trap message the agent sets the state to error

error (4) - an ACK trap has not been acknowledged within the specified number of retries for this management station. Only force mode traps are transmitted to the management station until the link state is reset to ready.

<Informative> The replacement design does not have a need for this object because it relies upon the native Confirmed-Class and Unconfirmed-Class PDUs. If the notification is sent via a Confirmed-Class, it is the

responsibility of the SNMP engine to discover missing confirmations and retry.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.11"

REFERENCE "NTCIP 1103 v03 Section 6.3"  
::= { trapMgmtEntry 11 }

#### 13.5.6.12 Trap Antistreaming Rate

trapMgmtAntiStreamRate OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
UNITS "traps/minute"  
MAX-ACCESS read-write  
STATUS deprecated

##### DESCRIPTION

"<Definition> The maximum number of traps that can be generated on a specific link (trap channel) in one minute. The agent shall reset the anti-streaming counter at the start of each minute. If the anti-streaming rate is reached the agent shall set the 'Trap channel anti-streaming mode activated' bit in the trapMgmtErrStatus, send the current trap and cease sending any additional traps on this link (trap channel) until the start of the next minute.

<Supplanted by> ISO26048-1-Notification.fdNotifyChannelAntiStreamRate (ISO 26048-1)

<Informative> The replacement object has a syntax of Unsigned32.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.12"

DEFVAL { 10 }  
::= { trapMgmtEntry 12 }

#### 13.5.6.13 Trap Error Status

trapMgmtErrStatus OBJECT-TYPE  
SYNTAX Integer32 (0..255)  
MAX-ACCESS read-only  
STATUS deprecated

##### DESCRIPTION

"<Definition> Trap channel status mask. When a bit = 1 the error status is true. When a bit = 0 the error status is false.

Bit 7: Reserved Bit  
Bit 6: Reserved Bit  
Bit 5: Reserved Bit  
Bit 4: Reserved Bit  
Bit 3: Reserved Bit  
Bit 2: Trap channel has trapMgmtLinkStateStatus = error  
Bit 1: Trap channel anti-streaming mode activated  
Bit 0: Trap channel queue full

<Informative> The replacement design does not have an equivalent object.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.13"

::= { trapMgmtEntry 13 }

#### 13.5.6.14 Trap Lost Trap Counter

trapMgmtLostTraps OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS deprecated

##### DESCRIPTION

"<Definition> Counter for the number of traps that have been discarded due to the queue for this trap channel being full.

<Supplanted by> ISO26048-1-Notification.fdNotifyChannelDroppedCount (ISO 26048-1)

<Informative> The replacement object expands the definition for dropping a notification packet for any reason before transmission. Note: Counters are not required to be zero-based.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.14"

::= { trapMgmtEntry 14 }

### 13.5.6.15 Trap Row Status

trapMgmtRowStatus OBJECT-TYPE  
SYNTAX NtcipRowStatusStatic  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> This object allows for the management of rows within the table.

<Supplanted by> ISO26048-1-Notification.fdNotifyChannelRowStatus (ISO 26048-1)

<Informative> The replacement object has a syntax of RowStatus, which provides a dynamic table.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.15"

DEFVAL { invalid }  
::= { trapMgmtEntry 15 }

### 13.5.6.16 Trap Sequence Number

trapMgmtSeqNum OBJECT-TYPE  
SYNTAX Integer32 (1..255)  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION

"<Definition> This object contains the sequence number of the last new trap that has been sent on this link. It is included in the trap message to assist a management station in identifying duplicate trap messages or detect when trap messages are missed. The agent shall increment this counter on each new trap message, but not retries, sent to the management station. The first trap message sent after a power-on reset or after trapMgmtRowStatus is successfully activated has trapMgmtSeqNum = 1. The sequence number is reset to 1 after a trap is sent with trapMgmtSeqNum == 255.

<Supplanted by> ISO26048-1-Notification.fdNotifyChannelSeqNum (ISO 26048-1)

<Informative> The replacement object has a syntax of Counter32.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.16"

REFERENCE "NTCIP 1103 v03 Section 6.2.6"  
::= { trapMgmtEntry 16 }

### 13.5.6.17 Trap Acknowledge Sequence Number

trapMgmtSeqNumAck OBJECT-TYPE  
SYNTAX Integer32 (0..255)  
MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> This object is set to the sequence number of the trap that is being acknowledged by the management station. If the value set equals either the current value of trapMgmtSeqNumAck or zero (0), then the agent changes the trapMgmtLinkStateStatus from pending or error to ready and set this object to zero (0) indicating no ack traps are awaiting acknowledgement. If this object is set to any other value, then the agent ignores the set and continue the normal acknowledge process including retries.

<Informative> Within the replacement design, it is the responsibility of the SNMP engine to ensure that Confirmed-Class notifications are acknowledged.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.6.1.17"

REFERENCE "NTCIP 1103 v03 Section 6.2.7"  
 ::= { trapMgmtEntry 17 }

### 13.5.7 Trap Table

trapTable OBJECT-TYPE

SYNTAX SEQUENCE OF TrapEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> The table specifies the trap operational mode for each event registered in the eventLogConfigTable (NTCIP 1201) and necessary to run NTCIP Trap operations.

Each entry in the trapTable can be individually enabled and disabled and is assigned its own mode per trap channel. An event can be sent to multiple trap channels by creating multiple entries in the trapTable for the same event, each with a different trapMgmtManagerIndex.

<Table Type> static

<Supplanted by> ISO26048-1-Notification.fldNotifyFactoryTable (ISO 26048-1)

<Informative> The replacement table is dynamic.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.7"

::= { trapMgmt 7 }

trapEntry OBJECT-TYPE

SYNTAX TrapEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"<Definition> This defines a row in the trapTable.

<Supplanted by> ISO26048-1-Notification.fldNotifyFactoryEntry (ISO 26048-1)

<Informative> The replacement table uses a dual index with owner and name.

It is not directly connected to the trigger table; it is called by it as one potential action. The replacement table adds the following columns:

- An event id that provides a concise identifier for the event, similar to eventConfigID.
- The notification channel to use; i.e., in the new design, a specific event can result in multiple notifications (presumably to different managers)
- The object to be sent (i.e., the new design allows for different

```
information to be logged and to be sent)
- A StorageType column
- A RowStatus column

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.7.1"
INDEX      { eventConfigID, trapMgmtManagerIndex }
::= { trapTable 1 }

TrapEntry ::= SEQUENCE {
    trapDestEnable      Integer32,
    trapMode            Integer32,
    trapAggregationTime Integer32,
    trapCounter         Counter32}

```

### 13.5.7.1 Trap Destination Enabled

```
trapDestEnable OBJECT-TYPE
SYNTAX      Integer32 (0..1)
MAX-ACCESS read-write
STATUS      deprecated
DESCRIPTION
    "<Definition> Setting this object to one (1) enables the trap
    (eventConfigID) for transmission through the trap channel
    (trapMgmtManagerIndex).
    A value of zero (0) disables events with eventConfigID from being acted on
    with respect to the transmission of traps through the specific trap channel
    identified in the trapMgmtTable.

    <Supplanted by> ISO26048-1-Notification.fdnotifyFactoryRowStatus (ISO
    26048-1)

    <Informative> The transmission (or queuing) of the trap also depends on
    entries in the eventLogConfigTable and the trapMgmtTable. Entries in all
    three tables are required to be enabled for the trap to be queued or
    transmitted.
    The replacement object allows creation and deletion of the entry as well.

    <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.7.1.1"
DEFVAL      { 0 }
::= { trapEntry 1 }

```

### 13.5.7.2 Trap Mode

```
trapMode OBJECT-TYPE
SYNTAX      INTEGER { forced (1),
                    ack-noQ (2),
                    ack (3),
                    noack-noQ (4),
                    noack (5),
                    ack-Aggr (6),
                    noack-Aggr (7) }
MAX-ACCESS read-write
STATUS      deprecated
DESCRIPTION
    "<Definition> This object defines a number of different methods for
    handling the trap data prior to or as it is transferred to the trap
    channel.

    <Format> The trap modes are as follows:
    1 - forced: NOACK, the trap is sent (not queued) regardless of the

```



trapMgmtLinkStateStatus.

2 - ack\_noQ: ACK, the trap is sent if the trapMgmtLinkStateStatus is READY; otherwise the trap is simply dropped (lost); however, if the trap is transmitted, it is acknowledged and the retry mechanism is invoked.

3 - ack: ACK, the trap is sent if the trapMgmtLinkStateStatus is READY, otherwise the trap message shall be queued.

4 - noack\_noQ: NOACK, the trap is sent if the trapMgmtLinkStateStatus is READY or PENDING; otherwise the trap is simply dropped (lost).

5 - noack: NOACK, the trap is sent if the trapMgmtLinkStateStatus is READY or PENDING; it is queued if the channel status is ERROR.

6 - ack\_Aggr: ACK, the trap data (eventConfigID, timestamp, and reported data) are added to the ackTrapChain for transmission to the management station when a chain termination condition occurs and the trapMgmtLinkStateStatus is READY. ackTrapChains are not queued and are transmitted prior to any queued traps.

7 - noack\_Aggr: NOACK, the trap data (eventConfigID, timestamp, and reported data) are added to the noackTrapChain for transmission to the management station when a chain termination condition occurs and the trapMgmtLinkStateStatus is READY or PENDING; otherwise the noackTrapChain grows accepting new traps.

8+: reserved (not used in a proprietary manner)

<Supplanted by> ISO26048-1-Notification.fdnotifyFactoryAckEnabled & fdnotifyFactoryQueueEnabled (ISO 26048-1)

<Informative> The replacement object values map to the values of this object as follows:

forced - there is no true parallel because the queue in trapMgmt is an artifact of the management of acknowledged traps with no defined role related to the anti-streaming rate. Within fdNotifyChannel, the queue solely relates to the antistreaming rate because the replacement design relies on the SNMP engine to manage responses and the link state is effectively always READY. There is no way to force an override of anti-streaming.

Ack\_noQ - AckEnabled = true and QueueEnabled = false: if the anti-streaming rate has not been exceeded, the notification is sent as an SNMP Inform message; otherwise, the notification is dropped.

Ack - AckEnabled = true and QueueEnabled = true: if the anti-streaming rate has not been exceeded, the notification is sent as an SNMP Inform message; otherwise, it is added to the queue. If the addition of the packet to the queue would cause the queue depth to be exceeded, the oldest notification is deleted to make room for the most recent notification.

Noack\_noQ - AckEnabled = false and QueueEnabled = false: if the anti-streaming rate has not been exceeded, the notification is sent as an SNMP Trap message; otherwise, the notification is dropped.

Noack - AckEnabled = false and QueueEnabled = true: if the anti-streaming rate has not been exceeded, the notification is sent as an SNMP Trap message; otherwise, the notification is added to the queue. If the addition of the packet to the queue would cause the queue depth to be exceeded, the oldest notification is deleted to make room for the most recent notification.

Ack\_Aggr - AckEnabled=true, QueueEnabled=false, AggregationTime>0: Same as ack\_noQ; there is no queue-jumping logic defined.

Noack\_Aggr - AckEnabled=false, QueueEnabled=false, AggregationTime>0: Same as noack\_noQ; there is no queue-jumping logic defined.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.7.1.2"

REFERENCE "NTCIP 1103 v03, Sections 6.2.3 and 6.2.4"

```
DEFVAL      { forced }  
::= { trapEntry 2 }
```

### 13.5.7.3 Trap Maximum Aggregation Time

```
trapAggregationTime OBJECT-TYPE  
SYNTAX      Integer32 (0..65535)  
UNITS       "seconds"  
MAX-ACCESS  read-write  
STATUS      deprecated
```

#### DESCRIPTION

"<Definition> maximum time (in seconds) that this trap can wait for transmission while being aggregated within a trap chain (for noack\_Aggr and ack\_Aggr traps only). The value of zero (0) indicates immediate trap chain transmission. The first trap with a mode specifying 'timed' aggregation starts this timer. The timer is reset when the trap chain is sent. The aggregation timers for each entry in the trap chain time concurrently and the first one to expire causes the entire trap chain to be sent to the management station.

<Informative> This is larger than 8 bits to allow aggregation times to support 5 minutes and longer.

<Supplanted by> ISO26048-1-Notification.fDNotifyFactoryAggregationTime (ISO 26048-1)

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.7.1.3"

```
REFERENCE "NTCIP 1103 v03 Section 6.2.4"
```

```
DEFVAL      { 0 }  
::= { trapEntry 3 }
```

### 13.5.7.4 Trap Counter

```
trapCounter OBJECT-TYPE  
SYNTAX      Counter32  
MAX-ACCESS  read-only  
STATUS      deprecated
```

#### DESCRIPTION

"<Definition> this keeps track of the number of eventConfigID traps sent to the trap channel identified by the trapMgmtManagerIndex since the last (power-on) reset. It is incremented with the transmission or queuing (or addition to a trap chain) of each trap. By reading this parameter, a management station can verify the number of traps triggered for transmission or queuing by this event for this trap channel.

<Supplanted by> ISO26048-1-Notification.fDNotifyFactoryEventCount

<Informative> Counters are not zero-based.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.7.1.4"

```
::= { trapEntry 4 }
```

## 13.6 NTCIP TRAP DATA

### 13.6.1 Event Trap

```
trapEvent NOTIFICATION-TYPE  
OBJECTS    { trapData }  
STATUS     deprecated
```

#### DESCRIPTION

"<Definition> Indicates that one of the user-defined event specified in the eventLogConfigTable has occurred. The generation of the trap is governed by

the rules defined in SNMP, Section 6 above, and the trapMgmtTable.  
The instances of the variables associates with this trap shall indicate those associated with the event notification being sent.

<Supplanted by> ISO26048-1-Notification.fdnNotificationPacket  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.4.2.0.1"

::= { ntcipTrapNotifications 1 }

### 13.7 CLEAR EVENT DATA

#### 13.7.1 Clear Event Class

eventClearClasses OBJECT-TYPE  
SYNTAX Integer32 (0..255)  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition> This object identifies the event class to be cleared from the report node. A SET of n = 5..255, n <= maxEventClasses shall cause all information related to that class to be cleared from the report node. This includes clearing the event class table of eventClassNumber = n data, clearing all event configurations related to eventClassNumber = n, and clearing all event log entries for class n. A SET of 0 shall clear all classes as described. That is, completely clear the report node with the exception that the preconfigured event classes, their configurations, and their preconfigured event log entries are not cleared. A GET shall always return zero (0).

If a device standard, Classes 1..4 are preconfigured and cannot be cleared. An attempt to clear Classes 1..4 shall return badValue. A value of n > maxEventClasses or > 255 if maxEventClasses is not configured, shall also return badValue.

<Supplanted by> ISO26048-1-Log.fdlLogsDeleteAllConfiguration & fdLogManagerRowStatus (ISO 26048-1)

<Informative> An administrator can delete the configuration and data for all logs or an owner can choose to delete any specific configuration and associated data.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8.1"

::= { eventClearObjects 1 }

#### 13.7.2 Clear Event Configuration

eventClearConfiguration OBJECT-TYPE  
SYNTAX Integer32 (0..65535)  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION

"<Definition>This object contains the event configuration(s) to clear from the report node. A SET of n = 1..65535, n <= maxEventLogConfigs shall cause all information related to that configuration to be cleared from the report node. This includes clearing the event configuration table for all eventConfigID = n data, and clearing all event log entries for eventConfigID = n. A SET of 0 shall clear all configurations within the device as described (i.e. completely clear the report node with the exception that the eventClassTable is and preconfigured event configuration are not altered). A GET shall always return zero (0).

<Supplanted by> LOG-MIB. fdLogEventFactoryRowStatus (ISO 26048-1) & ISO26048-1-CondTrigger.fCondTriggerRowStatus (ISO 26048-1)

<Informative> This object cannot be included in a block object. The device shall respond with badValue if the eventLogConfig = n does not exist. The device shall respond with badValue if an attempt to clear a preconfigured event log entries is made.

A user can separately delete the trigger and the factory.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8.2"

::= { eventClearObjects 2 }

### 13.7.3 Clear Event Log Table

eventClearLog OBJECT-TYPE

SYNTAX Integer32 (0..1)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to clear the eventLogTable. A SET of zero has no effect on the eventLogTable. A SET = 1 shall cause all event log entries to be deleted from the eventLogTable. Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).

<Supplanted by> ISO26048-1-Log.fLogsClearAllLogs

<Informative> The replacement object uses a syntax of TruthValue.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8.3"

::= { eventClearObjects 3 }

### 13.7.4 Clear Report Objects

clearReportObjects OBJECT-TYPE

SYNTAX Integer32 (0..1)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to effectively clear the report object and report block tables. A SET of zero has no effect on the tables. A SET = 1 shall set the row status object of all rows within both reportObjectDefinitionTable and reportBlockTable to invalid, effectively clearing the tables in one action. Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).

