A Proposed User Comment Draft to the Joint Committee on the NTCIP

NTCIP 1201 v04.04

National Transportation Communications for ITS Protocol

Global Object (GO) Definitions

February 13, 2023

Published by

American Association of State Highway and Transportation Officials (AASHTO)
444 North Capitol Street, N.W., Suite 249
Washington, D.C. 20001

Institute of Transportation Engineers (ITE)
1627 I ("Eye") Street, N.W., Suite 600
Washington, D.C. 20006

National Electrical Manufacturers Association (NEMA)
1300 North 17th Street, Suite 1752
Rosslyn, Virginia 22209-3801

Revision History for NTCIP 1201 v04

to be removed once approved

Revision	Date	Author	Description (latest on top)
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

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 Web site, subject to the terms and conditions of this license agreement:

a) you may download one copy of each PDF file for personal, noncommercial, and intraorganizational use only;
b) ownership of the PDF file is not transferred to you; you are licensed to use the PDF file;
c) you may make one more electronic copy of the PDF file, such as to a second hard drive or burn to a CD;
d) you agree not to copy, distribute, or transfer the PDF file from that media to any other electronic media or device;
e) you may print one paper copy of the PDF file;
f) you may make one paper reproduction of the printed copy;
g) any permitted copies of the PDF file must retain the copyright notice, and any other proprietary notices contained in the file;
h) 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;
i) 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, including derivative works, of the DD or MIB, including copies for commercial distribution, provided that:

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

b) the copies or derivative works are not made part of the standard publications or works offered by other standard developing organizations or publishers or as works-for-hire not associated with commercial hardware or software products intended for field implementation;
c) use of the DD or MIB is restricted in that the syntax fields may only be modified to define: 1) a more restrictive subrange; or 2) a subset of the standard enumerated values; or 3) a set of deprecated and defined enumerated values for systems supporting multiversion interoperability;
d) the description field may be modified but only to the extent that: 1) the more restrictive subrange is defined; and 2) only those bit values or enumerated values that are supported are listed.

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 particular 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 circumstances. 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 v04 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:

- Doug Crawford (Chair)
- Steve Bostrom
- Wolfgang Buckel
- Michael Forbis
- Joe Gorman
- Terry Haukom
- Patrick Leung
- Robert Lopes
- Alex Mousadi
- Satya Muthuswamy
- Bob Rausch
- Rodney Schilling
- Shea Tomsin
- Walt Townsend

Other individuals providing input include:

- Ashraf Ahmed
- Justin Anderson
- Kingsley Azubike
- Ralph Boaz
- Russ Brookshire
- Patrick Chan
- Stan Chow
- Deborah Curtis
- Dustin DeVoe
- Brian Doherty
- Edward Fok
- Sai Kiran Golla
- Jacob Grivette
- Mohammad Iraki
- Haydar Issa
- Denver Kruse
- AJ Lahiri
- Stephane Lapierre
- Christopher Lyons
- Gary Molnar
- Keith Patton
- Mahesh Pawar
- Frank Perry
- Nu Rosenbohm
- Jay Schultz
- Kellen Shain
- Douglas Tarico
- Nicola Taveres
- John Thai
- Andrew Valdez
- Kenneth Vaughn
- Sankeer Vulupala
- Robert White

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:

- U.S. Department of Transportation, Research and Innovative Technology Administration
- U.S. Department of Transportation, Federal Highways Administration
- AECOM
- Applied Information
- City of Anaheim, CA
- ConSysTec
- Caltrans
- Daktronics
- Econolite Control Products, Inc.
- Florida Department of Transportation
- Georgia DOT
- Kapsch
- KLD Engineering
- Michigan DOT
- Minnesota DOT
- Nevada DOT
- Noblis
- Parsons
- Pillar Consulting
- Q-Free
- Siemens ITS
- Signalisation Ver-Mac, Inc.

- TransCore
- Transport Canada
- Trevilon
- Washington State DOT
- 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.

NOTE—NTCIP 1201:1996 is sometimes referenced as NTCIP 1201:1997. In NTCIP 1201 v03, it is referenced as NTCIP 1201 v01 (following a revised NTCIP standards designation convention).

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.

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.

A User Comment should be submitted to this address:

NTCIP Coordinator
National Electrical Manufacturers Association
1300 North 17th Street, Suite 1752
Rosslyn, Virginia 22209-3801
e-mail: 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

Version	Date	Description (latest on top)
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 in each 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 one normative and three 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

FIGURES

Page

TABLES

Page

< This page is intentionally left blank. >

Section 1
GENERAL

1.1 SCOPE

The messaging between a Transportation Management Center and its field devices is accomplished by using the NTCIP Application Layer services to convey requests to access or modify values stored in a given device; the parameters used to store these values are referred to as objects. Objects are instances of object types; some object types have singular instances within a device, while others can occur in conceptual tables and have multiple instances. This document identifies and defines object types that may be supported by multiple device types (e.g., actuated signal controllers and dynamic message signs). In the NTCIP family of standards, object types for a given device type are grouped in a device-type-specific data dictionary standard.

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 standard listed.

AASHTO / ITE / NEMA NTCIP 8004 v03	*Structure and Identification of Management Information (SMI)* published (Pending)
IETF RFC 2021	*Remote Network Monitoring Management Information Base Version 2 using SMIv2*, January 1997
IETF RFC 2578	*Structure of Management Information Version 2 (SMIv2)*, April 1999
IETF RFC 2579	*Textual Conventions for SMIv2*, April 1999
IETF RFC 2580	*Conformance Statements for SMIv2*, April 1999
IETF RFC 3411	*An Architecture for Describing Simple Network Management Protocol (SNMP) Management Framework*, December 2002
IETF RFC 3416	*Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)* , December 2002
ISO/IEC 2382:2015	*Information technology — Vocabulary*
ISO 20684-1	*Intelligent transport systems — Roadside equipment SNMP data interface — Part 1: Overview*

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 8005 v02	*Procedures for Creating Management Information Base (MIB) Files* published (Pending)
IETF RFC 854	*Telnet Protocol Specification*
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 3418	Management Information Base (MIB) for the *Simple Network Management Protocol (SNMP)*, December 2002
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
ISO 15784-2	*Intelligent transport systems — Data exchange involving roadside equipment communication — Part 2: Centre to field device communications using SNMP*
ISO 20684-2	*Intelligent transport systems — Roadside equipment SNMP data interface — Part 2: Generalized field device basic management*
ISO 20684-3	*Intelligent transport systems — Roadside equipment SNMP data interface — Part 3: Triggers*
ISO 20684-4	*Intelligent transport systems — Roadside equipment SNMP data interface — Part 4: Notifications*
ISO 20684-5	*Intelligent transport systems — Roadside equipment SNMP data interface — Part 5: Logs*
ISO 20684-6	*Intelligent transport systems — Roadside equipment SNMP data interface — Part 6: Commands*
ISO 20684-7	*Intelligent transport systems — Roadside equipment SNMP data interface — Part 7: Support features*
NEMA TS 2-2016	*Traffic Controller Assemblies with NTCIP Requirements*

1.2.3 Contact Information

1.2.3.1 IAB Documents

For Internet Architecture Board (IAB) documents, contact:

Internet Architecture Board (IAB)
www.rfc-editor.org
www.rfc-editor.org/repositories.html

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
e-mail: 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

1.3 GENERAL STATEMENTS

Sections 2 and 3 define objects that represent the current preferred design. Sections 4 through 16 define objects that are deprecated and represent designs from previous versions of NTCIP. The object types defined in this document are expected to be used by different device types, such as actuated signal controllers (ASC), variable message signs, and ramp meter controllers. The object types are defined using the OJBECT-TYPE macro defined in RFC 2578 per the rules defined in NTCIP 8004.

The MODULE-COMPLIANCE statements contained within each MIB duplicates and extends requirements that may be contained within the PRL of that standard. A device-specific module (e.g., the module for NTCIP 1202) will typically contain one MODULE-COMPLIANCE statement for each revision of the module. Each version will import object groups as needed and identify any appropriate constraints for implementation within that device type. In case of any conflict between the PRL and a referenced MODULE-COMPLIANCE statement, the PRL take precedence. To conform to a MIB defined within this document, implementations shall:

- support the defined MAX-ACCESS of each supported object, unless the associated MIB compliance table indicates a different minimum access for the object type;
- support at least the defined minimum access for each object type when one is indicated in the associated MIB compliance table,
- support the full range of values defined by the SYNTAX for each supported object, unless the associated MIB compliance table indicates a refined SYNTAX for the object type;
- support at least the range of values as defined in the refined SYNTAX for each object type when one is indicated in the associated MIB compliance table.

While conformance requirements frequently correspond to the nodal structure, they may relate to objects that are not lexicographically ordered. For example, the NTCIP1201-RecMechV2 MIB requires support for objects under the recMechV2 and adminRecMechV2 nodes, as defined by the NTCIP1201-RecMechV2 MIB and objects under the fdOwnerTable node, as defined by ISO 20684-3.

This document contains four informative annexes. The logic for using some of the more complex features of this standard is provided in Annex A. Annex B provides informative class diagrams and object tree diagrams to provide the reader with a higher-level overview of the object types defined in this document. Annex C is reserved for potential future inclusion of test procedures. Annex D provides information about changes between each version of this standard.

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. Information technology Electrical and electronic terms not defined here are used in accordance with their definitions in ISO/IEC 2382. English words not defined here or in ISO/IEC 2382 are used in accordance with their definitions in *Webster's New Collegiate Dictionary*.