<Informative> There is no parallel to this object in the replacement design.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8.4"

::= { eventClearObjects 4 }

### 13.7.5 Clear Report Block Table

clearReportBlockTable OBJECT-TYPE

SYNTAX Integer32 (0..1)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to effectively clear the report block table. A SET of zero (0) has no effect on the tables. A SET of

one (1) shall set the row status object of all rows within reportBlockTable to invalid, effectively clearing the table in one action. Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).

<Informative> There is no parallel to this object in the replacement design.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8.5"

::= { eventClearObjects 5 }

### 13.7.6 Clear Watch Objects

clearWatchObjects OBJECT-TYPE

SYNTAX Integer32 (0..1)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to effectively clear the watch object and watch block tables. A SET of zero has no effect on the tables. A SET = 1 shall set the row status object of all rows within both watchObjectDefinitionTable and watchBlockTable to 'invalid' effectively clearing the tables in one action. Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).

<Informative> There is no parallel to this object in the replacement design.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8.6"

::= { eventClearObjects 6 }

### 13.7.7 Clear Watch Block Table

clearWatchBlockTable OBJECT-TYPE

SYNTAX Integer32 (0..1)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to effectively clear the watch block table. A SET of zero (0) has no effect on the tables. A SET of one (1) shall set the row status object of all rows within watchBlockTable to 'invalid' effectively clearing the table in one action. Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).

<Informative> There is no parallel to this object in the replacement design.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8.7"

::= { eventClearObjects 7 }

### 13.7.8 Clear Trap Management Table

clearTrapMgmtTable OBJECT-TYPE

SYNTAX Integer32 (0..1)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to effectively clear the trap management table and, as a consequence, the trap table. A SET of zero (0) has no effect on the tables. A SET of one (1) shall set the row status

object of all rows within trapMgmtTable to 'invalid' effectively clearing the table in one action. Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).  
Note: Because the trapMgmtIndex is also an index of trapTable, this action also effectively removes all rows of the trapTable.

<Informative> There is no parallel to this object in the replacement design.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.8.8"

```
::= { eventClearObjects 8 }  
END
```

## Section 14 DEPRECATED RECORDING MECHANISM MIB

The text provided from Section 14.1 through the end of Section 14 (except the headings) constitutes the standard NTCIP1201-RecMech MIB.

**\*\* All objects contained in this MIB are deprecated. \*\***

### 14.1 HEADER

```
NTCIP1201-RecMech DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY, OBJECT-TYPE, Integer32, Unsigned32,
Opaque, zeroDotZero
                                FROM SNMPv2-SMI
                                -- RFC 2578
application
                                FROM NTCIP8004-Transportation;

recMech MODULE-IDENTITY
LAST-UPDATED "20230701145436Z"
ORGANIZATION "NTCIP BSP2 WG"
CONTACT-INFO
  "name: NTCIP Coordinator
   email: ntcip@nema.org
   postal: National Electrical Manufacturers Association
           1300 North 17th Street, Suite 1752
           Rosslyn, Virginia 22209-3801"
DESCRIPTION
  "<Definition> This MIB defines the SMIV2 representation of the
  NTCIP1103v0352-recMech MIB, which was defined in NTCIP 1103. This MIB
  defines objects related to the recording mechanism functions that are found
  in devices.
  This MIB was deprecated in NTCIP 1201 v04 due to security issues in the
  structure of the MIB. The objects have been replaced by objects in the
  NTCIP1201v04-recMech MIB.
  *** All objects in this MIB have been deprecated. ***
  <Informative> The recording mechanism class table is presented first to
  ease -- the readability of the standard.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9"
REVISION "20230701145436Z"
DESCRIPTION
  "NTCIP 1201 v04 - Upgraded format to SMIV2 and deprecated all objects."
REVISION "201612310000Z"
DESCRIPTION
  "NTCIP 1103 v03 - Original version of this MIB."
 ::= { application 9 }
```

### 14.2 OBJECTS

#### 14.2.1 Maximum Recording Mechanism Classes Parameter

```
maxRecClasses OBJECT-TYPE
SYNTAX      Integer32 (1..254)
UNITS       "RecClasses"
```

MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition> The object defines the number of rows in the recClassTable that this device supports. This is a static table.  
  
<Informative> The recClassTable has been replaced with a dynamic table, which does not require an object indicating the maximum number of rows.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.1"

::= { recMech 1 }

#### 14.2.2 Recording Mechanism Class Table

recClassTable OBJECT-TYPE  
SYNTAX SEQUENCE OF RecClassEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition> This table is used to configure recording mechanism limits and recording table maintenance.  
  
<Table Type> static  
  
<Supplanted by> ISO26048-1-Recording.fRecordingClassTable  
  
<Informative> The replacement table is dynamic.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2"

::= { recMech 2 }

recClassEntry OBJECT-TYPE  
SYNTAX RecClassEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object defines a row in the Recording Mechanism Class Table  
  
<Supplanted by> ISO26048-1-Recording.fRecordingClassEntry  
  
<Informative> The replacement table has a dual index consisting of an owner and a name, both SnmpAdminStrings. The table also adds columns for StorageType and RowStatus.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2.1"

INDEX { recClassNumber }  
::= { recClassTable 1 }

RecClassEntry ::= SEQUENCE {  
    recClassNumber Integer32,  
    recClassLimit Integer32,  
    recClassClearTime Unsigned32,  
    recClassDescription OCTET STRING,  
    recClassNumRecordings Integer32,  
    recClassRecordingCounter Integer32}

#### 14.2.2.1 Recording Mechanism Class Number Parameter

recClassNumber OBJECT-TYPE  
SYNTAX Integer32 (1..254)



```
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
  "<Definition>This is a class value that is to be configured.
  <Supplanted by> ISO26048-1-Recording.fdRecordingClassName
  <Informative> The replacement object is an SnmpAdminString.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2.1.1"
 ::= { recClassEntry 1 }
```

#### 14.2.2.2 Recording Mechanism Class Limit Parameter

```
recClassLimit OBJECT-TYPE
SYNTAX Integer32 (0..254)
UNITS "recordings"
MAX-ACCESS read-write
STATUS deprecated
DESCRIPTION
  "<Definition>This object specifies the maximum number of recordings of the
  associated class to store in the device. Once the limit is reached, the
  oldest recording of the matching class (based on recordingTriggerTime) is
  overwritten by any new recording of the same class. If the value of this
  object is set to a number smaller than the current number of rows within
  this class in the recRecordingTable, then the oldest entries shall be
  lost/deleted. The sum of all recording mechanism class limits shall not
  exceed the maxRecRecordings object; if a SET operation to this object
  causes the sum of recClassLimit objects to exceed maxRecRecordings, then
  the agent shall respond with a genErr.
  The recording cannot be logged if the recClass has an recClassLimit of zero
  (0).
  <Supplanted by> ISO26048-1-Recording.fdRecordingClassSizeLimit
  <Informative> The replacement object defines a maximum size in octets
  rather than the number of recordings.
  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2.1.2"
 ::= { recClassEntry 2 }
```

#### 14.2.2.3 Recording Mechanism Class Clear Time Parameter

```
recClassClearTime OBJECT-TYPE
SYNTAX Unsigned32
UNITS "seconds"
MAX-ACCESS read-write
STATUS deprecated
DESCRIPTION
  "<Definition>This object is used to clear multiple recordings from the
  recRecordingTable. All completed recordings of this class that have a
  recRecordingTriggerTime equal to or less than this object shall be cleared
  from the recRecordingTable. If this object has a value greater than the
  current value of globalTime, it shall prevent the triggering of any
  recordings of this class.
  <Supplanted by> ISO26048-1-Recording.fdRecordingClassClearDate &
  recMechV2ClassClearTime
  <Informative> The SMIV1 syntax for this object was Counter; however, by
  convention, an SNMPv1 Counter is not writable and SNMPv3 prohibits set
  operations. The syntax SMIV2 syntax for this object has been updated to
```

Unsigned32 to avoid this conflict. The superseding objects provide a date/time pair.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2.1.3"

```
DEFVAL      { 0 }  
 ::= { recClassEntry 3 }
```

#### 14.2.2.4 Recording Mechanism Class Description Parameter

recClassDescription OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object specifies a description of the class in ASCII characters.

<Supplanted by> ISO26048-1-Recording.fidRecordingClassDescription

<Informative> The replacement object uses a syntax of SnmpAdminString to clearly indicate that this should be readable text in any language.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2.1.4"

```
 ::= { recClassEntry 4 }
```

#### 14.2.2.5 Recording Mechanism Class Number of Rows in Recording Table Parameter

recClassNumRecordings OBJECT-TYPE

SYNTAX Integer32 (0..255)

UNITS "recordings"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The number of recordings for this class that currently exist in the recRecordingTable.

<Supplanted by> ISO26048-1-Recording.fidRecordingClassNumRecordings

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2.1.5"

```
 ::= { recClassEntry 5 }
```

#### 14.2.2.6 Class Recording Counter Parameter

recClassRecordingCounter OBJECT-TYPE

SYNTAX Integer32 (0..65535)

UNITS "recordings"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> This object is a counter that gets incremented every time a recording occurs for this class; it shall initialize to zero at power up. The value shall roll over each time it exceeds the maximum of 65535.

<Supplanted by> ISO26048-1-Recording.fidRecordingClassRecordingCtr

<Informative> The replacement object has a syntax of ZeroBasedCounter32.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2.1.6"

```
 ::= { recClassEntry 6 }
```

#### 14.2.3 Maximum Recording Configurations

maxRecConfigs OBJECT-TYPE

SYNTAX Integer32 (1..65534)

```
UNITS      "RecordType"
MAX-ACCESS read-only
STATUS     deprecated
DESCRIPTION
  "<Definition>The number of rows that exist in the static
  recMechV2RecordingConfig table for this device.

  <Informative> The recMechV2RecordingConfigTable has been replaced with a
  dynamic table, which does not require an object indicating the maximum
  number of rows.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.3"
 ::= { recMech 3 }
```

#### 14.2.4 Minimum Recording Sample Period

```
recMinSamplePeriod OBJECT-TYPE
SYNTAX      Integer32 (1..65535)
UNITS       "0.1 milliseconds"
MAX-ACCESS read-only
STATUS     deprecated
DESCRIPTION
  "<Definition> The minimum sample period for recordings supported by the
  device in units of 0.1 milliseconds.

  <Supplanted by> NTCIP1201-RecMechV2.recMechMinSamplePeriod

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.4"
 ::= { recMech 4 }
```

#### 14.2.5 Maximum Recording Sample Period

```
recMaxSamplePeriod OBJECT-TYPE
SYNTAX      Integer32 (1..65535)
UNITS       "0.1 milliseconds"
MAX-ACCESS read-only
STATUS     deprecated
DESCRIPTION
  "<Definition>The maximum sample period for recordings supported by the
  device in units of 0.1 milliseconds.

  <Supplanted by> NTCIP1201-RecMechV2.recMechMaxSamplePeriod

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.5"
 ::= { recMech 5 }
```

#### 14.2.6 Recording Sample Period Resolution

```
recSamplePeriodResolution OBJECT-TYPE
SYNTAX      Integer32 (1..65535)
UNITS       "0.1 milliseconds"
MAX-ACCESS read-only
STATUS     deprecated
DESCRIPTION
  "<Definition>The sample period resolution for recordings supported by the
  device in units of 0.1 milliseconds. Allowable sample periods are
  restricted to (recMinSamplePeriod + recSamplePeriodResolution * n) where n
  is integer, 0 <= n, and n <= (recMaxSamplePeriod-
  recMinSamplePeriod)/recSamplePeriodResolution

  <Supplanted by> ISO26048-1-Recording.fdRecordingSamplePeriodResolution

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.6"
```

```
::= { recMech 6 }
```

#### 14.2.7 Recording Configuration Table

```
recConfigTable OBJECT-TYPE
SYNTAX      SEQUENCE OF RecConfigEntry
MAX-ACCESS not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition>A table containing Recording Mechanism Configuration
  information. The number of rows in this table is equal to the maxRecConfigs
  object. This table defines the parameters that the device monitors to
  create a recording.

  <Table Type> static

  <Supplanted by> ISO26048-1-Recording.fRecordingFactoryTable & ISO26048-1-
  CondTrigger.fCondTriggerTable (ISO 26048-1)

  <Informative> The replacement design divides the recConfigTable into two
  tables: one that defines triggers that can be used to start recordings or
  other actions and a second table that defines what to record when the
  recording action is triggered.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7"
```

```
::= { recMech 7 }
```

```
recConfigEntry OBJECT-TYPE
SYNTAX      RecConfigEntry
MAX-ACCESS not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition>This object defines an entry in the recording configuration
  table.

  <Supplanted by> ISO26048-1-Recording.fRecordingFactoryEntry & ISO26048-1-
  CondTrigger.fCondTriggerEntry (ISO 26048-1)

  <Informative> The replacement trigger table has two indicies: an owner and
  a name. The replacement recording factory table has three indicies, an
  owner, the class name, and the factory name.
  In addition to the other changes described, the trigger table adds the
  following columns:
  - a textual description of the trigger
  - an indication whether the comparison is based on the current object value
  or a delta from its previous reading
  - an octet-based comparison value since SNMPv3 discourages the use of
  Opaque
  - a wildcard that allows defining the same condition on multiple comparison
  OIDs (e.g., all rows of a table)
  - indications of the target and context of the comparison object; in other
  words, the comparison can be performed by a proxy agent or can reference
  another device to get the object value to compare against
  - the frequency at which the comparison is made
  - a truthDuration that allows the configuration to require the evaluation
  to be true for some length of time prior to firing the trigger.
  - startup states that define whether the triggers startup in a fired or
  unfired state (for hysteresis, there are two startups)
  - a pointer to the action table that identifies the action(s) to be
  performed.
```

- an error message object that allows a device to report configuration errors.
- counters for the number of times the trigger has fired, had evaluation errors, and activation errors.
- an indication of the type of storage to use
- a RowStatus object. The recMech factory table adds the following columns:
- the security data used to activate the row and that will be used to retrieve values for the recording
- object contexts, which allows a proxy/hrbrid agent to capture information from another context
- A StorageType object

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1"

```
INDEX      { recConfigID }  
 ::= { recConfigTable 1 }
```

```
RecConfigEntry ::= SEQUENCE {  
    recConfigID      Integer32,  
    recConfigClass   Integer32,  
    recConfigMode    INTEGER,  
    recConfigCompareValue Integer32,  
    recConfigCompareValue2 Integer32,  
    recConfigCompareOID OBJECT IDENTIFIER,  
    recConfigRecordOID OBJECT IDENTIFIER,  
    recConfigTriggerPoint Integer32,  
    recConfigSamplePeriod Integer32,  
    recConfigSampleOID OBJECT IDENTIFIER,  
    recConfigNumEntries Integer32,  
    recConfigAction  INTEGER,  
    recConfigStatus  INTEGER}
```

#### 14.2.7.1 Recording Configuration ID Parameter

```
recConfigID OBJECT-TYPE  
SYNTAX      Integer32 (1..65534)  
MAX-ACCESS read-only  
STATUS      deprecated  
DESCRIPTION  
    "<Definition>This object contains the row number which is used to identify  
    the recording associated with this row in the recConfigTable. The number of  
    recording configuration IDs shall not exceed the value indicated in the  
    maxRecConfigs object.  
  
    <Supplanted by> ISO26048-1-Recording.fidRecordingFactoryName  
  
    <Informative> The replacement object is an SnmpAdminString  
  
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.1"  
 ::= { recConfigEntry 1 }
```

#### 14.2.7.2 Recording Configuration Class Parameter

```
recConfigClass OBJECT-TYPE  
SYNTAX      Integer32 (1..254)  
MAX-ACCESS read-write  
STATUS      deprecated  
DESCRIPTION  
    "<Definition> This object contains the class value to assign to the  
    recording associated with this row in the recording configuration table."
```

This value is used in the recording table to organize various recordings defined in this table into logical groupings. This value shall not exceed the maxRecClasses object value.

<Supplanted by> ISO26048-1-Recording.fidRecordingClassName

<Informative> A recording cannot be recorded if the RecClass has an recClassLimit of zero (0).

The replacement object is an index for the table

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.2"

DEFVAL { 1 }  
::= { recConfigEntry 2 }

#### 14.2.7.3 Recording Configuration Mode Parameter

recConfigMode OBJECT-TYPE

SYNTAX INTEGER { other (1),  
onChange (2),  
greaterThanValue (3),  
smallerThanValue (4),  
hysteresisBound (5),  
periodic (6),  
andedWithValue (7) }

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object specifies the mode of operation for this recording. The modes are defined as follows:

Value	Description
other	the recording mode of operation is not described in this standard, refer to the device manual.
onChange	trigger a recording when the object value referenced by recConfigCompareOID changes. The values of recConfigCompareValue and recConfigCompareValue2 are ignored in this mode.
greaterThanValue	trigger a recording when the object value referenced by recConfigCompareOID becomes greater than the value of recConfigCompareValue for the time (tenth seconds) defined by recConfigCompareValue2 (zero means immediate logging).
smallerThanValue	trigger a recording when the object value

referenced by recConfigCompareOID becomes less than the value of recConfigCompareValue for the time (tenth seconds) defined by recConfigCompareValue2 (zero means immediate logging).

**hysteresisBound** trigger a recording when the object value referenced by recConfigCompareOID becomes less than or greater than the bound values. The lowerbound value is the lower value of recConfigCompareValue and recConfigCompareValue2; the upperbound value is the higher value of the two values.

When the object value becomes greater than the upper bound value, subsequent triggering of upperbound conditions shall not occur until the object value becomes less than the lower bound value.

When the object value becomes less than the lower bound value, subsequent triggering of lowerbound conditions shall not occur until the object value becomes greater than the upper bound value.

**periodic** trigger a recording every x seconds, where x is defined by the value stored in recConfigCompareValue. The values stored in recConfigCompareValue2 and recConfigCompareOID are ignored in this mode.

**andedWithValue** trigger a recording when the object value referenced by recConfigCompareOID ANDED with the value of recConfigCompareValue is NOT equal to zero for the time (tenth seconds) defined by recConfigCompareValue2 (zero means immediate logging). This allows

monitoring of a specific bit; the condition becomes true anytime that any one of the selected bits become true.

<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerMode (ISO 26048-1)

<Informative> The replacement object adds the following modes: equal, notEqual, creation, deletion. It also distinguishes between an integer-based and octet-string-based bitwise comparison, which is required due to the elimination of Opaque.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.3"

DEFVAL { onChange }  
 ::= { recConfigEntry 3 }

#### 14.2.7.4 Recording Configuration Compare Value Parameter

recConfigCompareValue OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object contains the comparison value to use with recConfigMode values (greaterThanValue, smallerThanValue, hysteresisBound). No value within this object is necessary when the recConfigMode-object has the value onChange (2).

<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerValue (ISO 26048-1)

<Informative> The value is a signed, 4-octet integer

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.4"

DEFVAL { 0 }  
 ::= { recConfigEntry 4 }

#### 14.2.7.5 Recording Configuration Compare Value 2 Parameter

recConfigCompareValue2 OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>If the recConfigMode is set to hysteresisBound, this object specifies the second comparison value for the hysteresis. If the recConfigMode is set to greaterThanValue, smallerThanValue, or andedWithValue, this object specifies the time (in tenth of seconds, +1 tenth / -0 tenths) for which the samples used for comparison are true prior to the triggering condition becoming true. If the recConfigMode is set to onChange or periodic, the value of this object shall be ignored. The amount of time the condition istrue is measured in tenths of a second. The accuracy of this timer is limited to +1 tenth of a second and:0 tenths of a second. If the trigger is true for at least the time shown in this parameter +1 tenth of a second, the condition shall trigger a recording. It is recognized that some designs only sample the condition periodically, in which case the condition is deemed true for at least the time indicated by this object before the trigger becomes true and the trigger shall always become true if the condition is true for a duration equal to the value shown in this object plus 1 tenth of a second.



<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerValue2 (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.5"  
DEFVAL { 0 }  
::= { recConfigEntry 5 }

#### 14.2.7.6 Recording Configuration Compare Object Identifier Parameter

recConfigCompareOID OBJECT-TYPE  
SYNTAX OBJECT IDENTIFIER  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object contains the object identifier which references the value against which the comparison is made. If the recConfigMode is set to periodic, the value of this object shall be ignored. If the recConfigMode is set to greaterThanValue, smallerThanValue or hysteresisBound, this object is required to reference an object whose SYNTAX resolves to a ranged or unranged INTEGER. As with all other objects that are sub-ranged by a given implementation, an agent should return a badValue error if it receives a set command indicating a OID which is not supported by the implementation or which is not zeroDotZero.  
<Supplanted by> ISO26048-1-CondTrigger.fdCondTriggerObject (ISO 26048-1)  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.6"  
DEFVAL { zeroDotZero }  
::= { recConfigEntry 6 }

#### 14.2.7.7 Recording Configuration Record Object Identifier Parameter

recConfigRecordOID OBJECT-TYPE  
SYNTAX OBJECT IDENTIFIER  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
" <Definition> This object contains the object identifier which indicates what value to record in a recording (e.g., signal states). As with all other objects that are sub-ranged by a given implementation, an agent should return a badValue error if it receives a set command indicating a value which is not supported by the implementation. The valid value range of this object shall not include any values, other than zeroDotZero, that do not correspond to objects that may exist within the agent, although it may be further restricted.  
The valid value range of this object shall not include objects under the following nodes:  
Security - { nema transportation devices global security }  
CHAP - { nema transportation protocols layers chap }  
<Supplanted by> ISO26048-1-Recording.fdRecordingFactorySampleOID  
<Informative> The replacement object does not constrain values other than requiring the user to have rights to access the parameter when configuring and when sampling.  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.7"  
DEFVAL { zeroDotZero }  
::= { recConfigEntry 7 }

#### 14.2.7.8 Recording Configuration Trigger Point Parameter

recConfigTriggerPoint OBJECT-TYPE

SYNTAX Integer32 (0..100)  
UNITS "percent"  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
"<Definition> This object contains the value of the recording trigger point in percent relative to the recConfigNumEntries. The device needs to collect pre-event records prior to the trigger occurring and ends the recording after recConfigNumEntries have been recorded. A value of zero (0) means to start the recording once the trigger condition occurs whereas a value of 100 means to stop the recording with the last record being the one collected immediately following the trigger condition occurring. If the trigger point is less than 100 then at least one record entry needs to occur after the trigger point."  
<Supplanted by> ISO26048-1-Recording.fDRecordingFactoryPreSamples  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.8"  
DEFVAL { 80 }  
::= { recConfigEntry 8 }

#### 14.2.7.9 Recording Configuration Sample Period Parameter

recConfigSamplePeriod OBJECT-TYPE  
SYNTAX Integer32 (0..65535)  
UNITS "0.1 milliseconds"  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
"<Definition> This object contains the sample period for recordings collected at specified by this configuration. The sample period is expressed in units of 0.1 milliseconds. Allowable sample periods are restricted to a value of zero (0) or  $(\text{recMinSamplePeriod} + \text{recSamplePeriodResolution} * n)$  where n is integer,  $0 \leq n$ , and  $n \leq (\text{recMaxSamplePeriod} - \text{recMinSamplePeriod}) / \text{recSamplePeriodResolution}$ . If the value is zero (0), then the samples are not collected on a periodic basis, but rather a new sample is collected whenever the value of the object specified by recConfigSampleOID changes (i.e. similar to event log 'on-change' mode). A set to any other value results in a badValue response."  
<Supplanted by> ISO26048-1-Recording.fDRecordingFactorySamplePeriod  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.9"  
DEFVAL { 1000 }  
::= { recConfigEntry 9 }

#### 14.2.7.10 Recording Configuration Sample OID Parameter

recConfigSampleOID OBJECT-TYPE  
SYNTAX OBJECT IDENTIFIER  
MAX-ACCESS read-write  
STATUS deprecated  
DESCRIPTION  
"<Definition> This object contains the object identifier which references the value against which the 'on-change' comparison is made. If recConfigSamplePeriod is non-zero, then the value of this object shall be ignored. As with all other objects that are sub-ranged by a given implementation, an agent should return a badValue error if it receives a

set command indicating a OID which is not supported by the implementation or which is not zeroDotZero.

<Supplanted by> ISO26048-1-Recording.fDRecordingFactoryMonitorOID

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.10"

DEFVAL { zeroDotZero }  
::= { recConfigEntry 10 }

#### 14.2.7.11 Recording Configuration Number Entries Parameter

recConfigNumEntries OBJECT-TYPE

SYNTAX Integer32 (0..4095)

UNITS "records"

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition> This object contains the maximum number of records in a recording defined by this configuration. A recording which collects its full amount of pre-events and post events creates a recording of this number of entries. If this object is zero (0), then no recordings are created based on this configuration.

<Supplanted by> ISO26048-1-Recording.fDRecordingFactorySampleLimit

<Informative> If one wants to use block objects to retrieve a recording, then one should consider that block starting index value is limited to the range 00..255, and the number of entries above 255 is dependent on the size of the recEntry and packet size limitations.

The replacement object has a syntax of ITSUnsigned16.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.11"

DEFVAL { 100 }  
::= { recConfigEntry 11 }

#### 14.2.7.12 Recording Configuration Action Parameter

recConfigAction OBJECT-TYPE

SYNTAX INTEGER { other (1),  
disabled (2),  
record (3) }

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>The value of this object indicates what action shall take place when this configuration is triggered.  
other - indicates that the action is other than defined in this standard.  
disabled - no recording is created due to this configuration.  
record - a recording is created in the recording table when this configuration is triggered.

<Supplanted by> ISO26048-1-Recording.fDRecordingFactoryRowStatus

<Informative> The replacement table is dynamic.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.12"

DEFVAL { disabled }  
::= { recConfigEntry 12 }

#### 14.2.7.13 Recording Configuration Status Parameter

recConfigStatus OBJECT-TYPE

```
SYNTAX      INTEGER { other (1),
                    disabled (2),
                    record (3),
                    error (4) }
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition>The value of this object indicates the current status of the
  configured recording. Upon setting any object in this row of the
  recConfigTable, the agent determines if the setting is valid, and sets this
  object to one of the following states:
  other indicates that the action is successfully set to a mode other than
  that defined in this standard
  disabled indicates that the action is set to disabled
  record indicates that the action is successfully set to the record state
  after passing consistency checks.
  error indicates that the requested action could not be implemented due to a
  consistency check

  <Supplanted by> ISO26048-1-Recording.fdRecordingFactoryRowStatus
  <Informative> The replacement table is dynamic.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.13"
 ::= { recConfigEntry 13 }
```

#### 14.2.8 Maximum Recordings Parameter

```
maxRecordings OBJECT-TYPE
SYNTAX      Integer32 (1..65534)
UNITS       "records"
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
  "<Definition>The maximum, fixed number of rows that can be used within the
  recRecordingTable.

  <Informative> The replacement for recRecordingTable is not associated with
  an explicit row limitation just a size limitation given by
  recMechV2ClassTableSizeLimit and adminRecMechSizeLimit.

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.8"
 ::= { recMech 8 }
```

#### 14.2.9 Recording Table

```
recRecordingTable OBJECT-TYPE
SYNTAX      SEQUENCE OF RecRecordingEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
  "<Definition>A table containing information about Recordings both completed
  and in process. A request for an object from a row that has not been
  instantiated or has been cleared shall return a noSuchName error.

  <Table Type> dynamic

  <Supplanted by> ISO26048-1-Recording.fdRecordingRecordingTable

  <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9"
 ::= { recMech 9 }
```

```
recRecordingEntry OBJECT-TYPE
SYNTAX      RecRecordingEntry
MAX-ACCESS  not-accessible
STATUS      deprecated
DESCRIPTION
    "<Definition>This object defines an entry in the recording table
    <Supplanted by> ISO26048-1-Recording.fldRecordingRecordingEntry
    <Informative> The index for the replacement table has a preceding owner
    column. It also has a delete column that allows for the deletion of the
    recording, including all of its samples.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1"
INDEX       { recordingClass, recordingNumber }
 ::= { recRecordingTable 1 }

RecRecordingEntry ::= SEQUENCE {
    recordingClass      Integer32,
    recordingNumber     Integer32,
    recordingID         Integer32,
    recordingConfigID   Integer32,
    recordingTriggerTime OCTET STRING,
    recordingStatus     INTEGER,
    recordingTriggerRecNum Integer32,
    recordingNumEntries Integer32}

```

#### 14.2.9.1 Recording Class Parameter

```
recordingClass OBJECT-TYPE
SYNTAX      Integer32 (1..254)
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
    "<Definition>This object contains the class of the associated recording as
    defined in the recConfigTable.
    <Supplanted by> ISO26048-1-Recording.fldRecordingClassName
    <Informative> The syntax of the replacement object is SnmpAdminString.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.1"
 ::= { recRecordingEntry 1 }

```

#### 14.2.9.2 Recording Number Parameter

```
recordingNumber OBJECT-TYPE
SYNTAX      Integer32 (1..254)
MAX-ACCESS  read-only
STATUS      deprecated
DESCRIPTION
    "<Definition>The recording number within this class for this recording.
    Recording numbers shall be assigned starting at 1 and shall increase to the
    value specified by the associated recClassLimit for the class associated
    with the rows. Recordings shall maintain a chronological ordering in the
    table, based on their recordingTriggerTime value, with the oldest recording
    of a class occupying the row with recordingNumber = 1, and subsequent
    recordings filling subsequent rows. This ordering shall be maintained for
    those rows still remaining when recordings are cleared.

```

<Supplanted by> ISO26048-1-Recording.fRecordingRecordingIndex

<Informative> The syntax of the replacement object is ITSPositive16. The replacement table does not attempt to keep the oldest entry in slot one; clearance of rows is based on overall size rather than number of entries, which makes this type of row management more difficult - but since row numbers do not change, managers can keep track of the last row number retrieved and decrease the probability of missing a recording. A manager can also use a get-next operation to discover where the rows start in the table.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.2"

::= { recRecordingEntry 2 }

#### 14.2.9.3 Recording ID Parameter

recordingID OBJECT-TYPE

SYNTAX Integer32 (1..65534)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>This object contains the recording ID that is used as an index into recording entries table (recEntriesTable) when accessing record entries that belong to this recording. The recording ID is assigned to a recording upon creation of its first record entry and does not change throughout the life of the recording. The recording IDs are not necessarily assigned sequentially.

<Informative> The replacement for the recEntriesTable uses the four indices owner, class name, recording ID, and sample number rather than using a single identifier to join class name and recording.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.3"

::= { recRecordingEntry 3 }

#### 14.2.9.4 Recording Configuration ID Parameter

recordingConfigID OBJECT-TYPE

SYNTAX Integer32 (1..65535)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>This object contains the recording configuration ID (from the recConfigTable) that caused this table entry. It indicates the row in the recConfig table responsible for this recording entry.

<Supplanted by> ISO26048-1-Recording.fRecordingRecordingTrigger

<Informative> The syntax of the replacement object is SnmpAdminString.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.4"

::= { recRecordingEntry 4 }

#### 14.2.9.5 Recording Trigger Time Parameter

recordingTriggerTime OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition> The time that the recording was triggered at. This object consists of a string of six (6) octets. The first four (4) octets reflect

the value of globalTime when the recording was triggered. The last two (2) octets reflect the time in milliseconds the trigger occurred after the start of the second. The recording shall be detected and timestamped within one recSamplePeriodResolution unit of time from the recording being triggered.

<Supplanted by> ISO26048-1-Recording.fidRecordingRecordingDate & recMechV2RecordingTime

<Informative> The replacement objects conform to the new date/time formats.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.5"

::= { recRecordingEntry 5 }

#### 14.2.9.6 Recording Status Parameter

recordingStatus OBJECT-TYPE

SYNTAX INTEGER { available (1),  
preevent (2),  
triggered (3),  
complete (4) }

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The value of this object reflects the state of the recording located in this row.

<Format>

Value	Description
available	this row is available for a new recording to be initiated for this recording class. When a recording is cleared, the value is set to available
preevent	the recording defined by recConfigID is collecting pre-event record entries. Note: If the trigger point is set to zero (0) percent or the trigger condition is
already	satisfied when the recConfigID is configured, then the value transitions straight from available to triggered.
triggered	the recording defined by recConfigID has triggered and is now collecting post event records. If the trigger point
was	set to 100 percent, then the value transitions straight
from	preevent to complete. If a recording was triggered and the device experienced a power failure, then upon power restoration it shall change the value to complete. Triggered recordings shall survive a power outage.
complete	the recording defined by recConfigID is now complete (i.e. collected all of its post events) and ready for retrieval.

Completed recordings shall survive a power outage.

<Supplanted by> ISO26048-1-Recording.fDRecordingRecordingStatus  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.6"

::= { recRecordingEntry 6 }

#### 14.2.9.7 Recording Trigger Record Number Parameter

recordingTriggerRecNum OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>This object contains the record entry number of the trigger record entry. A value of zero (0) means that the recording has not triggered yet.