ASC	Actuated Signal Controller
Class	An abstraction of any kind of object that may be described. NOTE—Equivalent to an ISO 14817 Object Class.
Component	A central system, field device, etc., that supports NTCIP.
Control Object	writeable object that immediately implements the result of set operations even when this object is in the transaction state

Cyclic Redundancy Check (CRC)	Polynomial algorithm performed on a specified range of data resulting in a 16 or 32 bit value
Data Value	The value of a data element.
DST	Daylight Saving Time
Deprecated	Defined in NTCIP 8004 v02.
Feature	A capability of a component.
Interchangeable	A 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.
Interoperability	The ability of two or more systems or components to exchange information and use the information that has been exchanged. NOTE—See IEEE *Standards Dictionary, Glossary of Terms and Definitions*.
Management Information Base (MIB	A collection of objects defined using Abstract Syntax Notation One (ASN.1) that can be accessed via a network management protocol.
Obsolete	Defined in NTCIP 8004 v02.
Parameter Object	writeable object that has set operations buffered when this object is in the transaction state
Point-to-MultiPoint Protocol (PMPP)	A transportation specific subnetwork layer protocol that enables communication between multiple devices on the same communications line/channel.
PRL	Protocol Requirements List
Profile	Refers to a set of protocols, each of which operates independently on one of the seven (7) OSI Layers, if this layer is utilized. NOTE—Different protocols are utilized at the same layer within different profiles.
Simple Transportation Management Protocol (STMP)	Part of the Transportation Management Protocols of the NTCIP effort. NOTE—See NTCIP 1103 v02. STMP provides a simple and bandwidth efficient mechanism to communicate with field devices.
Status Object	read-only object that is not writeable

Transaction Object	parameter object that cannot be altered without using the transaction state of this object
UML	Unified Modeling Language

Section 2
GLOBAL STRUCTURE OF MANAGEMENT INFORAMTION (SMI) [NORMATIVE]

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

2.1 GLOBAL SMI MIB HEADER

```
NTCIP1201-Global   DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-IDENTITY
                                                FROM SNMPv2-SMI
                                                          -- RFC 2578
devices, deviceAdmin
                                                FROM NTCIP8004-Transportation;

global MODULE-IDENTITY
  LAST-UPDATED "202212120500Z"
  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
    "This MIB defines the structure of management information under the
     global node. In particular, it defines the global and globalAdmin nodes
     that are used by other MIBs contained in NTCIP 1201.

     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6

     Copyright (C) 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                          Joint NEMA, AASHTO, and ITE
                     NTCIP Management Information Base
                           DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
```



```
    specifies the capabilities of a specific implementation. By convention,
    such a companion MIB module might have the same module name followed by
    "-", followed the enterprise's Internet Assigned Number Authority
    (IANA)-assigned enterprise number.

    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 OR 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, OR 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 you.

    Use of these materials does not constitute an endorsement or affiliation
    by or between AASHTO, ITE, or NEMA and you, your company, or your
    products and services.

    NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202212120500Z"
  DESCRIPTION
    "NTCIP 1201 v04 - The first version of the NTCIP1201-Global MIB.
     The nodes defined within this MIB were previously defined in NTCIP1201-
     v03 in SMIv1 format. "
  ::= { devices 6 }
```

2.2 NODE DEFINITIONS

```
adminGlobal  OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used to group NTCIP 1201 objects that should only be accessible
     by users with administrator rights. Grouping these objects under the
     administrator node allows easy configuration of access control rights.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6"
  ::= { deviceAdmin 6 }

globalV2     OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used to group the MIBs associated with the second major version
     of global objects.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9"
  ::= { global 9 }

adminGlobalV2 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used to group globalV2 objects that should only be accessible
```

```
     by users with administrator rights.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9"
  ::= { adminGlobal 9 }

END --NTCIP1201-Global
```

Section 3
DATABASE MANAGEMENT INFORMATION BASE (MIB)
[NORMATIVE]

The text provided from Section 3.1 through the end of Section Section 3 (except the headings) constitutes the standard NTCIP1201-DbMgmtV2 MIB.

The following table indicates which object groups and capabilities are required for each version of NTCIP 1201 for the objects contained within this MIB.

Table 1: Compliance Statement for NTCIP1201-DbMgmtV2 MIB

Group	v04
dbMgmtV2GroupR1	M

3.1 DATABASE MANAGEMENT MIB HEADER

```
NTCIP1201-DbMgmtV2   DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY
                                                FROM SNMPv2-SMI
                                                        -- RFC 2578
OBJECT-GROUP
                                                FROM SNMPv2-CONF
                                                        -- RFC 2580
SnmpAdminString
                                                FROM SNMP-FRAMEWORK-MIB
                                                        -- RFC 3411
globalV2, adminGlobalV2
                                                FROM NTCIP1201-Global;


dbMgmtV2 MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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
    "This MIB defines a mechanism to allow a manager to modify a potentially
     large number of interrelated configuration parameters within a field
     device as a single operation.

     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.1

     Copyright © 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.
```

```
                    Joint NEMA, AASHTO, and ITE
                  NTCIP Management Information Base
                        DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
     specifies the capabilities of a specific implementation. By convention,
     such a companion MIB module might have the same module name followed by
     "-", followed the enterprise's Internet Assigned Number Authority
     (IANA)-assigned enterprise number.

     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 OR 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, OR 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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  DESCRIPTION
    "NTCIP 1201 v04 - The first version of the NTCIP1201-DbMgmtV2 MIB
     The objects contained within this MIB replace similar objects that
     were previously defined in NTCIP1201-v03 in SMIv1 format. The
     previous objects have been deprecated, but are documented in SMIv2
     format in NTCIP1201-DbMgmtV1."
  ::= {globalV2 1}
```

3.2 NODE DEFINITIONS

```
adminDbMgmtV2 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "<Definition> A node used to group administrative objects for the
     dbMgmtV2 feature.
```



```
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.1"
  ::= {adminGlobalV2 1}

dbMgmtV2Conformance OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "<Definition> A node used to contain conformance concepts for the
     dbMgmtV2 feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.1.127"
::= {dbMgmtV2 127}

dbMgmtV2Compliances OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "<Definition> A node used to contain compliance statements for the
     dbMgmtV2 feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.1.127.1"
::= {dbMgmtV2Conformance 1}

dbMgmtV2Groups OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "<Definition> A node used to contain object groups for the dbMgmtV2
     feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.1.127.2"
::= {dbMgmtV2Conformance 2}
```

3.3 DATABASE TRANSACTION OBJECTS

3.3.1 Database Mode

```
dbMgmtV2Mode   OBJECT-TYPE
SYNTAX      INTEGER {   normal (1),
                        transaction (2),
                        verify (3),
                        done (4)
                        }
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
      "<Definition>
      This object allows a manager (through its command generator) to modify
      a potentially large number of interrelated configuration parameters
      within a field device (through its command responder). In the normal
      state, set operations are processed without any special rules.  In the
      transaction state, value assignments for parameter objects shall be
      buffered until the verify state performs a consistency check.  When the
      consistency check completes, the command responder automatically
      transitions to the done state where a normal or transaction command may
      be issued. The normal state is the default state of this object upon
      initialization.

      <Format>
      All writable objects shall be treated as control objects unless
      otherwise specified (e.g., within a MIB or device-type standard).
      This object is a control object.
```

Implementations shall not change the designation (e.g., control, parameter, or transaction) of any writable object type from that assigned in the standards to which it claims conformance.

The following rules are applied during the processing of SetRequest-PDUs as defined in RFC 3416 Clause 4.2.5:

NORMAL:
When this object is in the 'normal' state:
1. An attempt to set a transaction object shall cause Step 9 of RFC 3416 Clause 4.2.5 to fail, thereby resulting in a ‘notWritable’ error.
2. An attempt to set this object to any value other than 'transaction' shall cause Step 10 of RFC 3416 Clause 4.2.5 to fail, thereby resulting in an 'inconsistentValue' error.
3. When setting this object to 'transition', the second phase of the set operation shall include the command responder copying all the parameter objects currently stored in the device into a buffer. A failure of this copy operation shall result in a failure in the second phase of the set operation and thereby result in a 'commitFailed' error and this object remaining in the 'normal' state.

TRANSACTION: When this object is in the 'transaction' state:
1. An attempt to set a parameter object using a securityName other than the securityName that was used to start this transaction session shall cause Step 9 of RFC 3416 Clause 4.2.5 to fail, thereby resulting in a 'notWritable' error.
2. An attempt to set this object using a securityName other than the securityName that was used to start this transaction session shall cause Step 9 of RFC 3416 Clause 4.2.5 to fail, thereby resulting in a 'notWritable' error.
3. An attempt to set this object to any value other than 'normal' or 'verify' shall cause Step 10 of RFC 3416 Clause 4.2.5 to fail, thereby resulting in an 'inconsistentValue' error.
4. Within the second phase of the set operation, the command responder shall buffer the creation of and assignments to parameter objects rather than immediately implementing them.
5. Within the second phase of the set operation to set this object to 'normal', the agent shall discard all data in the transaction session buffer.

NOTE: Control and status objects are processed normally given the above constraints. For example, another securityName can be used to set a control object without error. However, if that request also includes a parameter object, an 'inconsistentValue' error will be generated and neither the control object nor the parameter object will be altered. If the securityName used to initiate the transaction session is used to request setting both a control and a parameter object in a single request, the control object is immediately implemented while the parameter object request is buffered. GetRequest-PDUs are not affected by transaction sessions; they return the values currently implemented while ignoring any buffered values.