<Supplanted by> ISO26048-1-Recording.fDRecordingRecordingTriggerSample

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.7"

::= { recRecordingEntry 7 }

#### 14.2.9.8 Number of Entries in Recording Parameter

recordingNumEntries OBJECT-TYPE

SYNTAX Integer32 (0..65535)

UNITS "entries"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The current number of recording entries in this recording. A value of zero (0) is only valid if the recording is available for use.

<Informative> A completed recording does not always have recConfigNumEntries record entries in it.

<Supplanted by> ISO26048-1-Recording.fDRecordingRecordingNumSamples

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.8"

::= { recRecordingEntry 8 }

#### 14.2.10 Maximum Recording Entries Parameter

maxRecEntries OBJECT-TYPE

SYNTAX Integer32 (1..65535)

UNITS "records"

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"<Definition>The maximum, fixed number of rows that can be used within the recEntriesTable.

<Informative> The replacement for recEntriesTable is not associated with an explicit row limitation just a size limitation given by recMechV2ClassTableSizeLimit and adminRecMechSizeLimit.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.10"

::= { recMech 10 }



#### 14.2.11 Recording EntriesTable

recEntriesTable OBJECT-TYPE  
SYNTAX SEQUENCE OF RecEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition>A table containing the discrete Recording entry records. A request for an object from a row that has not been instantiated or has been cleared shall return a noSuchName error.  
  
<Table Type> dynamic  
  
<Supplanted by> ISO26048-1-Recording.fldRecordingSampleTable  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.11"  
 ::= { recMech 11 }

recEntry OBJECT-TYPE  
SYNTAX RecEntry  
MAX-ACCESS not-accessible  
STATUS deprecated  
DESCRIPTION  
" <Definition>This object defines an entry in the recording entry table. All entries within a recording shall be ordered chronologically from oldest to newest.  
  
<Supplanted by> ISO26048-1-Recording.fldRecordingSampleEntry  
  
<Informative> The index for the replacement table uses four fields: owner, class, recording, and sample.  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.11.1"  
INDEX { recordingID, recEntryNumber }  
 ::= { recEntriesTable 1 }

RecEntry ::= SEQUENCE {  
    recEntryNumber Integer32,  
    recSampleTime OCTET STRING,  
    recValue Opaque}

##### 14.2.11.1 Record Entry Number Parameter

recEntryNumber OBJECT-TYPE  
SYNTAX Integer32 (0..65535)  
MAX-ACCESS read-only  
STATUS deprecated  
DESCRIPTION  
" <Definition>The entry number within this recording for this record. Entry numbers shall be assigned starting at 1 and shall increase up to the value specified by the associated recConfigNumEntries. A value of zero indicates that the row is unused (cleared).  
  
<Supplanted by> ISO26048-1-Recording.fldRecordingSampleNum  
  
<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.11.1.1"  
 ::= { recEntry 1 }

##### 14.2.11.2 Record Entry Sample Time Parameter

recSampleTime OBJECT-TYPE

SYNTAX OCTET STRING  
MAX-ACCESS read-only  
STATUS deprecated

DESCRIPTION

"<Definition>The time that the record entry was sampled at. This object consists of a string of six (6) octets. The first four (4) octets reflect the value of controllerLocalTime when the entry was sampled. The last two (2) octets reflect the time in milliseconds the sample occurred after the start of the second. The entry shall be collected within one recSamplePeriodResolution unit of time from the sample being triggered and timestamped with the time of collection

<Supplanted by> ISO26048-1-Recording.fidRecordingSampleTime

<Informative> The syntax for the replacement object is ITSDailyTimeStamp, which does not provide the date. However, the date is provided in the Trigger time. Assuming the sample period is relatively short, this should not cause a problem. For recordings with many samples and/or long sample intervals, the date can be calculated - with the one exception of on change sampling for rare events. owner, class, recording, and sample.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.11.1.2"

::= { recEntry 2 }

#### 14.2.11.3 Record Entry Value Parameter

recValue OBJECT-TYPE  
SYNTAX Opaque  
MAX-ACCESS read-only  
STATUS deprecated

DESCRIPTION

"<Definition>The value of this object is set to the BER encoding of the value referenced by the recConfigRecordOID of the associated recordingConfigID when the entry was collected. Its length is variable. The value shall not contain any padding characters either before or after the values.

<Supplanted by> ISO26048-1-Recording.fidRecordingSampleValue

<Informative> Opaque objects are doubly wrapped. For SNMP operations, which use BER, this would be {type, length, {type, length, value}}. For example, a zero-length octet string, would be encoded in BER as 0x44 02 04 00. For STMP or SFMP operations, which use OER, this would be { length, {type, length, value}}. For example, the same example would be encoded in OER as 0x02 04 00.

The syntax of the replacement object is ITSOerString.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.11.1.3"

::= { recEntry 3 }

#### 14.2.12 Total Recordings Counter Parameter

numRecordings OBJECT-TYPE  
SYNTAX Integer32 (0..65535)  
UNITS "recordings"  
MAX-ACCESS read-only  
STATUS deprecated

DESCRIPTION

"<Definition> This object is a counter that gets incremented every time a recording is completed and shall initialize to zero at power up. The value shall roll over each time it exceeds the maximum of 65535.

<Supplanted by> ISO26048-1-Recording.fRecordingOwnerRecordingCtr & adminRecMechV2RecordingCtr

<Informative> The syntax of the replacement objects is ZeroBasedCounter32.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.12"

::= { recMech 12 }

#### 14.2.13 Clear Recording Classes

recClearClasses OBJECT-TYPE

SYNTAX Integer32 (0..255)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to clear a recording class from recClassTable. A SET of zero has no effect on any recording classes. A SET = n, n <= maxRecClasses, shall cause recClassNumber = n to be deleted from the recClassTable and all related recording configurations, their recordings, their recording entries, their recording activities shall be deleted from the recConfigTable, recRecordingTable, and recEntriesTable respectively. A SET of 255 shall cause all entries in the recClassTable, recConfigTable, recRecordingTable, and recEntriesTable to be deleted. Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).

<Supplanted by> ISO26048-1-Recording.fRecordingClassRowStatus, recMechV2OwnerClearAllClasses, & adminRecMechClearAllClasses

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.13"

::= { recMech 13 }

#### 14.2.14 Clear Recording Configurations

recClearConfigurations OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-write

STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to clear a recording configuration from the recConfigTable. A SET of zero has no effect on any recording configurations. A SET = n, n <= maxRecConfigs, shall cause recConfigID = n to be deleted from the recRecordingTable and all related recordings, their recording entries, and their recording activities shall be deleted from the recRecordingTable and recEntriesTable respectively. A SET of n = 65535 shall cause all entries in the recConfigTable, recRecordingTable, and recEntriesTable to be deleted. Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).

<Supplanted by> ISO26048-1-Recording.fRecordingFactoryRowStatus, recMechV2OwnerClearAllFactories, adminRecMechClearAllFactories, COND\_NOTIFICATIONS-MIB.fCondTriggerRowStatus (ISO 26048-1)

<Informative> There is not a clear all for the conditional trigger.

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.14"

::= { recMech 14 }

#### 14.2.15 Clear Recording Data

recClearRecordingData OBJECT-TYPE

SYNTAX Integer32 (0..65535)  
MAX-ACCESS read-write  
STATUS deprecated

DESCRIPTION

"<Definition>This object commands the device to clear a recording from the recRecordingTable and recEntriesTable. A SET of zero has no effect on any recordings. A SET = n, n <= maxRecordings, shall cause recRecordingID = n to be deleted from the recRecordingTable and all record entries related to the recording to be deleted from recEntriesTable. A SET of n = 65535 shall cause all entries in both the recRecordingTable and recEntriesTable to be deleted.

Upon performing the action requested, the device shall SET this object to zero (0). A GET shall always return zero (0).

<Informative> A device shall immediately start collecting new pre-event records for all active recording configurations.

<Supplanted by> ISO26048-1-Recording.fRecordingRecordingDelete, recMechV2OwnerClearAllRecordings, & adminRecMechClearAllRecordings

<Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.15"

::= { recMech 15 }  
END

## Annex A OBJECT TREE AND UML CLASS DIAGRAMS [INFORMATIVE] [DEPRECATED]

Annex A provides an overview of NTCIP 1201 object types by presenting relevant branches of the object identifier naming tree and by presenting the object types in UML Class Diagrams. The information presented in Annex A is defined elsewhere in this document; however, these figures provide a high-level overview of the data contained in this document in a concise manner and are provided as a useful reference tool.

### A.1 OBJECT IDENTIFIER NAMING TREE

Figure 2 depicts the location of nodes representing the MODULE-IDENTITY macros contained within this document.

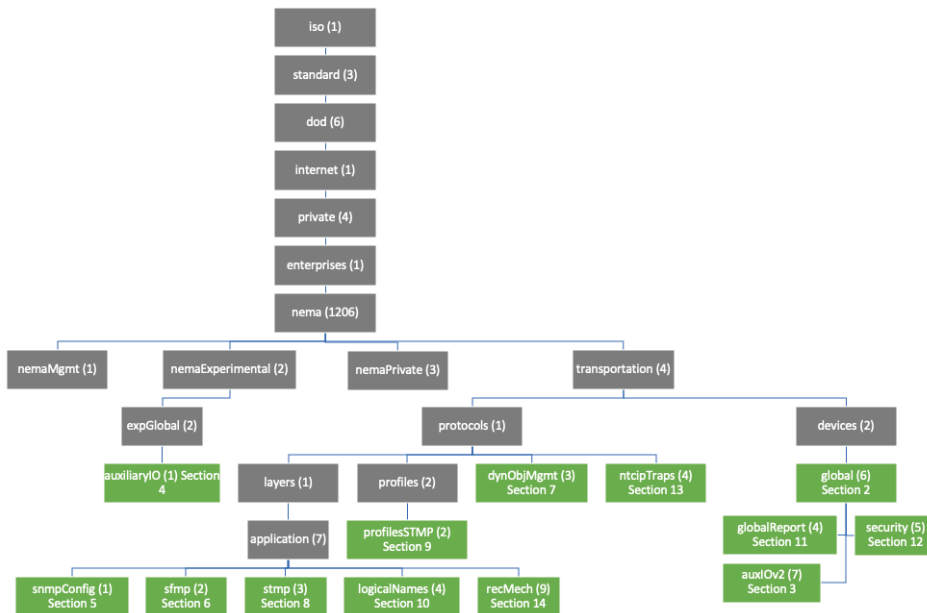


Figure 2: MODULE-IDENTITY nodes

Figure 3 through Figure 18 extend the tree shown in Figure 2 by showing the relevant nodes defined in each of the MIBs contained in Section 2 through Section 14 .

In each diagram, gray boxes represent simple nodes, green boxes represent MIBs (i.e., defined with MODULE-IDENTITY), and boxes shown in blue represent object types. An ellipsis (...) represents a repetition of the initial characters for the given context. For example, within the recMech MIB, the ellipsis either represents the text "recMech" or, when occurring within a specific table, the root of the table name (e.g., "recMechClass").

NOTE — The object types within a MIB typically appear under the MODULE-IDENTITY node but can include objects located in other parts of the tree. This is especially true for administrative objects so that appropriate data access can be more easily configured.

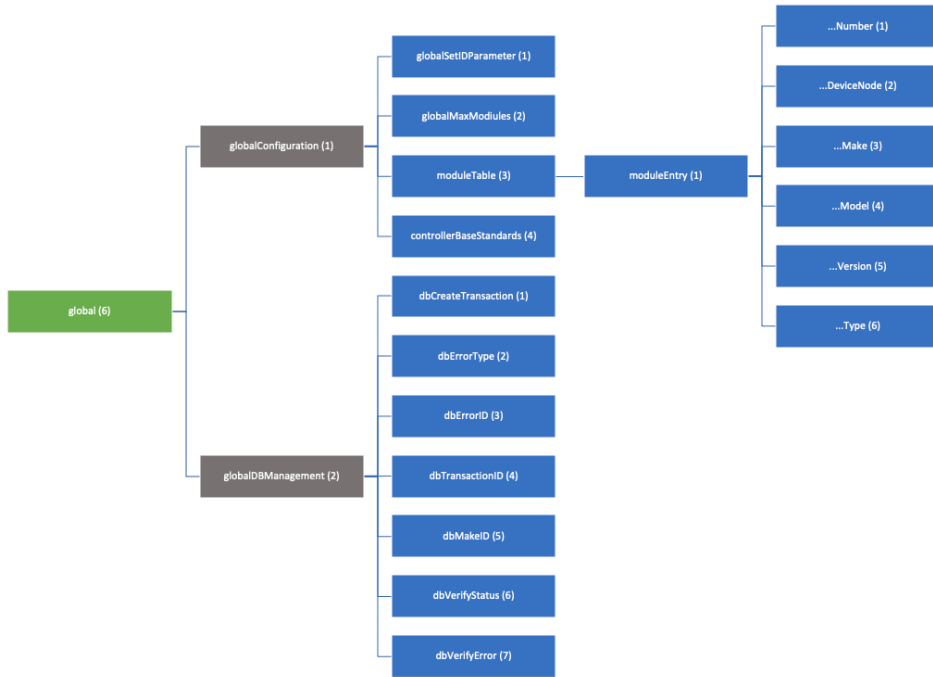


Figure 3: Nodes Defined in NTCIP 1201-Global MIB (Part 1 of 3: Base Objects)



Figure 4: Nodes Defined in NTCIP 1201-Global MIB (Part 2 of 3: Time Management)

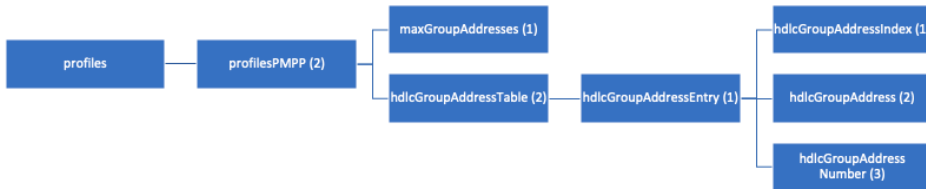


Figure 5: Nodes Defined in NTCIP 1201-Global MIB (Part 3 of 3: PMPP Objects)