VERIFY: Upon entry into the verify state, the command responder shall initiate the consistency check process.

```
When this object is in the 'verify' state:
1. An attempt to set a parameter object using a securityName
   other than the securityName that was used to start this
   transaction session shall cause Step 9 of RFC 3416 Clause
   4.2.5 to fail, thereby resulting in a 'notWritable' error.
2. An attempt to set this object using a securityName other than
   the securityName that was used to start this transaction
   session shall cause Step 9 of RFC 3416 Clause 4.2.5 to fail,
   thereby resulting in a 'notWritable' error.
3. An attempt to set a parameter object shall cause Step 10 of
   RFC 3416 Clause 4.2.5 to fail, thereby resulting in an
   'inconsistentValue' error.
4. An attempt to set this object to any value shall cause Step 10
   of RFC 3416 Clause 4.2.5 to fail, thereby resulting in an
   'inconsistentValue' error.

When consistency checks are complete the device shall update the
values of dbVerifyStatusV2 and dbVerifyErrorV2 and automatically
transition this object to the 'done' state.

DONE: When this object is in the 'done' state:
1. The value of dbVerifyStatusV2 and dbVerifyErrorV2 shall
   indicate whether the consistency checks found any errors.
2. An attempt to set a parameter object using a securityName
   other than the securityName that was used to start this
   transaction session shall cause Step 9 of RFC 3416 Clause
   4.2.5 to fail, thereby resulting in a 'notWritable' error.
3. An attempt to set this object using a securityName other than
   the securityName that was used to start this transaction
   session shall cause Step 9 of RFC 3416 Clause 4.2.5 to fail,
   thereby resulting in a 'notWritable' error.
4. An attempt to set a parameter object shall cause Step 10 of
   RFC 3416 Clause 4.2.5 to fail, thereby resulting in an
   'inconsistentValue' error.
5. An attempt to set this object to any value other than 'normal'
   or 'transaction' shall cause Step 10 of RFC 3416 Clause 4.2.5
   to fail, thereby resulting in an 'inconsistentValue' error.
6. An attempt to set this object to any value when this object is
   in the process of implementing the buffer while shifting from
   the 'done' state to the 'normal' state shall cause Step 10 of
   RFC 3416 Clause 4.2.5 to fail, thereby resulting in an
   'inconsistentValue' error.
7. An attempt to set this object to a value of 'transaction' shall
   cause the command responder to re-enter the 'transaction' state
   without affecting the contents of the buffer.
8. An attempt to set this object to a value of 'normal' when
   dbVerifyStatusV2 has a value of 'doneWithNoError' shall result in
   the second phase of the set operation including the assignment of
   all the buffered data to the device's parameter objects. Any failure
   in the assignment process shall result in a failure in the second
   phase of the set operation and thereby result in the Response-PDU
   indicating either a 'commitFailed' or 'undoFailed' error, as
   appropriate, and this object remaining in the 'done' state. The
   values of dbVerifyStatusV2 and dbVerifyErrorV2 shall be updated at
   the end of the operation as appropriate.
```

```
      9. An attempt to set this object to a value of 'normal' when
         dbVerifyStatusV2 has a value other than 'doneWithNoError' shall
         result in the transaction buffer being discarded during the second
         phase of the set operation.

      Usage
      Consistency checks are intended to analyze parameter objects 'in
      context' by treating them as an interrelated whole rather than
      separate non-related data items. Other standards (e.g., device-
      type standards) should define the minimum consistency check rules
      to be implemented for their context. Implementations may include
      additional, custom consistency checks beyond those standardized.

      NOTE: Other requests (e.g., GetRequest-PDUs and SetRequests
      containing only control objects) can be processed as normal
      regardless of the state of this object.

      <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.1.1”
DEFVAL      {normal}
::= { dbMgmtV2 1 }
```

3.3.2 Database Status

```
dbMgmtV2Status   OBJECT-TYPE
SYNTAX      INTEGER {   normal (1),
                        transaction (2)
                        verifying (3),
                        doneWithNoError (4),
                        doneWithError (5),
                        commitFailed (6),
                        undoFailed (7) }
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
      “<Definition> This object indicates the status of the consistency
      checking associated with the dbCreateTransactionV2 object.

      <Format> When dbCreateTransactionV2 is in the 'normal' or
      'transaction' state, this object shall indicate the respective
      value. When dbCreateTransactionV2 is in the 'verify' state, this
      object shall indicate a value of 'verifying'.

      When in the 'done' state, this object shall have one of the
      following values:
      doneWithNoError – The consistency checks did not identify any
        inconsistencies with the buffered parameter objects.
      doneWithError – The consistency checks identified one or more
        inconsistencies with the buffered parameter objects.
      commitFailed – the attempt to assign the buffered parameter
        objects failed and the device has been restored to its previous
        state; dbCreateTransactionV2 can be set to 'transaction' and
        the process tried again
      undoFailed – the attempt to assign the buffered parameter objects
        failed and the device’s database is in an unknown state with
        potential inconsistencies; dbCreateTransactionV2 can be set to
        'transaction' and the process tried again.
```



```
      <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.1.2"
::= { dbMgmtV2 2 }
```

3.3.3 Database Error

```
dbMgmtV2Error   OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
      "<Definition> This object contains a textual description of or a
      reference to an error or condition that was found by the
      consistency checks implemented with the dbCreateTransactionV2
      process. The value of this object shall be updated each time
      dbVerifyStatusV2 is updated. If dbCreateTransactionV2 is in the
      'normal' state due to a reset command of dbTransactionReset, this
      field shall indicate 'Reset received' as the first characters in
      the string. Otherwise, unless the status reflects an error state
      (i.e., doneWithError, commitFailed, or undoFailed), this object
      shall indicate 'No error' as the first characters in the string.
      <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.1.3"
::= { dbMgmtV2 3 }
```

3.3.4 Database Transaction Reset

```
adminDbMgmtV2Reset   OBJECT-TYPE
SYNTAX      INTEGER {   normal (1),
                        reset (2)}
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
      "<Definition> This object allows a user with sufficient rights to
      force the dbCreateTransactionV2 object back to the 'normal'
      state.

      <Format>
      An attempt to set this object to 'normal' shall cause Step 6 of
      RFC 3416 Clause 4.2.5 to fail, thereby resulting in a
      'wrongValue' error.

      An attempt to set this object to 'reset' when the
      dbCreateTransactionV2 is in the process of implementing the
      buffer while shifting from the 'done' state to the 'normal' state
      shall cause Step 10 of RFC 3416 Clause 4.2.5 to fail, thereby
      resulting in an 'inconsistentValue' error.

      An attempt to set this object to 'reset' when the
      dbCreateTransactionV2 is in the 'verify' state shall cause Step
      10 of RFC 3416 Clause 4.2.5 to fail, thereby resulting in an
      'inconsistentValue' error.

      Phase 2 of the set operation shall include discarding any
      information that is in the buffer.

      <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.1.1"
::= { adminDbMgmtV2 1 }
```

3.4 DATABASE MANAGEMENT GROUP

```
dbMgmtV2GroupR1 OBJECT-GROUP
  OBJECTS { dbMgmtV2Mode,
            dbMgmtV2Status,
            dbMgmtV2Error,
            adminDbMgmtV2Reset }
  STATUS     current
  DESCRIPTION
    "<Definition> The objects necessary for managing the database transaction
     feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.1.127.2.1"
  ::= {dbMgmtV2Groups 1}


END  --NTCIP1201-DbMgmtV2
```


Section 4
RECORDING MECHANISM MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

The text provided from Section 4.1 through the end of Section Section 4 (except the headings) constitutes the standard NTCIP1201-RecMechV2 MIB.

The following table indicates which object groups and capabilities are required for each version of NTCIP 1201 for the objects contained within this MIB.

Table 2: Compliance Statement for NTCIP1201-GlobalV1 MIB

Group	v04
ISO20684-7-Owner.fdOwnerGroupR1	M [1, 2]
recMechV2GroupR1	M [3, 4, 5]

[1] For the purpose of this MIB, implementations can restrict fdOwnerStorageType to any sub-range that includes at least one value other than 'other'.
[2] For the purpose of this MIB, implementations can restrict fdOwnerID to any range with a lower bound of 1.
[3] Implementations can restrict recMechV2ClassID to any range with a lower bound of 1.
[4] Implementations can restrict recMechV2FactoryID to any range with a lower bound of 1.
[5] Implementations can restrict recMechV2FactorySampleLimit to any range with a lower bound of 1.
[6] Implementations can restrict recMechV2RecordingID to any range with a lower bound of 1.

4.1 MIB HEADER

```
NTCIP1201-RecMechV2   DEFINITIONS ::= BEGIN
IMPORTS

zeroDotZero, MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Unsigned32
                                                FROM SNMPv2-SMI
                                                  -- RFC 2578
RowPointer, RowStatus, StorageType, TruthValue, VariablePointer
                                                FROM SNMPv2-TC
                                                  -- RFC 2579
OBJECT-GROUP
                                                FROM SNMPv2-CONF
                                                  -- RFC 2580
SnmpAdminString, SnmpSecurityModel, SnmpSecurityLevel
                                                FROM SNMP-FRAMEWORK-MIB
                                                  -- RFC 3411
ITSCounter16, ITSCounter32, ITSClear, ITSDateStamp, ITSDailyTimeStamp,
ITSOerString, ITSPercent, ITSPositive8, ITSUnsigned8, ITSUnsigned16
                                                FROM ISO20684-1-TextConv
fdOwnerEntry, fdOwnerID
                                                FROM ISO20684-7-Owner
globalV2, adminGlobalV2
                                                FROM NTCIP1201-Global;

recMechV2 MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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
    "The MIB that defines a high-resolution recording mechanism that allows a
     device to record information at a high rate for a defined period of
     time.

     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2

     Copyright (C) 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                              Joint NEMA, AASHTO, and ITE
                          NTCIP Management Information Base
                                DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
     specifies the capabilities of a specific implementation. By convention,
     such a companion MIB module might have the same module name followed by
     "-", followed the enterprise's Internet Assigned Number Authority
     (IANA)-assigned enterprise number.

     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 OR 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, OR 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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.
```



```
     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  DESCRIPTION
    "NTCIP 1201 v04 - Original version of NTCIP1201-RecMechV2 MIB.
     The previous MIB module was renamed to NTCIP1201-RecMechV1. RecMechV2
     was significantly revised to be triggered by the ISO 20684-3 trigger
     mechanism."
  REVISION "202007150000Z"
  DESCRIPTION
    "Updated IMPORTS statement to reference NTCIP8004v02 (electronic only)"
  REVISION "201611070000Z"
  DESCRIPTION
    "NTCIP 1103 v03 - Original version of recording mechanism in SMIv1
     format."
  ::= {globalV2 2}
```

4.2 NODE DEFINITIONS

```
adminRecMechV2 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used to group administrative objects for the recMech feature.

     This node is intended to assist administrators in configuring access
     control. In general, administrator-level accounts can be given access to
     this node. In addition, it is envisioned that administrators may wish to
     grant owner-level accounts read-only access to the appropriate row
     within the adminRecMechTable so that owners can discover the amount of
     memory that they are allowed to access for recordings.

     <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.2"
  ::= {adminGlobalV2 2}

recMechV2Conformance OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used to contain conformance concepts for the rechMechV2 feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.127"
  ::= {recMechV2 127}

recMechV2Compliances OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used to contain compliance statements for the recMechV2 feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.127.1"
::= {recMechV2Conformance 1}

recMechV2Groups OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used to contain object groups for the recMechV2 feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.127.2"
::= {recMechV2Conformance 2}
```

4.3 ADMINISTRATIVE OBJECTS

4.3.1 Total Number of Recordings

```
adminRecMechV2NumRecordings  OBJECT-TYPE
   SYNTAX      ITSUnsigned16
   UNITS       "recordings"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The number of in-progress and completed recordings
       currently stored, regardless of adminRecMechV2Owner.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.2.1"
::= { adminRecMechV2 1 }
```

4.3.2 Recording Counter

```
adminRecMechV2RecordingCtr  OBJECT-TYPE
   SYNTAX      ITSCounter32
   UNITS       "recordings"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The number of recordings that have been completed since
       the last reboot, regardless of adminRecMechV2Owner.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.2.2"
::= { adminRecMechV2 2 }
```

4.3.3 Clear All

```
adminRecMechV2ClearAll  OBJECT-TYPE
   SYNTAX      ITSClear
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
      "<Definition> A control object that destroys all rows in the
       recMechV2OwnerTable, recMechV2ClassTable, recMechV2FactoryTable,
       recMechV2RecordingTable, and recMechV2SampleTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.2.3"
::= { adminRecMechV2 3 }
```

4.3.4 Clear All Classes

```
adminRecMechV2ClearAllClasses  OBJECT-TYPE
   SYNTAX      ITSClear
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
      "<Definition> A control object that destroys all rows in the
       recMechV2ClassTable, recMechV2FactoryTable, recMechV2RecordingTable,
       and recMechV2SampleTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.2.4"
::= { adminRecMechV2 4 }
```

4.3.5 Clear All Factories

```
adminRecMechV2ClearAllFactories  OBJECT-TYPE
   SYNTAX      ITSClear
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
```



```
      "<Definition> A control object that destroys all rows in the
       recMechV2FactoryTable, recMechV2RecordingTable, and
       recMechV2SampleTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.2.5"
::= { adminRecMechV2 5 }
```

4.3.6 Clear All Recordings

```
adminRecMechV2ClearAllRecordings  OBJECT-TYPE
   SYNTAX      ITSClear
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
      "<Definition> A control object that destroys all rows in the
       recMechV2RecordingTable, and recMechV2SampleTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.126.6.9.2.6"
::= { adminRecMechV2 6 }
```

4.4 OWNER TABLE

```
recMechV2OwnerTable  OBJECT-TYPE
   SYNTAX      SEQUENCE OF RecMechV2OwnerEntry
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> This table is used to allow owners to manage the
       recording mechanism and the recording tables that they own.
       <Table Type> managed
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.1"
::= { recMechV2 1 }

recMechV2OwnerEntry  OBJECT-TYPE
   SYNTAX      RecMechV2OwnerEntry
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> This object defines a row in the RecMechV2OwnerTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.1.1"
   AUGMENTS { fdOwnerEntry }
::= { recMechV2OwnerTable 1 }

RecMechV2OwnerEntry ::= SEQUENCE {
   recMechV2OwnerNumRecordings       ITSUnsigned16,
   recMechV2OwnerRecordingCtr        ITSCounter32,
   recMechV2OwnerClearAll            ITSClear,
   recMechV2OwnerClearAllClasses     ITSClear,
   recMechV2OwnerClearAllFactories   ITSClear,
   recMechV2OwnerClearAllRecordings  ITSClear }
```

4.4.1 Number of Recordings for Owner

```
recMechV2OwnerNumRecordings  OBJECT-TYPE
   SYNTAX      ITSUnsigned16
   UNITS       "recordings"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The number of in-progress and completed recordings
       currently stored, regardless of adminRecMechV2Owner.
```

```
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.1.1.1"
::= { recMechV2OwnerEntry 1 }
```

4.4.2 Recording Counter for Owner

```
recMechV2OwnerRecordingCtr  OBJECT-TYPE
   SYNTAX      ITSCounter32
   UNITS       "recordings"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> A counter that increments upon the completion of each
       recording associated with the adminRecMechV2Owner.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.1.1.2"
::= { recMechV2OwnerEntry 2 }
```

4.4.3 Clear All for Owner

```
recMechV2OwnerClearAll  OBJECT-TYPE
   SYNTAX      ITSClear
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
      "<Definition> A control object that destroys all rows associated with
       the adminRecMechV2Owner in the recMechV2ClassTable,
recMechV2FactoryTable,
       recMechV2RecordingTable, and recMechV2SampleTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.1.1.3"
::= { recMechV2OwnerEntry 3 }
```

4.4.4 Clear All Classes for Owner

```
recMechV2OwnerClearAllClasses  OBJECT-TYPE
   SYNTAX      ITSClear
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
      "<Definition> A control object that destroys all rows associated with
       the adminRecMechV2Owner in the recMechV2ClassTable,
recMechV2FactoryTable,
       recMechV2RecordingTable, and recMechV2SampleTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.1.1.4"
::= { recMechV2OwnerEntry 4 }
```

4.4.5 Clear All Factories for Owner

```
recMechV2OwnerClearAllFactories  OBJECT-TYPE
   SYNTAX      ITSClear
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
      "<Definition> A control object that destroys all rows associated with
       the adminRecMechV2Owner in the recMechV2FactoryTable,
       recMechV2RecordingTable, and recMechV2SampleTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.1.1.5"
::= { recMechV2OwnerEntry 5 }
```

4.4.6 Clear All Recordings for Owner

```
recMechV2OwnerClearAllRecordings  OBJECT-TYPE
```



```
   SYNTAX      ITSClear
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
      "<Definition> A control object that destroys all rows associated with
       the adminRecMechV2Owner in the recMechV2RecordingTable, and
       recMechV2SampleTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.1.1.6"
::= { recMechV2OwnerEntry 6 }
```

4.5 CLASS TABLE

```
recMechV2ClassTable  OBJECT-TYPE
   SYNTAX  SEQUENCE OF RecMechV2ClassEntry
   MAX-ACCESS  not-accessible
   STATUS  current
   DESCRIPTION
      "<Definition>This table is used to manage the recording mechanism and
       the recording tables.
       <TableType> managed
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2"
::= { recMechV2 2 }

recMechV2ClassEntry  OBJECT-TYPE
   SYNTAX  RecMechV2ClassEntry
   MAX-ACCESS  not-accessible
   STATUS  current
   DESCRIPTION
      "<Definition>This object defines a row in the Recording Mechanism
        Manger Table. Rows defined within this table allow recordings from
        multiple triggers to be grouped together into a single class for
        easier management.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1"
   INDEX { fdOwnerID, recMechV2ClassID }
::= { recMechV2ClassTable 1 }

RecMechV2ClassEntry ::= SEQUENCE {
   recMechV2ClassID                ITSPositive8,
   recMechV2ClassDescription       SnmpAdminString,
   recMechV2ClassSizeLimit         Unsigned32,
   recMechV2ClassClearDate         ITSDateStamp,
   recMechV2ClassClearTime         ITSDailyTimeStamp,
   recMechV2ClassNumRecordings     ITSUnsigned8,
   recMechV2ClassRecordingCtr      ITSCounter16,
   recMechV2ClassStorageType       StorageType,
   recMechV2ClassRowStatus         RowStatus }
```

4.5.1 Class Name

```
recMechV2ClassID  OBJECT-TYPE
   SYNTAX      ITSPositive8
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> A unique name for the recording class within the scope
       of the adminRecMechV2Owner. All recordings assigned to the class will
       be managed according to the other columns of this row.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.1"
```

```
::= { recMechV2ClassEntry 1 }
```

4.5.2 Class Description

```
recMechV2ClassDescription  OBJECT-TYPE
   SYNTAX      SnmpAdminString
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> A description of the recording class.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.2"
::= { recMechV2ClassEntry 2 }
```