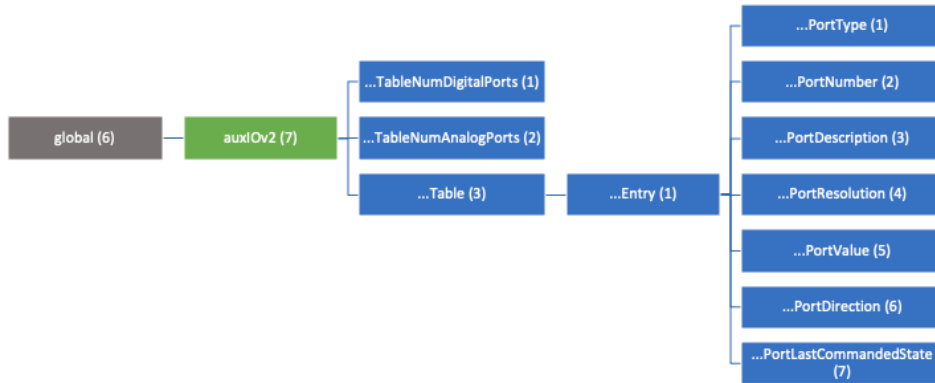


Figure 6: Nodes Defined in NTCIP1201-AuxIOv2 MIB

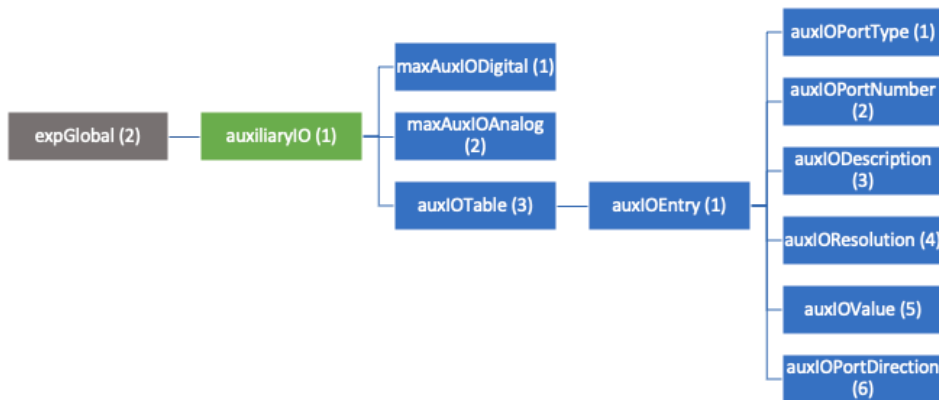


Figure 7: Nodes Defined in NTCIP1201-AuxIO MIB



Figure 8: Nodes Defined in NTCIP1201-SNMPConfig MIB





Figure 9: Nodes Defined in NTCIP1201-SFMP MIB

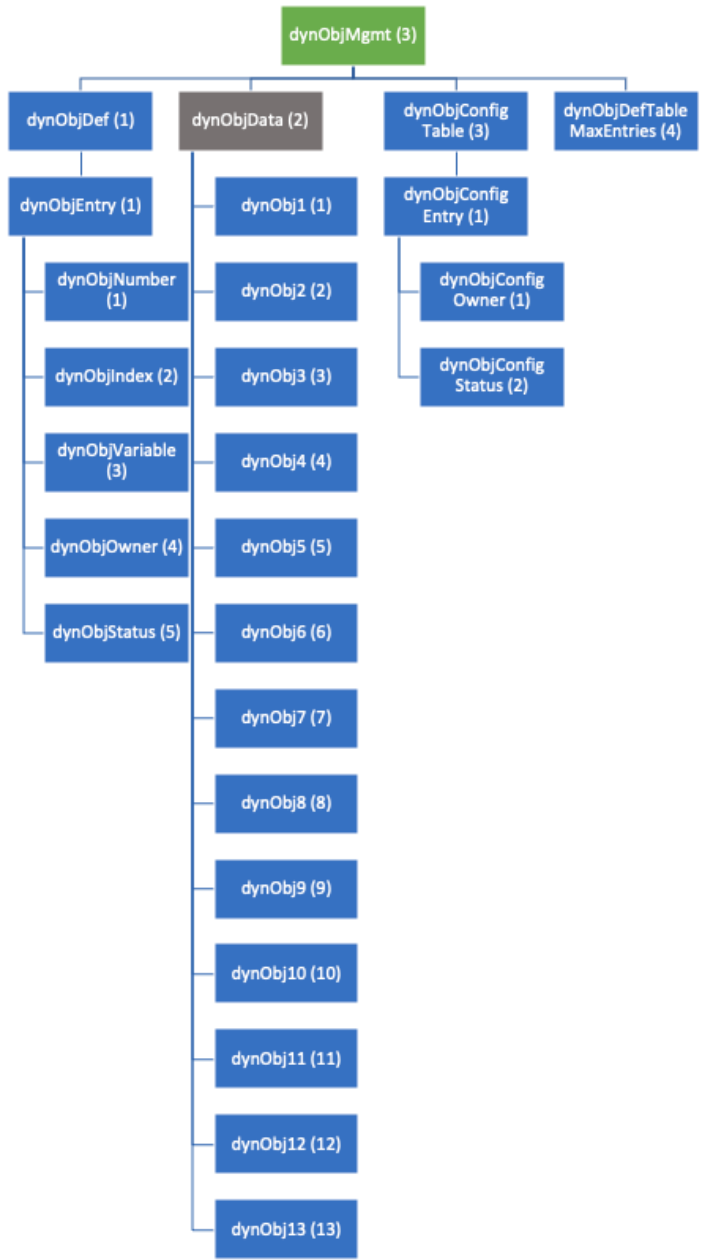


Figure 10: Nodes Defined in NTCIP1201-DynObjMgmt MIB



Figure 11: Nodes Under NTCIP1201-StmpStatistics MIB

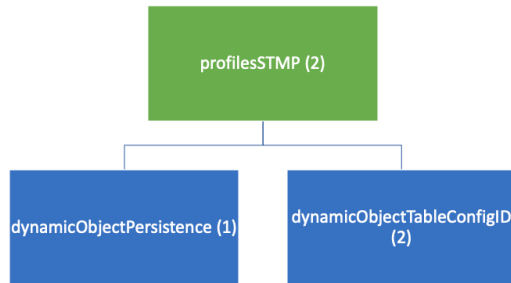


Figure 12: Nodes Under NTCIP1201-ProfilesSTMP MIB

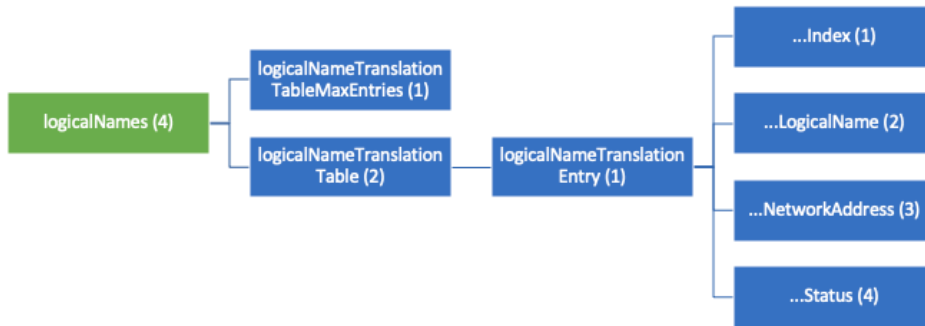
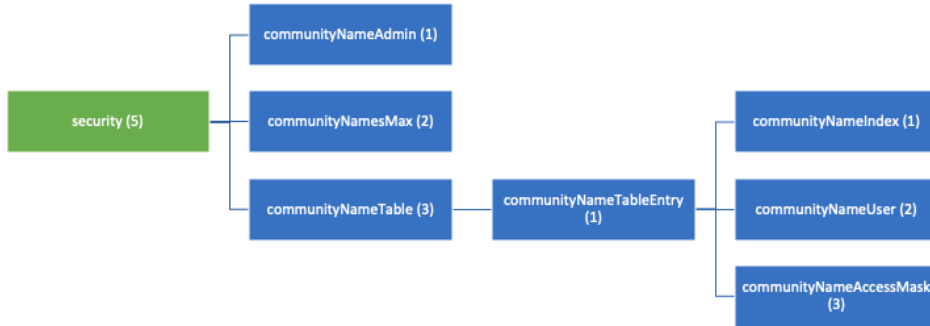


Figure 13: Nodes under NTCIP1201-LogicalNames MIB



Figure 14: Nodes Defined in NTCIP1201-GlobalReport MIB



**Figure 15: Nodes Defined in NTCIP1201-Security MIB**

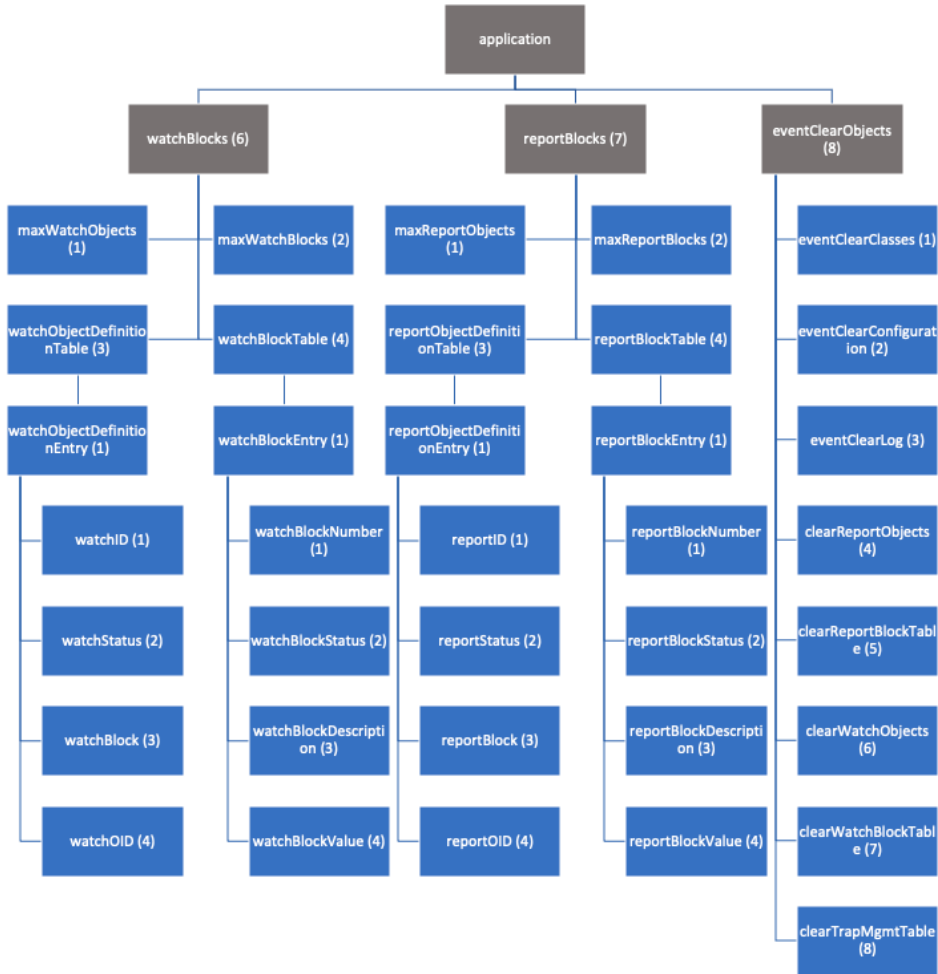


Figure 16: Nodes Defined in NTCIP1201-NtcipTraps MIB (Part 1 of 2: Application Objects)

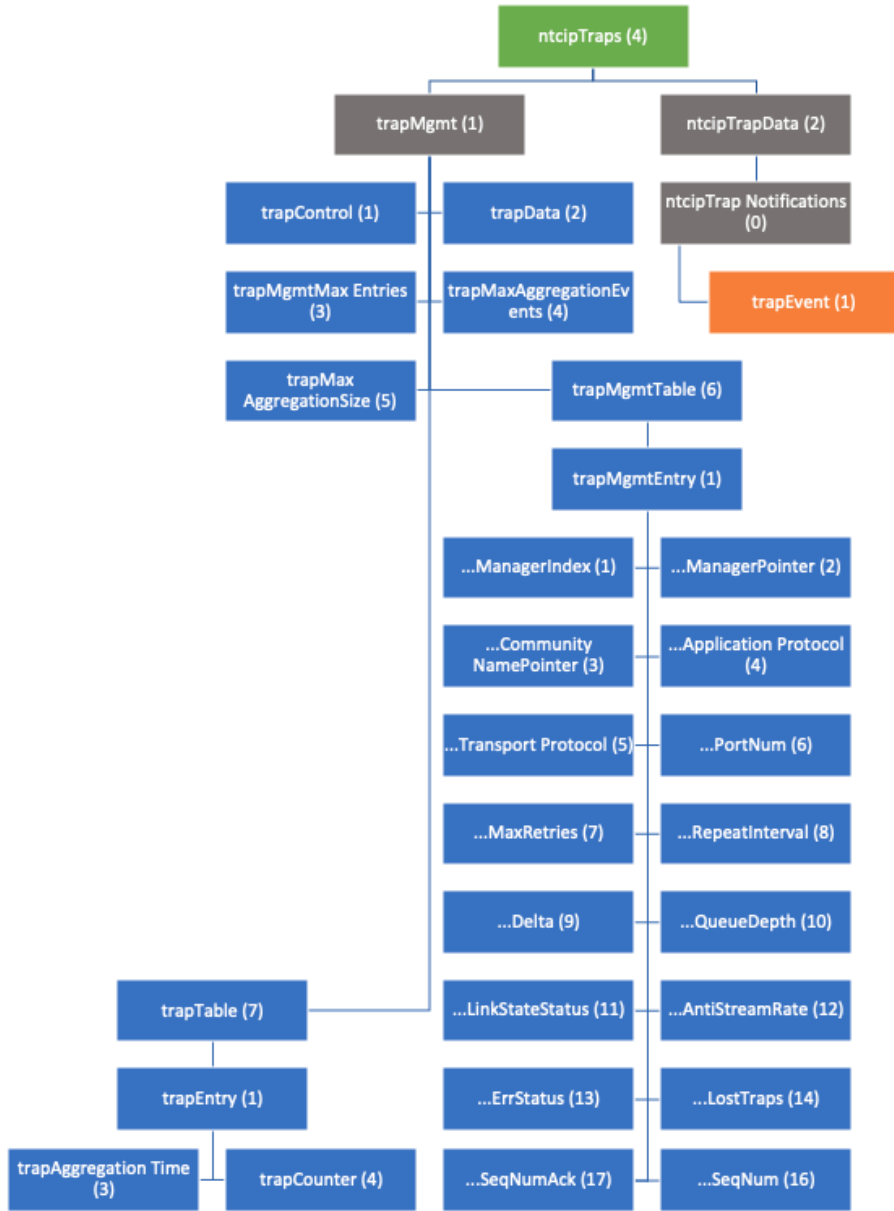


Figure 17: Nodes Defined in NTCIP1201-NtcipTraps MIB (Part 2 of 2: Trap Management)



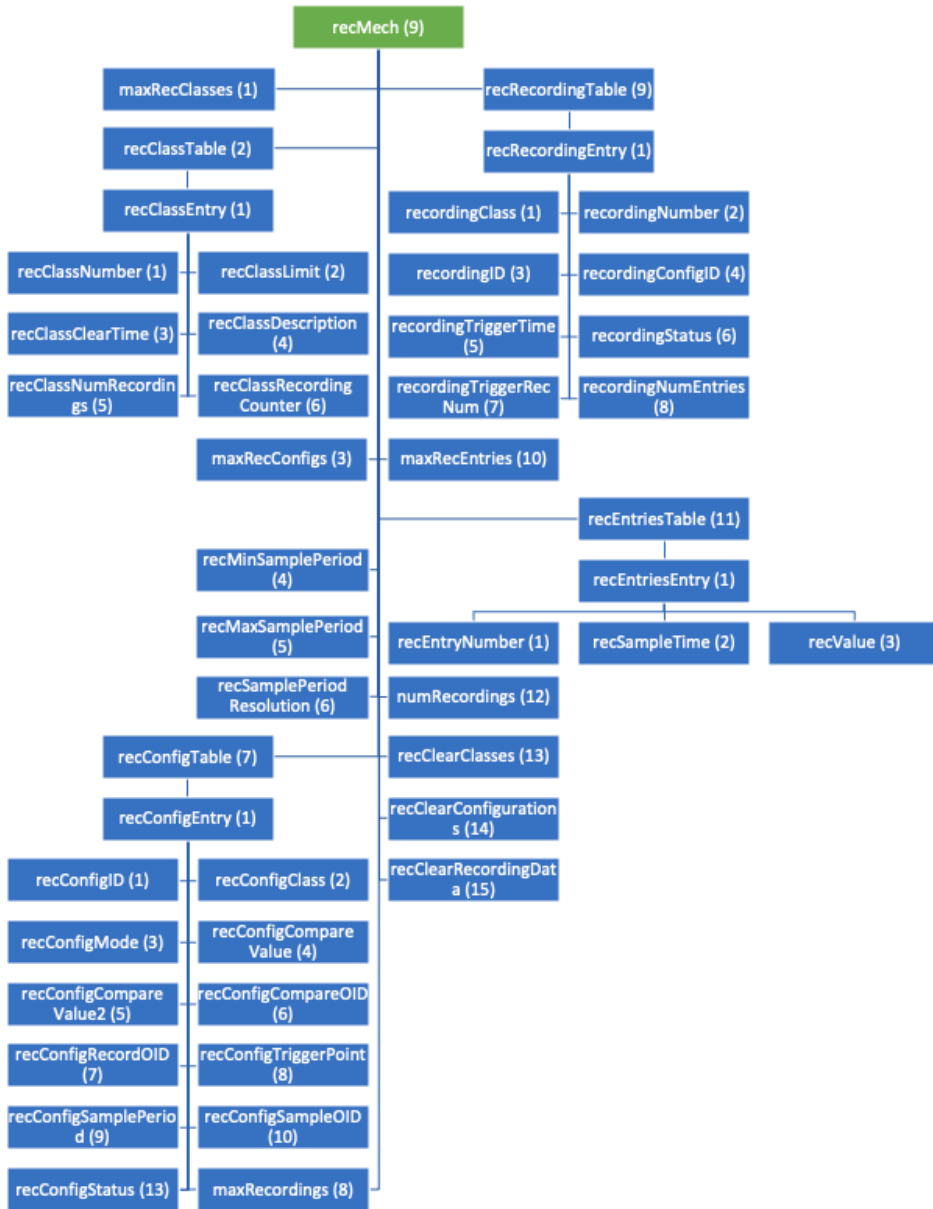


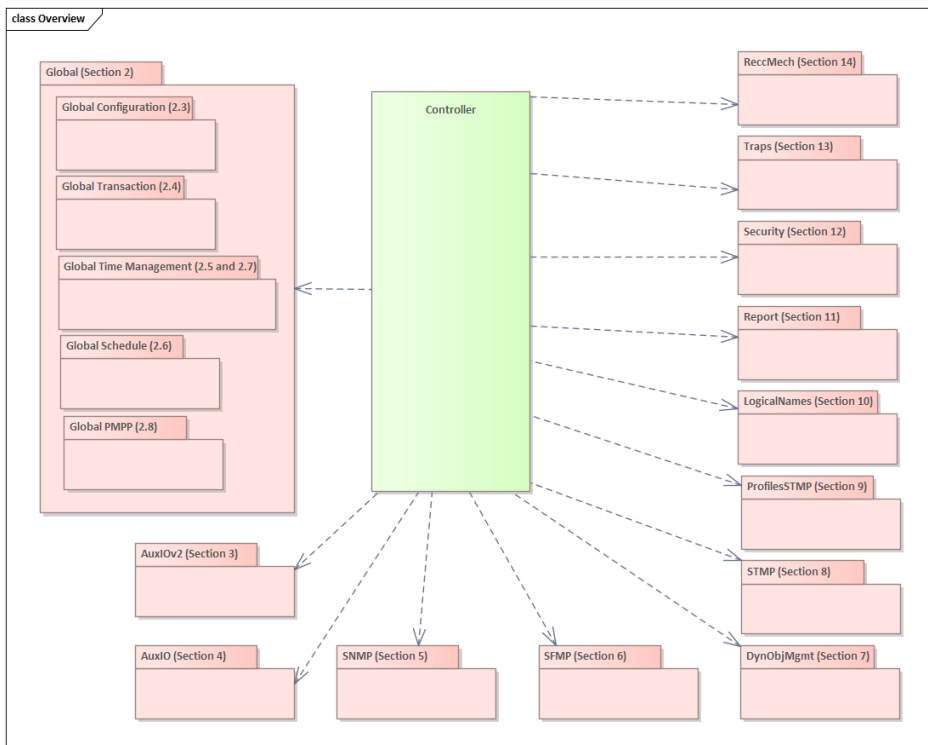
Figure 18: Nodes Defined in NTCIP1201-RecMech MIB

## A.2 UML CLASS DIAGRAMS

Although object types defined within MIBs are not formally structured into class models, they can easily be presented in UML class diagrams by mapping each object type to an attribute of a defined class. through provide UML class diagrams and associated mapping tables to depict how the object types defined in this document relate to one another and to objects defined in other standards.

### A.2.1 Controller Class Diagram

Figure 19 depicts the high-level components of data stored within a controller in UML notation.



**Figure 19: Controller Class Diagram**

Figure 23 indicates that a Controller can include the following major components, which correspond to the sections of this document:

1. Global features, which include:
  - a. Global configuration
  - b. Global transaction
  - c. Global time management
  - d. Global schedule
  - e. Global PMPP data
2. auxiliary input/output version 2
3. auxiliary input/output (version 1)
4. SNMP data

5. SFMP data
6. dynamic object management
7. STMP data
8. STMP profile data
9. Logical names
10. Event reporting
11. Security
12. Traps
13. Recording mechanism

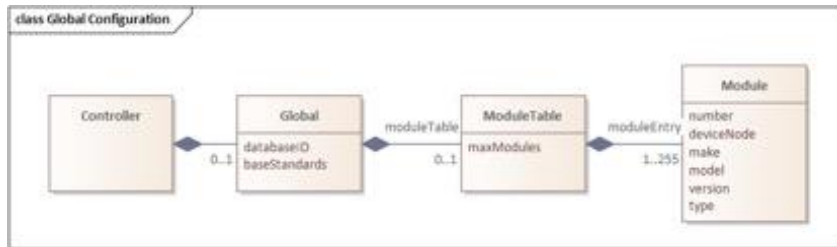
All parts of this document have been deprecated.

NOTE — Two different versions of the Aux I/O object definitions are currently defined. The first was defined under an experimental node of the global tree and was originally contained in NTCIP 1203 v01. Because of the applicability of the Aux I/O objects to more than dynamic message signs (DMS), these objects were moved to NTCIP 1201 v02, and moved from the experimental node to a permanent node under the global tree. To differentiate these two sets of objects, the objects associated with Aux I/O in NTCIP 1201 v02 have had 'v2' added in the object name. They are both now deprecated.

More detailed class diagrams for each feature are provided in Figure 20 through Figure 36 .

### A.2.2 Configuration Information

Figure 20 depicts the configuration data stored by a controller.



**Figure 20: Class Diagram of the Configuration Information**

A controller may have a database identifier, an indication of the standards that it supports, and zero or one module tables. If there is a module table, then the controller may additionally support an object defining the maximum number of modules supported within the table, which may be between one and 255, as indicated by the link to the Module class. For each module, the controller may support a variety of information, including:

1. the module number;
2. the device node to which the module relates;
3. the make of the module;
4. the model of the module;
5. the version of the module; and
6. the type of module.

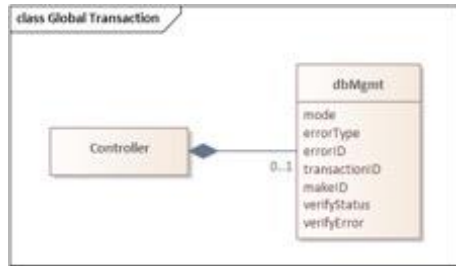
The mapping of each attribute in this class to object types are defined in Table 1.

**Table 1: Configuration Information Mapping to Class Diagram**

Property	Object Type
Global.databaseID	globalSetIDParameter
Global.baseStandards	controllerBaseStandards
ModuleTable.maxModules	globalMaxModules
Module.number	moduleNumber
Module.deviceNode	moduleDeviceNode
Module.make	moduleMake
Module.model	moduleModel
Module.version	modelVersion
Module.type	moduleType

**A.2.3 Transaction Information**

Figure 21 depicts the transaction state data stored by a controller.



**Figure 21: Class Diagram of the Transaction Service**

A controller may support a transaction feature. The following information characterizes the feature as defined in NTCIP 1103 v03:

1. a mode;
2. a verify status; and
3. a verification error message.

The errorType, errorID, transactionID, and makeID objects might also exist, but they were deprecated in previous versions of the MIB.

The mapping of each attribute in this class to object types are defined in Table 2.

**Table 2: Transaction Service Mapping to Class Diagram**

Property	Object Type
DbMgmt.mode	dbCreateTransaction
DbMgmt.errorType	dbErrorType
DbMgmt.errorID	dbErrorID
DbMgmt.transactionID	dbTransactionID
DbMgmt.makeID	dbMakeID
DbMgmt.verifyStatus	dbVerifyStatus
DbMgmt.verifyError	dbVerifyError

#### A.2.4 Time and Daylight Saving Time (DST) Information

Figure 22 depicts the time-related and DST-related data stored by a controller.

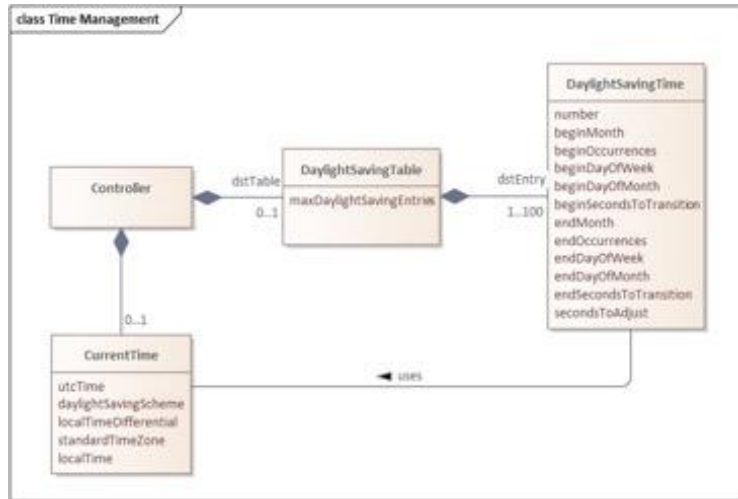


Figure 22: Class Diagram of Time/DST Information

A controller may store time-related information, including:

1. the current time in UTC
2. an indication of the daylight saving mode
3. an indication of the time zone when in standard time
4. an indication of the local time, which includes and accounts for DST

The controller might also support a local time differential from UTC, although this object was previously deprecated.

The controller may also support a DST Table. If this is supported, it is characterized by the maximum number of entries that it may contain, which is required to be at least one and may be no greater than 100. For each entry, the following information may be stored:

1. a DST number
2. a begin DST month indicating in which month the DST may begin
3. a begin DST occurrences parameter indicating the number of occurrences of the specific day of week required to have occurred within the selected month before DST begins [NOTE—"beginOccurrences" (sic) is misspelled in the figure, but is spelled correctly in the MIB object definitions.]
4. a begin DST day of week indicating on which day of week the DST may begin
5. a begin DST day of month indicating on which day of month the DST may begin
6. a begin DST Seconds to Transition parameter indicating after how many seconds after midnight of a particular day the DST may begin
7. an end DST month indicating in which month the DST may begin
8. an end DST occurrences parameter indicating number of occurrences of the specific day of week required to have occurred within the selected month before DST ends
9. an end DST day of week indicating on which day of week the DST may end

- 10. an end DST day of month indicating on which day of month the DST may end
- 11. an end DST Seconds to Transition parameter indicating after how many seconds after midnight of a particular day the DST may end
- 12. a seconds to adjust parameter indicating by how many seconds the DST time is offset from the local reference time when DST as defined by this entry is in effect

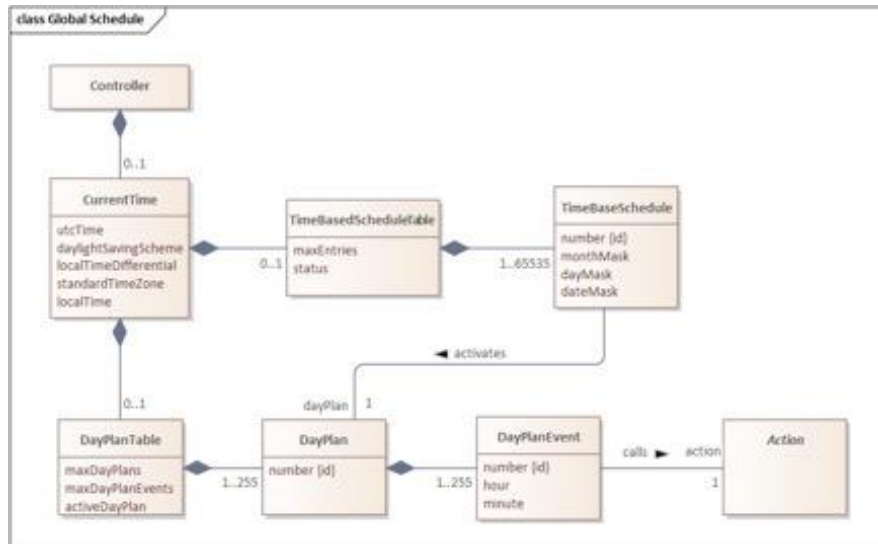
The mapping of each attribute in this class to object types are defined in Table 3.

**Table 3: Time/DST Information Mapping to Class Diagram**

Property	Object Type
CurrentTime.utcTime	globalTime
CurrentTime.daylightSavingScheme	globalDaylightSaving
CurrentTime.localTimeDifferential	globalLocalTimeDifferential
CurrentTime.standardTimeZone	controllerStandardTimeZone
CurrentTime.localTime	controllerLocalTime
DaylightSavingTable.maxDaylightSavingEntries	maxDaylightSavingEntries
DaylightSavingTime.number	dstEntryNumber
DaylightSavingTime.beginMonth	dstBeginMonth
DaylightSavingTime.beginOccurrences	dstBeginOccurrences
DaylightSavingTime.beginDayOfWeek	dstBeginDayOfWeek
DaylightSavingTime.beginDayOfMonth	dstBeginDayOfMonth
DaylightSavingTime.beginSecondsToTransition	dstBeginSecondsToTransition
DaylightSavingTime.endMonth	dstEndMonth
DaylightSavingTime.endOccurrences	dstEndOccurrences
DaylightSavingTime.endDayOfWeek	dstEndDayOfWeek
DaylightSavingTime.endDayOfMonth	dstEndDayOfMonth
DaylightSavingTime.endSecondsToTransition	dstEndSecondsToTransition
DaylightSavingTime.secondsToAdjust	dstSecondsToAdjust

**A.2.5 Generic Schedule Information**

Figure 23 depicts the generic schedule-related data stored by a controller.



**Figure 23: Class Diagram of Generic Schedule-Related Information**

Figure 29 indicates a controller may store schedule information, including:

1. the current time in UTC
2. an indication of the time zone when in standard time
3. an indication of the local time, which includes and accounts for DST

The controller may also support a timebase schedule table. If supported, it is characterized by the maximum number of entries that it may contain, which is required to be at least one and may be no greater than 65535, and a status. For each entry, the following information may be stored:

1. a schedule number
2. a month mask indicating which months the schedule may be active
3. a day mask indicating which days of the week the schedule may be valid
4. a date mask indicating which dates of the month the schedule may be active
5. a link to a day plan record

To have a link to a day plan, the day plan is also required to be supported, which in turn requires that its container class and the day plan table are also required to be supported. The day plan table is characterized by:

1. the maximum number of day plans that may be stored, which must be between one and 255
2. the maximum number of events that may occur during a day, which must be between one and 255
3. an indication of the day plan that is currently active

The day plan itself only consists of the day plan number and a link to between one and 255 day plan events. Each day plan event is described by:

1. a number;
2. the hour during which the event occurs;
3. the minute during which the event occurs;

4. the status of the action; and
5. a link to the specific action to be performed

The specific action to be performed is defined elsewhere because of the device-specific nature of actions.

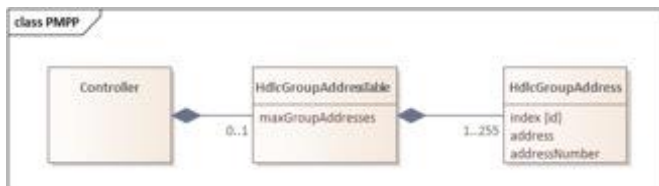
The mapping of each attribute in this class to object types are defined in Table 4.

**Table 4: Time Information Mapping to Class Diagram**

Property	Object Type
CurrentTime.utcTime	globalTime
CurrentTime.daylightSavingScheme	globalDaylightSaving
CurrentTime.localTimeDifferential	globalLocalTimeDifferential
CurrentTime.standardTimeZone	controllerStandardTimeZone
CurrentTime.localTime	controllerLocalTime
TimeBaseScheduleTable.maxEntries	maxTimeBaseScheduleEntries
TimeBaseScheduleTable.status	timeBaseScheduleTableStatus
TimeBaseSchedule.number	timeBaseScheduleNumber
TimeBaseSchedule.monthMask	timeBaseScheduleMonth
TimeBaseSchedule.dayMask	timeBaseScheduleDay
TimeBaseSchedule.dateMask	timeBaseScheduleDate
TimeBaseSchedule.dayPlan	timeBaseScheduleDayPlan
DayPlanTable.maxDayPlans	maxDayPlans
DayPlanTable.maxDayPlanEvents	maxDayPlanEvents
DayPlanTable.activeDayPlan	dayPlanStatus
DayPlan.number	dayPlanNumber
DayPlanEvent.number	dayPlanEventNumber
DayPlanEvent.hour	dayPlanHour
DayPlanEvent.minute	dayPlanMinute
DayPlanEvent.action	dayPlanActionNumberOID

### A.2.6 PMPP Information

Figure 24 depicts the database management version 2 data stored by a controller.



**Figure 24: Class Diagram of the PMPP Data**

A controller can support the HDLC Group Address Table with a defined number of group addresses and an entry in the table for each stored address. The mapping of each attribute in this class to object types are defined in Table 5.

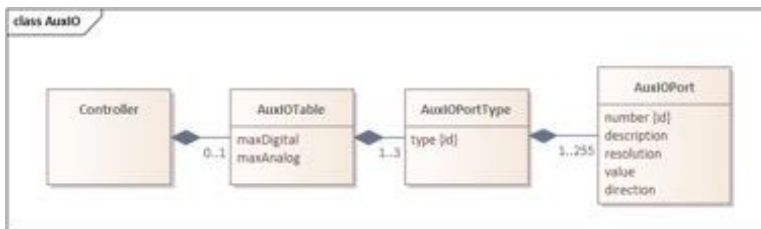


**Table 5: PMPP Mapping to Class Diagram**

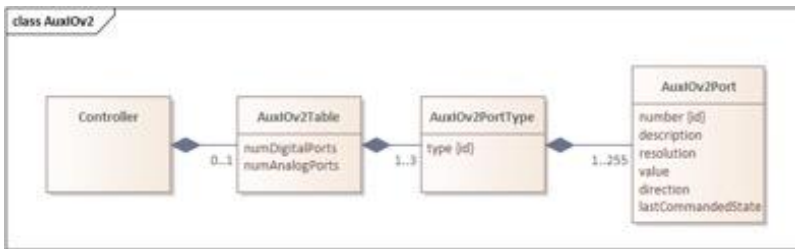
Property	Object Type
HdlcGroupAddressTable.maxGroupAddresses	maxGroupAddresses
HdlcGroupAddress.index	hdlcGroupAddressIndex
HdlcGroupAddress.address	hdlcGroupAddress
HdlcGroupAddress.addressNumber	hdlcGroupAddressNumber

**A.2.7 Auxiliary Input/Output Information**

Figure 25 and Figure 26 depict the auxiliary input/output data stored by a controller. Two diagrams are shown, one depicting the methods and object definitions originally defined in NTCIP 1203 v01 and the methods and objects defined in NTCIP 1201 v02.