4.5.3 Class Size Limit

```
recMechV2ClassSizeLimit  OBJECT-TYPE
   SYNTAX      Unsigned32
   UNITS       "octets"
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> This object specifies the maximum amount of memory to be
       used within the device to store recordings for the associated
       recording class. When the size limit is reached (e.g., due to a new
       recording or through the modification of this object) the oldest
       recording(s) within the class (based on the values of
       recMechV2RecordingTriggerDate and recMechV2RecordingTriggerTime) are
       bumped out of storage until the storage used for this recording class
       is less than or equal to the size defined by this object.

       An attempt to set this object to a value that would cause the sum of
       all recMechV2ClassSizeLimit objects for the associated
       adminRecMechV2Owner to exceed the adminRecMechSizeLimit for the
       adminRecMechV2Owner shall cause Step 10 of the set operation to fail,
       thereby resulting in an 'inconsistentVlaue' error.

       A recording that exceeds recMechV2ClassSizeLimit octets shall not be
       stored within the recording class.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.3"
::= { recMechV2ClassEntry 3 }
```

4.5.4 Class Clear Date

```
recMechV2ClassClearDate OBJECT-TYPE
   SYNTAX      ITSDateStamp
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> This object, along with recMechV2ClassClearTime, is used
       to clear multiple recordings from the recMechV2RecordingTable and
       recMechV2SampleTable. All completed recordings (and associated
samples)
       for this recording class that are associated with a
       recMechV2RecordingTriggerDate and recMechV2RecordingTriggerTime equal
to
       or older than the recMechV2ClassClearDate and recMechV2ClassClearTime
       shall be cleared from the recMechV2RecordingTable and
       recMechV2SampleTable.
       If the recMechV2ClassClearDate and recMechV2ClassClearTime refer to a
```



```
       future point in time, no recordings shall be initiated for the
       associated recording class.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.4"
::= { recMechV2ClassEntry 4 }
```

4.5.5 Class Clear Time

```
recMechV2ClassClearTime OBJECT-TYPE
   SYNTAX      ITSDailyTimeStamp
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> This object, along with recMechV2ClassClearDate, is used
       to clear multiple recordings from the recMechV2RecordingTable and
       recMechV2SampleTable. All completed recordings (and associated
samples)
       for this recording class that are associated with a
       recMechV2RecordingTriggerDate and recMechV2RecordingTriggerTime equal
to
       or older than the recMechV2ClassClearDate and recMechV2ClassClearTime
       shall be cleared from the recMechV2RecordingTable and
       recMechV2SampleTable.
       If the recMechV2ClassClearDate and recMechV2ClassClearTime refer to a
       future point in time, no recordings shall be initiated for the
       associated recording class.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.5"
::= { recMechV2ClassEntry 5 }
```

4.5.6 Class Number of Recordings

```
recMechV2ClassNumRecordings  OBJECT-TYPE
   SYNTAX      ITSUnsigned8
   UNITS       "recording"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The number of recordings for this recording class that
       are currently stored in the recMechV2RecordingTable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.6"
::= { recMechV2ClassEntry 6 }
```

4.5.7 Class Recording Counter

```
recMechV2ClassRecordingCtr  OBJECT-TYPE
   SYNTAX      ITSCounter16
   UNITS       "recording"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> This object is a counter that gets incremented
        every time a recording occurs for this class.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.7"
::= { recMechV2ClassEntry 7 }
```

4.5.8 Class Storage Type

```
recMechV2ClassStorageType  OBJECT-TYPE
   SYNTAX      StorageType
   MAX-ACCESS  read-create
```

```
   STATUS      current
   DESCRIPTION
      "<Definition> This object defines the storage type to use for this
       row of the recMechV2ClassTable. Rows having the value 'permanent'
       shall allow writable access to recMechV2ClassSizeLimit,
       recMechV2ClassClearDate, and recMechV2ClassClearTime.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.8"
::= { recMechV2ClassEntry 8 }
```

4.5.9 Class Row Status

```
recMechV2ClassRowStatus  OBJECT-TYPE
   SYNTAX      RowStatus
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> This object indicates the status of this row within the
       recMechV2ClassTable. Objects within this row can be modified
       regardless of the state of this object.

       Within the second phase of the set operation to set this object to
       'destroy', all rows within the recMechV2FactoryTable,
       recMechV2RecordingTable, and recMechV2SampleTable that use this
       recMechV2ClassName as an index shall also be destroyed.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.2.1.9"
::= { recMechV2ClassEntry 9 }
```

4.6 RECORDING FACTORY

```
recMechV2FactoryTable  OBJECT-TYPE
   SYNTAX  SEQUENCE OF RecMechV2FactoryEntry
   MAX-ACCESS  not-accessible
   STATUS  current
   DESCRIPTION
      "<Definition> This table is used to configure the information to be
       contained within each recording.
       <Table Type> managed
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3"
::= { recMechV2 3 }

recMechV2FactoryEntry  OBJECT-TYPE
   SYNTAX  RecMechV2FactoryEntry
   MAX-ACCESS  not-accessible
   STATUS  current
   DESCRIPTION
      "<Definition> This object defines a row in the Recording Mechanism
        Factory Table. Rows defined within this table define the
        information the device shall record when a recording is initiated.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1"
   INDEX { fdOwnerID, recMechV2ClassID, recMechV2FactoryID }
::= { recMechV2FactoryTable 1 }

RecMechV2FactoryEntry ::= SEQUENCE {
   recMechV2FactoryID               ITSPositive8,
   recMechV2FactorySecurityModel    SnmpSecurityModel,
   recMechV2FactorySecurityLevel    SnmpSecurityLevel,
   recMechV2FactorySecurityName     SnmpAdminString,
   recMechV2FactorySampleContext    SnmpAdminString,
```



```
   recMechV2FactorySampleOID        VariablePointer,
   recMechV2FactoryPreSamples       NtcipPercent,
   recMechV2FactorySamplePeriod     ITSUnsigned16,
   recMechV2FactoryMonitorContext   SnmpAdminString,
   recMechV2FactoryMonitorOID       VariablePointer,
   recMechV2FactorySampleLimit      ITSUnsigned8,
   recMechV2FactoryStorageType      StorageType,
   recMechV2FactoryRowStatus        RowStatus }
```

4.6.1 Recording Factory Name

```
recMechV2FactoryID  OBJECT-TYPE
   SYNTAX      ITSPositive8
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> This object provides a unique name for the recording
       factory within the context of the adminRecMechV2Owner.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.1"
::= { recMechV2FactoryEntry 1 }
```

4.6.2 Recording Factory Security Model

```
recMechV2FactorySecurityModel  OBJECT-TYPE
   SYNTAX      SnmpSecurityModel
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> The security model used to activate this row of the
       recMechV2FactoryTable and that is used to access the
       recMechV2FactoryObjectID and recMechV2FactorySampleOID.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.2"
::= { recMechV2FactoryEntry 2 }
```

4.6.3 Recording Factory Security Level

```
recMechV2FactorySecurityLevel  OBJECT-TYPE
   SYNTAX      SnmpSecurityLevel
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> The security level used to activate this row of the
       recMechV2FactoryTable and that is used to access the
       recMechV2FactoryObjectID and recMechV2FactorySampleOID.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.3"
::= { recMechV2FactoryEntry 3 }
```

4.6.4 Recording Factory Security Name

```
recMechV2FactorySecurityName  OBJECT-TYPE
   SYNTAX      SnmpAdminString (SIZE (0..32))
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> The security name used to activate this row of the
       recMechV2FactoryTable and that is used to access the
       recMechV2FactoryObjectID and recMechV2FactorySampleOID.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.4"
::= { recMechV2FactoryEntry 4 }
```

4.6.5 Context for Object to be Sampled

```
recMechV2FactorySampleContext  OBJECT-TYPE
   SYNTAX      SnmpAdminString (SIZE (0..32))
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> The management context of the object instance to be
       sampled for the recording. For example, the command responder might
       provide access to multiple MIB instances; this object disambiguates
       which MIB instance is to be used.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.5"
::= { recMechV2FactoryEntry 5 }
```

4.6.6 Object to be Sampled in Recording

```
recMechV2FactorySampleOID  OBJECT-TYPE
   SYNTAX      VariablePointer
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> The object identifier of the object instance to be
       sampled for the recording. The object instance shall be accessed using
       the security parameters as specified in recMechV2FactorySecurityModel,
       recMechV2FactorySecurityLevel, recMechV2FactorySecurityName, and
       recMechV2FactorySampleContext. (See RFC 3415 for details about how the
       view-based access control model uses these parameters.)

       An attempt to set this object to a value that is not within the read
       mib view for the supplied security parameters shall result in a
       failure of Step 10 of the set operation, thereby resulting in an
       'inconsistent value' error. NOTE: The object instance does not have to
       exist at the time that this object is set for the set to be
       successful.

       If the object identifier does not point to a valid object instance
       within the read mib view for the indicated security parameters at the
       time that the data is being sampled, the sample shall record a zero-
       length string.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.6"
::= { recMechV2FactoryEntry 6 }
```

4.6.7 Pre-Trigger Samples

```
recMechV2FactoryPreSamples  OBJECT-TYPE
   SYNTAX      ITSPercent
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> This object indicates the percentage of samples in a
       recording that are intended to be captured prior to the recording
       being triggered. When this value is greater than zero (0), the device
       will need to collect and buffer samples prior to the recording being
       triggered so that it can store the requisite information when a
       recording is triggered.

       <Format>
       A value of zero (0) means that the recording shall start with the
```



```
       first sample captured after the recording is triggered. A value of 100
       means that the recording will end with the last sample captured prior
       to or simultaneous with the recording being triggered, where
       simultaneous is defined as occurring within
       recMechV2FactorySamplePeriod / 2.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.7"
::= { recMechV2FactoryEntry 7 }
```