**Figure 25: Class Diagram for Auxiliary Input/Output (NTCIP 1203 v01)**



**Figure 26: Class Diagram for Auxiliary Input/Output Version 2 (NTCIP 1201 v02)**

Figure 32 and Figure 33 indicate a controller may support an auxiliary input/output table (AuxIO2 and/or AuxIO). If either is supported, it is characterized by the maximum number of digital and analog ports supported by the device. Each port type is allocated to its own sub-table in the AuxIOPortType table, which contains multiple entries, one for each port, where each port is characterized by:

1. a number
2. a description
3. a resolution of the data supported by the port
4. a value
5. a direction
6. the last commanded state (only in NTCIP 1201 v02)

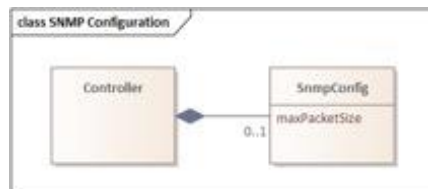
The mapping of each attribute in this class to object types are defined in Table 6.

**Table 6: Auxiliary Input and Output Mapping to Class Diagram**

Property	Object Type
AuxIOTable.maxDigitalPorts	maxAuxIODigital
AuxIOTable.maxAnalogPorts	maxAuxIOAnalog
AuxIOPortType.type	auxIOPortType
AuxIOPort.number	auxIOPortNumber
AuxIOPort.description	auxIODescription
AuxIOPort.resolution	auxIOResolution
AuxIOPort.value	auxIOValue
AuxIOPort.direction	auxIOPortDirection
AuxIOV2Table.maxDigitalPorts	maxAuxIOv2TableNumDigitalPorts
AuxIOV2Table.maxAnalogPorts	maxAuxIOv2TableNumAnalogPorts
AuxIOV2PortType.type	auxIOv2PortType
AuxIOV2Port.number	auxIOv2PortNumber
AuxIOV2Port.description	auxIOv2PortDescription
AuxIOV2Port.resolution	auxIOv2PortResolution
AuxIOV2Port.value	auxIOv2PortValue
AuxIOV2Port.direction	auxIOv2PortDirection
AuxIOV2Port.lastCommandedState	auxIOv2PortLastCommandedState

**A.2.8 SNMP Information**

Figure 27 depicts the NTCIP-defined SNMP data stored by a controller.



**Figure 27: Class Diagram of the NTCIP-Defined SNMP Data**

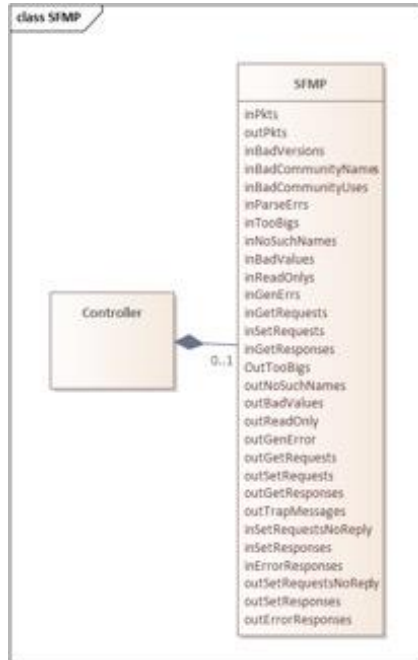
A controller can indicate the maximum SNMP packet size. The mapping of each attribute in this class to object types are defined in Table 7.

**Table 7: NTCIP-Defined SNMP Data Mapping to Class Diagram**

Property	Object Type
SnmpConfig.maxPacketSize	snmpMaxPacketSize

**A.2.9 SFMP Information**

Figure 28 depicts the SFMP data stored by a controller.



**Figure 28: Class Diagram of the SFMP Data**

A controller can indicate various statistics for SFMP. The mapping of each attribute in this class to object types are defined in Table 8.

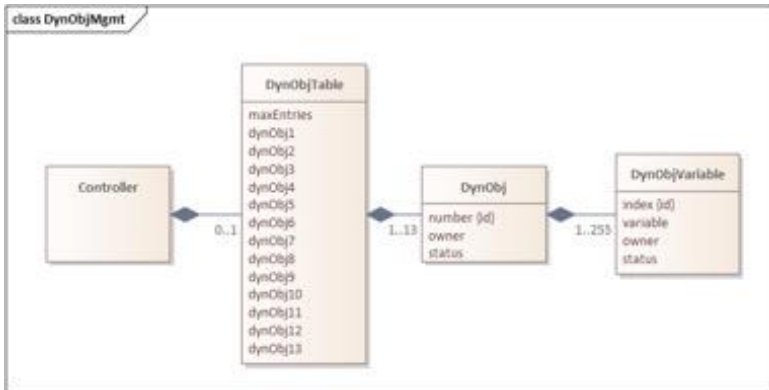
**Table 8: SFMP Data Mapping to Class Diagram**

Property	Object Type
SFMP.inPkts	sfmpInPkts
SFMP.outPkts	sfmpOutPkts
SFMP.inBadVersions	sfmpInBadVersions
SFMP.inBadCommunityNames	sfmpInBadCommunityNames
SFMP.inBadCommunityUses	sfmpInBadCommunityUses
SFMP.inParseErrs	sfmpInParseErrs
SFMP.inTooBig	sfmpInTooBig
SFMP.inNoSuchNames	sfmpInNoSuchNames
SFMP.inBadValues	sfmpInBadValues
SFMP.inReadOnly	sfmpInReadOnly
SFMP.inGenErrs	sfmpInGenErrs
SFMP.inGetRequests	sfmpInGetRequests
SFMP.inSetRequests	sfmpInSetRequests
SFMP.inGetResponses	sfmpInGetResponses
SFMP.outTooBig	sfmpOutTooBig

SFMP.outNoSuchNames	sfmpOutNoSuchNames
SFMP.outBadValues	sfmpOutBadValues
SFMP.outReadOnly	sfmpOutReadOnly
SFMP.outGenErrs	sfmpOutGenErrs
SFMP.outGetRequests	sfmpOutGetRequests
SFMP.outSetRequests	sfmpOutSetRequests
SFMP.outGetResponses	sfmpOutGetResponses
SFMP.outTrapMessages	sfmpOutTrapMessages
SFMP.inSetRequestsNoReply	sfmpInSetRequestsNoReply
SFMP.inSetResponses	sfmpInSetResponses
SFMP.inErrorResponses	sfmpInErrorResponses
SFMP.outSetRequestsNoReply	sfmpOutSetRequestsNoReply
SFMP.outSetResponses	sfmpOutSetResponses
SFMP.outErrorResponses	sfmpOutErrorResponses

**A.2.10 Dynamic Object Management Information**

Figure 29 depicts the data stored by a controller to manage dynamic objects.



**Figure 29: Class Diagram for Dynamic Object Management**

A controller can support the definition of dynamic objects along with a series of objects to manage the current value of each dynamic object, the latter feature was deprecated in NTCIP 1201 v02. The mapping of each attribute in this class to object types are defined in Table 9.

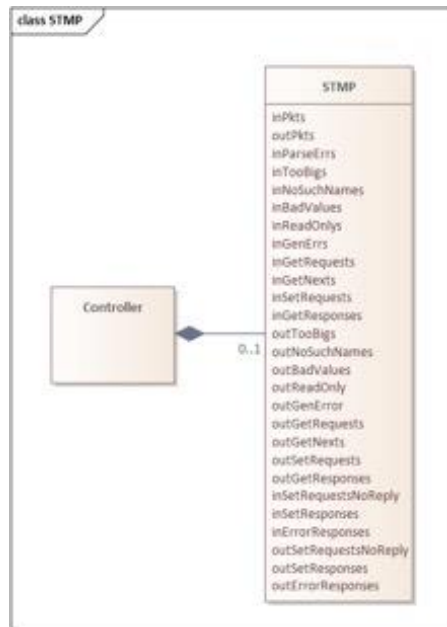
**Table 9: Dynamic Object Management Mapping to Class Diagram**

Property	Object Type
DynObjTable.maxEntries	dynObjDefTableMaxEntries
DynObjTable.dynObj1	dynObj1
DynObjTable.dynObj2	dynObj2
DynObjTable.dynObj3	dynObj3
DynObjTable.dynObj4	dynObj4

DynObjTable.dynObj5	dynObj5
DynObjTable.dynObj6	dynObj6
DynObjTable.dynObj7	dynObj7
DynObjTable.dynObj8	dynObj8
DynObjTable.dynObj9	dynObj9
DynObjTable.dynObj10	dynObj10
DynObjTable.dynObj11	dynObj11
DynObjTable.dynObj12	dynObj12
DynObjTable.dynObj13	dynObj13
DynObj.number	dynObjNumber
DynObj.owner	dynObjConfigOwner
DynObj.status	dynObjConfigStatus
DynObjVariable.index	dynObjIndex
DynObjVariable.variable	dynObjVariable
DynObjVariable.owner	dynObjOwner
DynObjVariable.status	dynObjStatus

**A.2.11 STMP Information**

Figure 30 depicts the STMP data stored by a controller.



**Figure 30: Class Diagram of STMP Data**

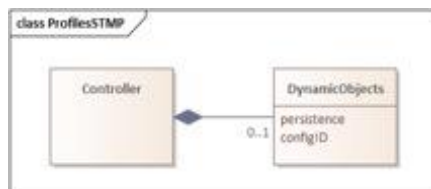
A controller can support various STMP statistics. The mapping of each attribute in this class to object types are defined in Table 10.

**Table 10: STMP Data Mapping to Class Diagram**

Property	Object Type
STMP.inPkts	stmpInPkts
STMP.outPkts	stmpOutPkts
STMP.inParsErrs	stmpInParsErrs
STMP.inTooBigs	stmpInTooBigs
STMP.inNoSuchNames	stmpInNoSuchNames
STMP.inBadValues	stmpInBadValues
STMP.inReadOnlys	stmpInReadOnlys
STMP.inGenErrs	stmpInGenErrs
STMP.inGetRequests	stmpInGetRequests
STMP.inGetNexts	stmpInGetNexts
STMP.inSetRequests	stmpInSetRequests
STMP.inGetResponses	stmpInGetResponses
STMP.outTooBigs	stmpOutTooBigs
STMP.outNoSuchNames	stmpOutNoSuchNames
STMP.outBadValues	stmpOutBadValues
STMP.outReadOnlys	stmpOutReadOnlys
STMP.outGenErrs	stmpOutGenErrs
STMP.outGetRequests	stmpOutGetRequests
STMP.outGetNexts	stmpOutGetNexts
STMP.outSetRequests	stmpOutSetRequests
STMP.outGetResponses	stmpOutGetResponses
STMP.inSetRequestsNoReply	stmpInSetRequestsNoReply
STMP.inSetResponses	stmpInSetResponses
STMP.inErrorResponses	stmpInErrorResponses
STMP.outSetRequestsNoReply	stmpOutSetRequestsNoReply
STMP.outSetResponses	stmpOutSetResponses
STMP.outErrorResponses	stmpOutErrorResponses

**A.2.12 Profiles STMP**

Figure 31 depicts the STMP profile data stored by a controller.



**Figure 31: Class Diagram of the STMP Profile Data**

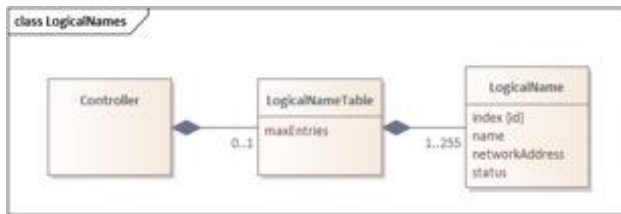
A controller can indicate the persistence and configuration identifier for dynamic objects. The mapping of each attribute in this class to object types are defined in Table 11.

**Table 11: STMP Profile Data Mapping to Class Diagram**

Property	Object Type
DynamicObjects.persistence	dynamicObjectPersistence
DynamicObjects.configID	dynamicObjectTableConfigID

**A.2.13 Logical Names Information**

Figure 32 depicts the logical names data stored by a controller.



**Figure 32: Class Diagram of the Logical Names Data**

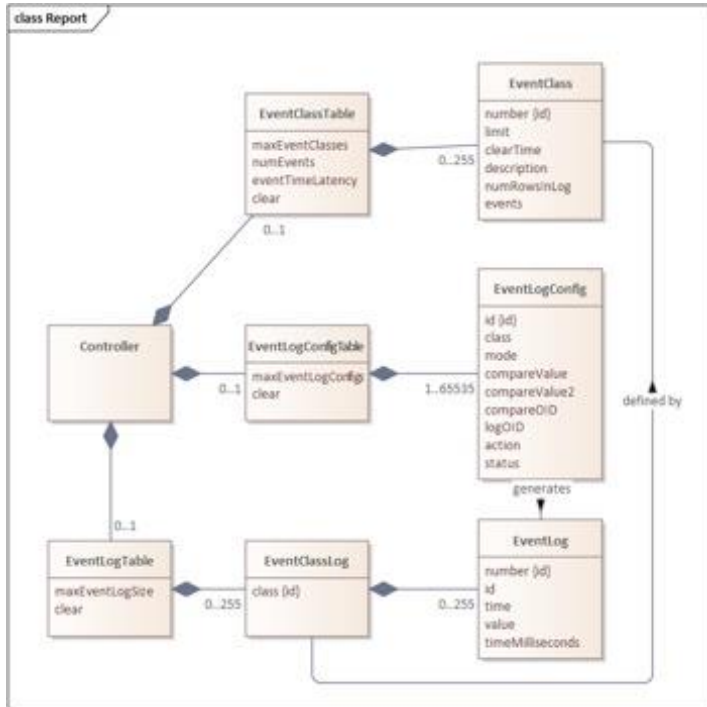
A controller can support a logical names table. The mapping of each attribute in this class to object types are defined in Table 12.

**Table 12: Logical Names Data Mapping to Class Diagram**

Property	Object Type
LogicalNameTable.maxEntries	logicalNameTranslationTableMaxEntries
LogicalName.index	logicalNameTranslationIndex
LogicalName.name	logicalNameTranslationLogicalName
LogicalName.networkAddress	logicalNameTranslationNetworkAddress
LogicalName.status	logicalNameTranslationStatus

**A.2.14 Event Report Information**

Figure 33 depicts the event report data stored by a controller.



**Figure 33: Class Diagram of the Event Report Data**

A controller can configure multiple event types and class types so that information about events can be stored in a log, organized by the defined classes. The mapping of each attribute in this class to object types are defined in Table 13.

**Table 13: Event Report Data Mapping to Class Diagram**

Property	Object Type
EventClassTable.maxEventClasses	maxEventClasses
EventClassTable.numEvents	numEvents
EventClassTable.eventTimeLatency	eventTimeLatency
EventClassTable.clear	eventClearClasses
EventClass.number	eventClassNumber
EventClass.limit	eventClassLimit
EventClass.clearTime	eventClassClearTime
EventClass.description	eventClassDescription
EventClass.numRowsInLog	eventClassNumRowsInLog
EventClass.events	eventClassNumEvents
EventLogConfigTable.maxEventLogConfigs	maxEventLogConfigs
EventLogConfigTable.clear	eventClearConfiguration



EventLogConfig.id	eventConfigID
EventLogConfig.class	eventConfigClass
EventLogConfig.mode	eventConfigMode
EventLogConfig.compareValue	eventConfigCompareValue
EventLogConfig.compareValue2	eventConfigCompareValue2
EventLogConfig.compareOID	eventConfigCompareOID
EventLogConfig.logOID	eventConfigLogOID
EventLogConfig.action	eventConfigAction
EventLogConfig.status	eventConfigStatus
EventLogTable.maxEventLogSize	maxEventLogSize
EventLogTable.clear	eventClearLog
EventClassLog.class	eventLogClass
EventLog.number	eventLogNumber
EventLog.id	eventLogID
EventLog.time	eventLogTime
EventLog.value	eventLogValue
EventLog.timeMilliseconds	eventLogTimeMilliseconds

#### A.2.15 Community Name Security Information

Figure 34 depicts the community name management data stored by a controller.



**Figure 34: Class Diagram of the Community Name Security Data**

A controller can support objects to manage the community names configured for the controller. The mapping of each attribute in this class diagram to object types are defined in Table 14.

**Table 14: Community Name Security Data Mapping to Class Diagram**

Property	Object Type
CommunityNameTable.adminName	communityNameAdmin
CommunityNameTable.namesMax	communityNamesMax
CommunityName.index	communityNameIndex
CommunityName.user	communityNameUser
CommunityName.accessMask	communityNameAccessMask

#### A.2.16 Trap Management Information

Figure 35 depicts the trap management data stored by a controller.

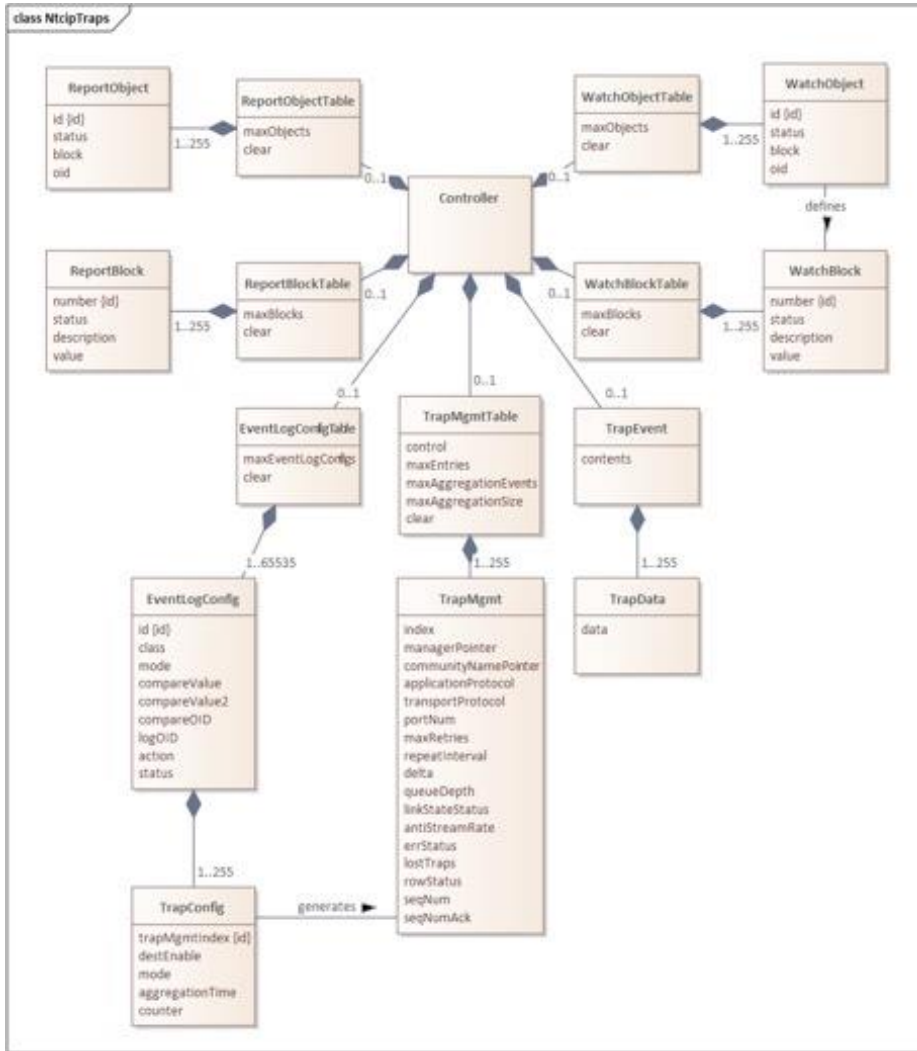


Figure 35: Class Diagram of the Trap Management Data

A controller can configure trap events, the data to send, and the parameters used to send the traps. This can include monitoring complex watch objects and sending complex report objects. The mapping of each attribute in this class to object types are defined in Table 15.

Table 15: Trap Management Data Mapping to Class Diagram

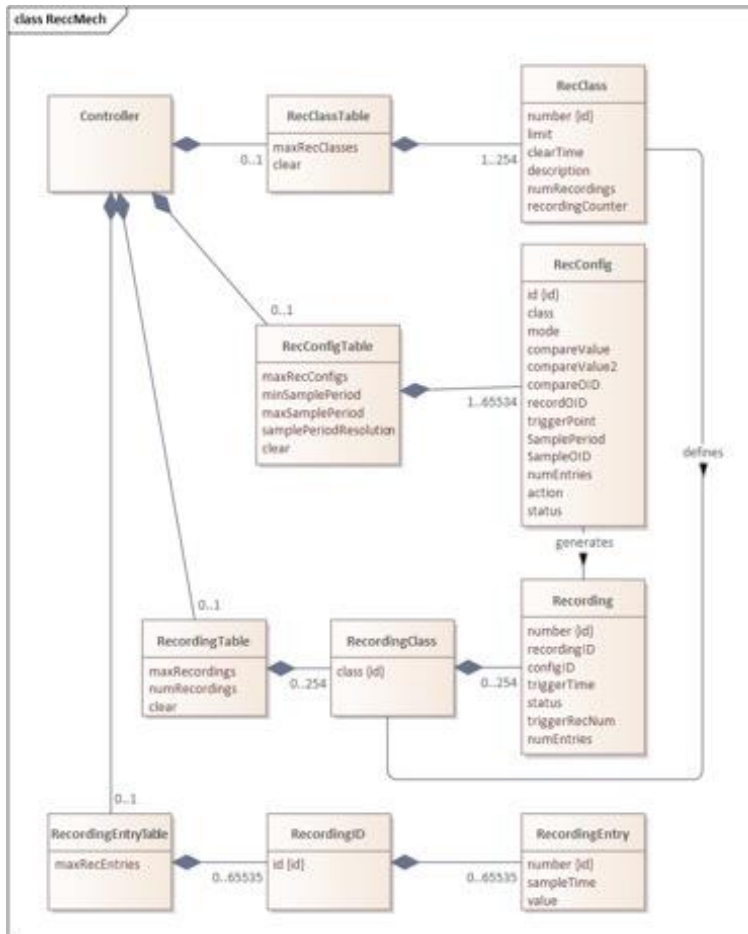
Property	Object Type
WatchObjectTable.maxObjects	maxWatchObjects

Property	Object Type
WatchObjectTable.clear	clearWatchObjects
WatchObject.id	watchID
WatchObject.status	watchStatus
WatchObject.block	watchBlock
WatchObject.oid	watchOID
WatchBlockTable.maxBlocks	maxWatchBlocks
WatchBlockTable.clear	clearWatchBlockTable
WatchBlock.number	watchBlockNumber
WatchBlock.status	watchBlockStatus
WatchBlock.description	watchBlockDescription
WatchBlock.value	watchBlockValue
ReportObjectTable.maxObjects	maxReportObjects
ReportObjectTable.clear	clearReportObjects
ReportObject.id	reportID
ReportObject.status	reportStatus
ReportObject.block	reportBlock
ReportObject.oid	reportOID
ReportBlockTable.maxBlocks	maxReportBlocks
ReportBlockTable.clear	clearReportBlockTable
ReportBlock.number	reportBlockNumber
ReportBlock.status	reportBlockStatus
ReportBlock.description	reportBlockDescription
ReportBlock.value	reportBlockValue
EventLogConfigTable.maxEventLogConfigs	maxEventLogConfigs
EventLogConfigTable.clear	eventClearConfiguration
EventLogConfig.id	eventConfigID
EventLogConfig.class	eventConfigClass
EventLogConfig.mode	eventConfigMode
EventLogConfig.compareValue	eventConfigCompareValue
EventLogConfig.compareValue2	eventConfigCompareValue2
EventLogConfig.compareOID	eventConfigCompareOID
EventLogConfig.logOID	eventConfigLogOID
EventLogConfig.action	eventConfigAction
EventLogConfig.status	eventConfigStatus
TrapConfig.trapMgmtIndex	trapMgmtManagerIndex
TrapConfig.destEnable	trapDestEnable
TrapConfig.mode	trapMode
TrapConfig.aggregationTime	trapAggregationTime
TrapConfig.counter	trapCounter
TrapMgmtTable.control	trapControl
TrapMgmtTable.maxEntries	trapMgmtMaxEntries
TrapMgmtTable.maxAggregationEvents	trapMaxAggregationEvents
TrapMgmtTable.maxAggregationSize	trapMaxAggregationSize

Property	Object Type
TrapMgmtTable.clear	clearTrapMgmtTable
TrapMgmt.index	trapMgmtManagerIndex
TrapMgmt.managerPointer	trapMgmtManagerPointer
TrapMgmt.communityNamePointer	trapMgmtCommunityNamePointer
TrapMgmt.applicationProtocol	trapMgmtApplicationProtocol
TrapMgmt.transportProtocol	trapMgmtTransportProtocol
TrapMgmt.portNum	trapMgmtPortNum
TrapMgmt.maxRetries	trapMgmtMaxRetries
TrapMgmt.repeatInterval	trapMgmtRepeatInterval
TrapMgmt.delta	trapMgmtDelta
TrapMgmt.queueDepth	trapMgmtQueueDepth
TrapMgmt.linkStateStatus	trapMgmtLinkStateStatus
TrapMgmt.antiStreamRate	trapMgmtAntiStreamRate
TrapMgmt.errStatus	trapMgmtErrStatus
TrapMgmt.lostTraps	trapMgmtLostTraps
TrapMgmt.rowStatus	trapMgmtRowStatus
TrapMgmt.seqNum	trapMgmtSeqNum
TrapMgmt.seqNumAck	trapMgmtSeqNumAck
TrapEvent.contents	trapEvent
TrapData.data	trapData

**A.2.17 Recording Mechanism (Version 1) Information**

Figure 36 depicts the data stored by a controller for the recording mechanism version 1 feature as used prior to NTCIP 1201 v04.



**Figure 36: Class Diagram of the Recording Mechanism (Version 1) Data**

indicates a controller can include data to manage the recording mechanism (version 1). The mapping of each attribute in this class to object types are defined in Table 16.

**Table 16: Recording Mechanism (Version 1) Data Mapping to Class Diagram**

Property	Object Type
RecClassTable.maxRecClasses	maxRecClasses
RecClassTable.clear	recClearClasses
RecClass.number	recClassNumber
RecClass.limit	recClassLimit
RecClass.clearTime	recClassClearTime
RecClass.description	recClassDescription

RecClass.numRecordings	recClassNumRecordings
RecClass.recordingCounter	recClassRecordingCounter
RecConfigTable.maxRecConfigs	maxRecConfigs
RecConfigTable.minSamplePeriod	recMinSamplePeriod
RecConfigTable.maxSamplePeriod	recMaxSamplePeriod
RecConfigTable.samplePeriodResolution	recSamplePeriodResolution
RecConfigTable.clear	recClearConfigurations
RecConfig.id	recConfigID
RecConfig.class	recConfigClass
RecConfig.mode	recConfigMode
RecConfig.compareValue	recConfigCompareValue
RecConfig.compareValue2	recConfigCompareValue2
RecConfig.compareOID	recConfigCompareOID
RecConfig.recordOID	recConfigRecordOID
RecConfig.triggerPoint	recConfigTriggerPoint
RecConfig.samplePeriod	recConfigSamplePeriod
RecConfig.sampleOID	recConfigSampleOID
RecConfig.numEntries	recConfigNumEntries
RecConfig.action	recConfigAction
RecConfig.status	recConfigStatus
RecordingTable.maxRecordings	maxRecordings
RecordingTable.numRecordings	numRecordings
RecordingTable.clear	recClearRecordingData
RecordingClass.class	recordingClass
Recording.number	recordingNumber
Recording.recordingID	recordingID
Recording.configID	recordingConfigID
Recording.triggerTime	recordingTriggerTime
Recording.status	recordingStatus
Recording.triggerRecNum	recordingTriggerRecNum
Recording.numEntries	recordingNumEntries
RecordingEntryTable.maxRecEntries	maxRecEntries
RecordingID.id	recordingID
RecordingEntry.number	recEntryNumber
RecordingEntry.sampleTime	recSampleTime
RecordingEntry.value	recValue

## Annex B SUMMARY OF CHANGES [INFORMATIVE]

To the extent reasonable, the NTCIP community attempts to minimize the number of changes to an NTCIP standard to minimize interoperability issues among different versions of a single NTCIP standard. However, on occasion, issues are identified with existing NTCIP standards that necessitate a change. When rectifying such issues, NTCIP standards attempt to minimize the impact on existing implementations. Annex B:

1. identifies each issue that has resulted in a significant revision from the previous major version of this document,
2. provides a description of the revision made, and
3. includes a brief analysis of the impact of each revision on existing implementations.

### B.1 INCLUSION OF MIBS FROM NTCIP 1103 V03

As a part of the migration to a more secure NTCIP, NTCIP 1103 v03 was deprecated and the MIBs formally contained in NTCIP 1103 were migrated to NTCIP 1201v04 per the recommendations of NTCIP 9014.

This specific change was a document management issue and does not affect implementations.

### B.2 UPGRADE TO SMIv2 AND DEPRECATE NTCIP 1103 V03 AND NTCIP 1201 V03 OBJECTS

The primary purpose of the NTCIP 1201 v04 update was to improve the security of NTCIP per the recommendations of NTCIP 9014. The decision to migrate to SNMPv3 required the conversion of NTCIP MIBs to SMIv2 format. The conversion to SNMPv3 also prevents any dialog from being backwards compatible with prior versions of NTCIP. As such, the NTCIP community determined that this would be an appropriate time to correct minor flaws that had been discovered over the decades of successful NTCIP deployments.

To conform to SMIv2 format, some objects had to be deprecated and replaced. For example, SMIv2 imposes stricter requirements on not allowing Counter objects to be writable, but several NTCIP objects were defined as writable Counters.

Further, some objects and tables had to be deprecated to address known security issues. For example, the event log did not adequately check security credentials when monitoring data to create events or to record information in the log. To correct this issue, multiple tables had to be deprecated and replaced.

Finally, during the analysis other potential areas of improvement were identified and the NTCIP community concluded it was better to address the issues during a major update rather than waiting for a later date.

The result of the analysis was nearly all NTCIP 1103 v03 and NTCIP 1201 v03 objects were recommended for deprecation due to some technical issue. Further, equivalents for most of these objects had already been developed within either Internet RFCs or ISO standards. As a result, the NTCIP community decided to:

1. reference Internet RFCs and ISO standards for objects that are applicable to all device types (eliminating the need to reference NTCIP 1201 for future designs);
2. work with ISO/TC 204 to ensure that features missing from the Internet RFCs would be included within ISO 26048-1, which becomes the primary replacement of NTCIP 1201 functionality for future design; and
3. upgrade the MIBs previously defined in NTCIP 1103 v03 and NTCIP 1201 v03 to SMIv2 to provide an unambiguous definition of how to exchange previously defined data using SNMPv3, as might be required for communications with a proxy agent that serves as a front-end to a SNMPv1 device.

While there were a few, relatively simple, existing objects that could have been reused in the current recommended design, supporting these objects in both the NTCIP 1201 and ISO 26048-1 MIBs would require the definition of a third MIB imported by both of the previous two. The NTCIP community decided it would be less confusing to deprecate the previously defined object-types and rely on the new standards to fully define their own operation.

The net impact of these changes is that the specific data objects defined for future implementations (i.e., as defined in ISO 26048-1) are not directly backwards compatible with prior NTCIP 1201 objects. However, the migration to SNMPv3 would have prevented this interoperability anyway. Many of the objects deprecated in this document have similar parallel objects defined in ISO 26048-1 (as specified in their "superseded by" clause) allowing implementations to support easier conversions.

NOTE — In some cases, objects previously defined are no longer needed due to the migration to SNMPv3 and the practical data capacity requirements for exchanging secure messages. For example, support for SFMP, STMP, and PMPP are being withdrawn while the trap management feature is being significantly revised to align with design elements included natively within SNMPv3. Thus, not all previous objects have replacements.

### **B.3 SUMMARY**

The result of the changes identified above was to upgrade all NTCIP 1103 v03 and NTCIP 1201 v03 MIBs to SMIv2 and to mark all of their objects as deprecated. Replacement objects for future design are contained in ISO 26048-1. Each object definition contained in this document provides a recommendation for what object(s) should supersede/supplant the object or why no superseding/supplanting object is identified.

§