4.6.8 Sample Period

```
recMechV2FactorySamplePeriod  OBJECT-TYPE
   SYNTAX      ITSUnsigned16
   UNITS       "0.1 milliseconds"
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> This object indicates the frequency at which samples
       shall be taken. Allowable sample periods are restricted to a value of
       zero (0) or
       recMechMinSamplePeriod + (recMechV2SamplePeriodResolution * n)

       where  n is an integer,
       0 <= n, and
       n <= (recMechMaxSamplePeriod - recMechMinSamplePeriod) /
       recMechV2SamplePeriodResolution

       If the value of this object is zero (0), samples shall be collected
       whenever the value of the object specified by
       recMechV2FactoryMonitorOID changes (i.e., based on an on-change event
       rather than based on time).

       If the value of this object is greater than zero, this object
       indicates the time that shall elapse between each sample collected
       for the recording produced by this factory and the value of
       recMechV2FactoryMonitorOID shall be ignored.

       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.8"
::= { recMechV2FactoryEntry 8 }
```

4.6.9 Context for Object to be Monitored

```
recMechV2FactoryMonitorContext  OBJECT-TYPE
   SYNTAX      SnmpAdminString (SIZE (0..32))
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> The management context to use when accessing
       recMechV2FactoryMonitorOID.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.9"
::= { recMechV2FactoryEntry 9 }
```

4.6.10 Object to be Monitored for Change Events

```
recMechV2FactoryMonitorOID  OBJECT-TYPE
   SYNTAX      VariablePointer
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> The object identifier of the object instance to be
```

```
       monitored for on-change events when recMechV2FactorySamplePeriod is
       zero (0).

       If recMechV2FactorySamplePeriod is non-zero, the value of this object
       shall be ignored.

       The object instance shall be accessed using the security parameters as
       specified in recMechV2FactorySecurityModel,
recMechV2FactorySecurityLevel,
       recMechV2FactorySecurityName, and recMechV2FactoryMonitorContext. An
       attempt to set this object to a value, other than zeroDotZero, that is
       not within the read mib view for the supplied security parameters
       shall result in a failure of Step 10 of the set operation, thereby
       resulting in an 'inconsistent value' error. NOTE: The object instance
       does not have to exist at the time that this object is set for the set
       to be successful.

       If this object does not point to a valid object instance within the
       read mib view for the indicated security parameters at the time that
       the data is being monitored, the on-change event shall not fire.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.10"
::= { recMechV2FactoryEntry 10 }
```

4.6.11 Number of Samples to Collect

```
recMechV2FactorySampleLimit  OBJECT-TYPE
   SYNTAX      ITSUnsigned16
   UNITS       "samples"
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> The intended number of samples to be included for
       recordings produced by this factory. A recording that collects its
       full number of samples prior to and after the recording trigger shall
       have a recording of this number of samples.
       If this object is zero (0), then no recordings are created
       based on this configuration.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.11"
::= { recMechV2FactoryEntry 11 }
```

4.6.12 Recording Factory Storage Type

```
recMechV2FactoryStorageType  OBJECT-TYPE
   SYNTAX      StorageType
   MAX-ACCESS  read-create
   STATUS      current
   DESCRIPTION
      "<Definition> This object defines the storage type to use for this row
       within the recMechV2FactoryTable. Permanent rows do not have to be
       editable.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.12"
::= { recMechV2FactoryEntry 12 }
```

4.6.13 Recording Factory Row Status

```
recMechV2FactoryRowStatus  OBJECT-TYPE
   SYNTAX      RowStatus
   MAX-ACCESS  read-create
   STATUS      current
```



```
   DESCRIPTION
      "<Definition> This object indicates the status of this row within the
       recMechV2FactoryTable. Objects within this row cannot be modified
       while this object is 'active'.
       Within the second phase of the set operation to set this object to
       'destroy', all rows within the recMechV2RecordingTable, and
       recMechV2SampleTable with an index that includes both
recMechV2ClassName
       and recMechV2FactoryName shall also be destroyed.

       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.3.1.13"
::= { recMechV2FactoryEntry 13 }
```

4.7 RECORDING TABLE

```
recMechV2RecordingTable  OBJECT-TYPE
   SYNTAX      SEQUENCE OF RecMechV2RecordingEntry
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> This table is used to provide information about each
       recording completed or in progress.
       <Table Type> managed
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4"
::= { recMechV2 4 }

recMechV2RecordingEntry  OBJECT-TYPE
   SYNTAX  RecMechV2RecordingEntry
   MAX-ACCESS  not-accessible
   STATUS  current
   DESCRIPTION
      "<Definition> This object defines a row in the recMechV2RecordingTable.
       Rows defined within this table capture information that applies to all
       samples collected within the recording.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1"
   INDEX {fdOwnerID, recMechV2ClassID, recMechV2RecordingID}
::= { recMechV2RecordingTable 1 }

RecMechV2RecordingEntry ::= SEQUENCE {
   recMechV2RecordingID             ITSPositive8,
   recMechV2RecordingTrigger        RowPointer,
   recMechV2RecordingTriggerDate    ITSDateStamp,
   recMechV2RecordingTriggerTime    ITSDailyTimeStamp,
   recMechV2RecordingStatus         INTEGER,
   recMechV2RecordingTriggerPoint   ITSUnsigned16,
   recMechV2RecordingNumSamples     ITSUnsigend16,
   recMechV2RecordingDelete         TruthValue }
```

4.7.1 Recording Index

```
recMechV2RecordingID  OBJECT-TYPE
   SYNTAX      ITSPositive8
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> The recording number within this class for this
        recording. Recording numbers shall be assigned starting at 1 and
        increasing sequentially by 1 with each new recording within the
```

```
        class. After the assigned value reaches 255, the process shall
        repeat, skipping the value 0. Recordings shall be assigned the index
        value in order of the trigger events occurring ensuring that the
        trigger date/time values are chronological (except for the rollover).
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1.1"
::= { recMechV2RecordingEntry 1 }
```

4.7.2 Recording Trigger

```
recMechV2RecordingTrigger  OBJECT-TYPE
   SYNTAX      RowPointer
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> This object indicates the trigger or event that caused
       the recording to be initiated. For example, if the recording was
       triggered by a conditional trigger, this object would indicate the OID
       of fdCondTriggerDescription for the row within the fdCondTriggerTable
       that caused the recording.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1.2"
::= { recMechV2RecordingEntry 2 }
```

4.7.3 Recording Trigger Date

```
recMechV2RecordingTriggerDate  OBJECT-TYPE
   SYNTAX      ITSDateStamp
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> This object contains the date on which the recording was
       triggered.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1.3"
::= { recMechV2RecordingEntry 3 }
```

4.7.4 Recording Trigger Time

```
recMechV2RecordingTriggerTime  OBJECT-TYPE
   SYNTAX      ITSDailyTimeStamp
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> This object indicates the time at which the recording was
       triggered.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1.4"
::= { recMechV2RecordingEntry 4 }
```

4.7.5 Recording Status

```
recMechV2RecordingStatus  OBJECT-TYPE
   SYNTAX      INTEGER { available (1),
                         pretrigger (2),
                         triggered (3),
                         complete (4)}
   MAX-ACCESS  read-only
   STATUS      current
   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.
                  Note: This state is only applicable to implementations that
                  have rows that can exist without any samples.
       pretrigger The recording is collecting pre-trigger samples.
                  Note: When the trigger point is zero (0) percent or the
                  trigger condition is already satisfied when the
                  recMechV2FactoryEntry is configured, this state does not
                  occur.
       triggered  The recording has been triggered and is currently
                  collecting post event records. Triggered recordings shall
                  survive a power outage.
                  Upon power up, any recording in the 'triggered' state shall
                  automatically transition to the 'complete' state.
                  Note: When the trigger point is 100 percent, this state
                  does not occur.
       complete   The recording is complete (e.g., all post-event samples
                  have been collected). Completed recordings shall survive a
                  power outage.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1.5"
::= { recMechV2RecordingEntry 5 }
```

4.7.6 Recording Trigger Point

```
recMechV2RecordingTriggerSample  OBJECT-TYPE
   SYNTAX      ITSUnsigned16
   UNITS       "samples"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> This object indicates the number (recMechV2SampleNumber)
       of the first post-event sample in the recording.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1.6"
::= { recMechV2RecordingEntry 6 }
```

4.7.7 Number of Samples in Recording

```
recMechV2RecordingNumSamples  OBJECT-TYPE
   SYNTAX      ITSUnsigned16
   UNITS       "samples"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The current number of samples contained in this
       recording. A completed recording does not necessarily have exactly
       recMechV2FactoryNumSamples samples recorded (e.g., due to changes in
       configuration, errors in collecting samples, etc.).
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1.7"
::= { recMechV2RecordingEntry 7 }
```

4.7.8 Delete Recording

```
recMechV2RecordingDelete  OBJECT-TYPE
   SYNTAX      TruthValue
   MAX-ACCESS  read-write
   STATUS      current
   DESCRIPTION
```

```
      "<Definition> A control that allows the deletion of the recording. The
       object shall always report false when read. Setting this object to
       true shall delete this row in the table and shall also delete all
       rows in the recMechV2SampleTable with the same fdOwnerID,
       recMechV2ClassID, and recMechV2RecordingID.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.4.1.8"
::= { recMechV2RecordingEntry 8 }
```

4.8 SAMPLE TABLE

```
recMechV2SampleTable  OBJECT-TYPE
   SYNTAX      SEQUENCE OF RecMechV2SampleEntry
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> A table containing the samples for each recording.
       <Table Type> managed
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.5"
::= { recMechV2 5 }

recMechV2SampleEntry  OBJECT-TYPE
   SYNTAX      RecMechV2SampleEntry
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> An entry in the recording mechanism sample table.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.5.1"
INDEX { fdOwnerID, recMechV2ClassID, recMechV2RecordingID, recMechV2SampleID}
::= { recMechV2SampleTable  1 }

RecMechV2SampleEntry::= SEQUENCE {
   recMechV2SampleID    ITSPositive16,
   recMechV2SampleTime   ITSDailyTimeStamp,
   recMechV2SampleValue  ITSOerString }
```

4.8.1 Sample Number

```
recMechV2SampleID  OBJECT-TYPE
   SYNTAX      ITSPositive16
   MAX-ACCESS  not-accessible
   STATUS      current
   DESCRIPTION
      "<Definition> The number used to uniquely identify this sample within
       this recording. Sample numbers shall be sequentially assigned starting
       at 1 and increasing by 1 until the configured number of post-event
       samples have been captured or the recording is otherwise completed.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.5.1.1"
::= { recMechV2SampleEntry 1 }
```

4.8.2 Sample Time

```
recMechV2SampleTime  OBJECT-TYPE
   SYNTAX      ITSDailyTimeStamp
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The time that the sample was captured. The entry shall be
        collected within one recMechV2SamplePeriodResolution unit of time
from
        the intended time and timestamped with the time of collection.
```



```
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.5.1.2"
::= { recMechV2SampleEntry 2 }
```

4.8.3 Sample Value

```
recMechV2SampleValue  OBJECT-TYPE
   SYNTAX      ITSOerString
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The OER-encoded value of the object instance referenced
       by recMechV2FactorySampleOID at the time when the sample was
       collected.

       <Format>
       The value shall not contain any padding characters either before or
       after the values. Decoding the value of this object requires
       knowledge of the syntax of the object referenced by
       recMechV2FactorySampleOID.

       For example, the value portion of the variableBinding encoding of
       this object could be 0x04 02 01 41, where:
       04 - octet string type for ITSOerString
       02 - length of 2 octets
       01 41 - sampled value of object referenced by
               recMechV2FactorySampleOID

       The exact meaning of 01 41 is based on the syntax of
       recMechV2FactorySampleOID; for example, the following table gives
       possible valid decodings depending on the syntax:
       Syntax           Decoding  Logic
       ITSInteger16     321       fixed-length two-octet integer encoding
       Counter64        65        one-octet length and one-octet value
       SnmpAdminString  'A'       one-octet length and one-octet value

       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.5.1.3"
::= { recMechV2SampleEntry 3 }
```

4.9 CAPABILITY OBJECTS

4.9.1 Minimum Sample Period

```
recMechV2MinSamplePeriod  OBJECT-TYPE
   SYNTAX      ITSPositive16
   UNITS       "0.1 milliseconds"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The minimum sample period for recordings supported by the
       device.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.6"
::= { recMechV2 6}
```

4.9.2 Maximum Sample Period

```
recMechV2MaxSamplePeriod  OBJECT-TYPE
   SYNTAX      ITSPositive16
   UNITS       "0.1 milliseconds"
   MAX-ACCESS  read-only
```

```
   STATUS      current
   DESCRIPTION
      "<Definition> The maximum sample period for recordings supported by the
        device.
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.7"
::= { recMechV2 7}
```

4.9.3 Sample Period Resolution

```
recMechV2SamplePeriodResolution  OBJECT-TYPE
   SYNTAX      Unsigned32 (1..50)
   UNITS       "0.1 milliseconds"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "<Definition> The sample period resolution for recordings supported
        by the device. Allowable sample periods are restricted to
        (recMinSamplePeriod + recSamplePeriodResolution * n), where:
        n is an integer,
        0 <= n, and
        n <= (recMaxSamplePeriod - recMinSamplePeriod) /
              recSamplePeriodResolution
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.8"
::= { recMechV2 8}
```

4.10 COMPLIANCE GROUPS

4.10.1 Recording Mechanism Group

```
recMechV2GroupR1 OBJECT-GROUP
  OBJECTS { adminRecMechV2SizeLimit,
            adminRecMechV2ClearAll,
            adminRecMechV2ClearAllFactories,
            adminRecMechV2ClearAllRecordings,
            adminRecMechV2RowStatus,
            adminRecMechV2NumRecordings,
            adminRecMechV2RecordingCtr,
            recMechV2MinSamplePeriod,
            recMechV2MaxSamplePeriod,
            recMechV2SamplePeriodResolution,
            recMechV2OwnerClearAll,
            recMechV2OwnerClearAllFactories,
            recMechV2OwnerClearAllRecordings,
            recMechV2OwnerNumRecordings,
            recMechV2OwnerRecordingCtr,
            recMechV2ClassDescription,
            recMechV2ClassSizeLimit,
            recMechV2ClassClearDate,
            recMechV2ClassClearTime,
            recMechV2ClassStorageType,
            recMechV2ClassRowStatus,
            recMechV2ClassNumRecordings,
            recMechV2ClassRecordingCtr,
            recMechV2FactorySampleContext,
            recMechV2FactorySampleOID,
            recMechV2FactoryPreSamples,
            recMechV2FactorySamplePeriod,
            recMechV2FactoryMonitorContext,
            recMechV2FactoryMonitorOID,
```



```
            recMechV2FactorySampleLimit,
            recMechV2FactoryStorageType,
            recMechV2FactoryRowStatus,
            recMechV2FactorySecurityModel,
            recMechV2FactorySecurityLevel,
            recMechV2FactorySecurityName,
            recMechV2RecordingDelete,
            recMechV2RecordingTrigger,
            recMechV2RecordingTriggerDate,
            recMechV2RecordingTriggerTime,
            recMechV2RecordingStatus,
            recMechV2RecordingTriggerPoint,
            recMechV2RecordingNumSamples,
            recMechV2SampleTime,
            recMechV2SampleValue }
  STATUS     current
  DESCRIPTION
    "The objects necessary for the recording mechanism version 2.
     NOTE: The recording mechanism is dependent upon the fdOwnerTable, as
     defined in ISO 20684-7. It is designed to be called by
     the ISO 20684-3 Action Table but could be called by other
     means as well.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.9.2.127.2.1"
  ::= {recMechV2Groups 1}


END -- NTCIP1201-RechMechV2
```

Section 5
DEPRECATED GLOBAL MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

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

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1201 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP 1201 for the objects contained within this MIB.

Table 3: Compliance Statement for NTCIP1201-GlobalV1 MIB

Group	v01	v01A1	v02	v03
globalConfigIDGroupR1	O	O	O	O
globalModuleGroupR1	O [1]	O [1]	O	O
globalBaseStandardsGroupR1			O	O
globalDbMgmtGroupR1	O [2, 3]			
globalDbMgmtGroupR2		O [2]	O	O
globalTimeMgmtUtcGroupR1	O [2]	O [2]	O	O
globalTimeMgmtDstGroupR1	O [2, 4]	O [2, 4]		
globalTimeMgmtDstGroupR2			O	O
globalTimeMgmtLocalGroupR1			O	O
globalTimeBaseEventGroupR1	O [2]	O [2]	O	O
globalTimeBaseEventGroupR1Ext				O
globalPmppGroupR1	O [3]	O [3]		
globalPmppGroupR2			O	O

[1] moduleVersion — Implementations are not required to support the format defined for the text string
[2] Implementations are not required to implement the defined default values for any object
[3] Interoperability issues were discovered with the original design
[4] globalDaylightSaving — Implementations are only required to support the disableDST and enableUSDST values

5.1 NTCIP HEADER

```
NTCIP1201-GlobalV1 DEFINITIONS ::= BEGIN
IMPORTS
   MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Integer32, Unsigned32
                                                    FROM SNMPv2-SMI
                                                      -- RFC 2578
   AutonomousPointer
                                                    FROM SNMPv2-TC
                                                      -- RFC 2579
   global
                                                    FROM NTCIP1201-Global
   profiles
                                                    FROM NTCIP8004-Transportation

globalV1 MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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

     Copyright (C) 2010, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.
```

```
                    Joint NEMA, AASHTO, and ITE
                  NTCIP Management Information Base
                        DISTRIBUTION NOTICE

    To the extent and in the limited event these materials are distributed
    by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
    module, AASHTO/ITE/NEMA extends the permission for users to make and/or
    distribute unlimited copies of the unedited MIB module, including copies
    for commercial distribution, provided that each copy made and/or
    distributed contains this Notice.

    The MIB module may be distributed with a companion MIB module that
    contains one or more invocations of the AGENT-CAPABILITIES macro that
    specifies the capabilities of a specific implementation. By convention,
    such a companion MIB module might have the same module name followed by
    "-", followed the enterprise's Internet Assigned Number Authority
    (IANA)-assigned enterprise number.

    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 OR 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, OR 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 you.

    Use of these materials does not constitute an endorsement or affiliation
    by or between AASHTO, ITE, or NEMA and you, your company, or your
    products and services.

    NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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)."
  ::= { global 8 }
```

5.2 OBJECT IDENTITIES

5.2.1 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 }

globalConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define conformance
      information for the global MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127"
::= { globalV1 127 }

globalCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define compliance
      statements for the global MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.1"
::= { globalConformance 1 }

globalGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define object groups for
      the global MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2"
::= { globalConformance 2 }
```

5.3 OBJECTS

5.3.1 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 20684-2)
      <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.1.1"
   ::= { globalConfiguration 1}
```

5.3.2 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}
```

5.3.3 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.
      <TableType> 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     OBJECT IDENTIFIER,
   moduleMake           OCTET STRING,
   moduleModel          OCTET STRING,
   moduleVersion        OCTET STRING,
   moduleType           INTEGER }
```

5.3.3.1 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 }
```

5.3.3.2 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.
      <Superseded 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 }
```

5.3.3.3 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.
      <Superseded 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 }
```

5.3.3.4 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.
      <Superseded 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 }
```

5.3.3.5 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.
      <Superseded 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 }
```


5.3.3.6 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 }
```

5.3.4 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

      <Superseded 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 }
```

5.4 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 }
--
```

5.4.1 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)			
		transaction	*verify*	*normal*	*done*
CURRENT STATE	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).
      <Superseded by> NTCIP1201v04-DB-MGMT.dbCreateTransactionV2
      <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 }
```

5.4.2 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 12012 v02
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.2"
::= { globalDBManagement 2 }
```

5.4.3 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 12012 v02
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.3"
::= { globalDBManagement 3 }
```

5.4.4 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 12012 v02
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.4"
::= { globalDBManagement 4 }
```

5.4.5 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 12012 v02
<Object Identifier> 1.3.6.1.4.1.1206.4.2.6.2.5"
::= { globalDBManagement  5 }
```

5.4.6 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.
      <Superseded by> NTCIP1201v04-DB-MGMT.dbVerifyStatusV2
      <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 }
```

5.4.7 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.
      <Superseded by> NTCIP1201v04-DB-MGMT.dbVerifyErrorV2
      <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 }
```

5.5 GLOBAL TIME MANAGEMENT NODE

```
globalTimeManagement   OBJECT-IDENTITY
  STATUS  current
  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 }
```

5.5.1 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> CLOCK-MIB.fdClockUtcDate & CLOCK-
      MIB.fdClockUtcTime (ISO 20684-7)
      <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}
```

5.5.2 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.
     disableDST -  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)

NOTE: 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 }
```

5.6 TIMEBASE EVENT SCHEDULER NODE

```
timebase OBJECT-IDENTITY
  STATUS  current
  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 }
```

5.6.1 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 }
```

5.6.2 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)
      <TableType> static
      <Superseded by> DAY-PLAN-MIB.fdDayPlanScheduleTable (ISO 20684-3)
      <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> DAY-PLAN-MIB.fdDayPlanScheduleEntry (ISO 20684-3)
      <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     INTEGER,
            timeBaseScheduleMonth      INTEGER,
            timeBaseScheduleDay        INTEGER,
            timeBaseScheduleDate       INTEGER,
            timeBaseScheduleDayPlan    INTEGER }
```

5.6.2.1 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanScheduleNumber (ISO 20684-
      3)
      <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 }
```

5.6.2.2 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanScheduleMonth (ISO 20684-3)
      <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 }
```

5.6.2.3 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanScheduleDayOfWeek (ISO
      20684-3)
      <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 }
```

5.6.2.4 Time Base Schedule Date Parameter

```
timeBaseScheduleDate   OBJECT-TYPE
SYNTAX      Unsigend32
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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanScheduleDayOfMonth (ISO
      20684-3)
      <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 }
```

5.6.2.5 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanScheduleDayPlan (ISO 20684-
      3)
      <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 }
```

5.6.3 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 }
```

5.6.4 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 }
```

5.6.5 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.
      <TableType> static
      <Superseded by> DAY-PLAN-MIB.fdDayPlanTable & DAY-PLAN-
      MIB.fdDayPlanTriggerTable (ISO 20684-3)
      <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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanEntry & DAY-PLAN-
      MIB.fdDayPlanTriggerEntry
      <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 OBJECT IDENTIFIER }
```

5.6.5.1 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanNumber (ISO 20684-3)
      <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 }
```

5.6.5.2 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanTriggerTime (ISO 20684-3)
      <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 }
```

5.6.5.3 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanTriggerTime (ISO 20684-3)
      <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 }
```

5.6.5.4 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanTriggerTime (ISO 20684-3)
      <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 }
```

5.6.5.5 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 transitionary 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanTriggerOwner & DAY-PLAN-
      MIB.fdDayPlanTriggerName (ISO 20684-3)
      <Informative> The replacement objects provide the two indicies
      necessary to identify a unique row into the fdActionTable.
      <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.5.1.5"
DEFVAL      { zeroDotZero }
::= { timeBaseDayPlanEntry 5 }
```

5.6.6 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.
      <Superseded by> DAY-PLAN-MIB.fdDayPlanSchedulerCurrentDayPlan
      (ISO 20684-3)
      <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.6"
::= { timebase 6 }
```

5.6.7 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.
       <Superseded by> DAY-PLAN-MIB.fdDayPlanSchedulerSelectedRule (ISO
       20684-3)
  <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.3.3.7"
::={timebase 7}
```

5.6.8 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.
      <Superseded 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 }
```

5.6.9 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> CLOCK-MIB.fdClockLocalStandardTimeZone (ISO
      20684-7)
      <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 }
```

5.6.10 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> CLOCK-MIB.fdClockLocalDate & CLOCK-
      MIB.fdClockLocalTime (ISO 20684-7)
      <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 }
```

5.7 DAYLIGHT SAVING TIME (DST) NODE

```
daylightSavingNode   OBJECT-IDENTITY
  STATUS  deprecated
  DESCRIPTION
    "This node is an identifier used to organize all objects for support of
     defining DST.  This function is common to most device types.  The
     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 }
```

5.7.1 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> CLOCK-MIB.fdClockDstMaxEntries (ISO 20684-7)
      <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 }
```

5.7.2 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.
       <TableType> static
      <Superseded by> CLOCK-MIB.fdClockDstTable (ISO 20684-7)
      <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> CLOCK-MIB.fdClockDstEntry (ISO 20684-7)
      <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 }
```

5.7.2.1 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> CLOCK-MIB.fdClockDstIndex (ISO 20684-7)
       <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 }
```

5.7.2.2 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,
      dstBeginOccurences, dstBeginDayOfWeek, dstBeginDayOfMonth,
      dstEndOccurances, 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> CLOCK-MIB.fdClockDstBeginMonth (ISO 20684-7)
     <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 }
```

5.7.2.3 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.

      NOTE: 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).
     <Superseded by> CLOCK-MIB.fdClockDstBeginOccurrences (ISO 20684-7)
     <Informative> 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 }
```

5.7.2.4 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> CLOCK-MIB.fdClockDstBeginDayOfWeek (ISO 20684-7)
     <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 }
```

5.7.2.5 Daylight Saving Time (DST) Beginning Day of Month Parameter

```
dstBeginDayOfMonth      OBJECT-TYPE
SYNTAX      Integer32 (1..31)
UNITS       "days"
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.
     <Unit> day of month
     <Superseded by> CLOCK-MIB.fdClockDstBeginDayOfMonth (ISO 20684-7)
     <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 }
```

5.7.2.6 Daylight Saving Time (DST) Beginning Seconds to Transition Parameter

```
dstBeginSecondsToTransition    OBJECT-TYPE
SYNTAX      Unsigned32
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).

      NOTE:  a set of parameters that causes a day
      transition that crosses the midnight boundary may result in
      unexpected behavior.
     <Superseded by> CLOCK-MIB.fdClockDstBeginTime (ISO 20684-7)
     <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 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 }
```

5.7.2.7 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> CLOCK-MIB.fdClockDstEndMonth (ISO 20684-7)
     <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 }
```

5.7.2.8 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.

       NOTE: 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).
     <Superseded by> CLOCK-MIB.fdClockDstEndOccurrences (ISO 20684-7)
     <Informative> 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 }
```

5.7.2.9 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> CLOCK-MIB.fdClockDstEndDayOfWeek (ISO 20684-7)
     <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 }
```

5.7.2.10 Daylight Saving Time (DST) Ending Day of Month Parameter

```
dstEndDayOfMonth  OBJECT-TYPE
SYNTAX      Integer32(1..31)
UNITS       "days"
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> CLOCK-MIB.fdClockDstEndDayOfMonth (ISO 20684-7)
     <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 }
```

5.7.2.11 Daylight Saving Time (DST) Ending Seconds to Transition Parameter

```
dstEndSecondsToTransition    OBJECT-TYPE
SYNTAX      Unsigned32
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).

      NOTE: a set of parameters that causes a day
      transition that crosses the midnight boundary may result in
      unexpected behavior.
     <Superseded by> CLOCK-MIB.fdClockDstEndTime (ISO 20684-7)
     <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 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 }
```

5.7.2.12 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> CLOCK-MIB.fdClockDstOffset (ISO 20684-7)
     <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 }
```

5.8 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 }
```

5.8.1 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 }
```

5.8.2 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.
      <TableType> 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 }
```

5.8.2.1 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}
```

5.8.2.2 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}
```

5.8.2.3 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.

      NOTE that in PMPP all group addresses are encoded in one byte.
```



```
      <Informative> The PMPP protocol has been deprecated and there is
      no replacement object.
      <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.3.2.1.3"
REFERENCE
"NTCIP 2101"
DEFVAL  { 0 }
::= { hdlcGroupAddressEntry 3}
```

5.9 COMPLIANCE GROUPS

5.9.1 Global Configuration Identifier Group

```
globalConfigIDGroup OBJECT-GROUP
  OBJECTS { globalSetIDParameter }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for monitoring the configuration of
      the device.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.1"
  ::= {globalGroups 1}
```

5.9.2 Global Module Group

```
globalModuleGroup OBJECT-GROUP
  OBJECTS { globalMaxModules,
            moduleNumber,
            moduleDeviceNode,
            moduleMake,
            moduleModel,
            moduleVersion,
            moduleType }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for determining the modules contained
      in the device.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.2"
  ::= {globalGroups 2}
```

5.9.3 Global Base Standards Group

```
globalBaseStandardsGroup OBJECT-GROUP
  OBJECTS { controllerBaseStandards }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for determining the standards
      supported by the device.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.3"
  ::= {globalGroups 3}
```

5.9.4 Global Database Management Group

```
globalDbMgmtV1GroupR1 OBJECT-GROUP
  OBJECTS { dbCreateTransaction,
            dbErrorType,
            dbErrorID,
            dbTransactionID,
            dbMakeID }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the original database
```

```
      management logic for the device.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.4"
  ::= {globalGroups 4}
```

5.9.5 Global Database Management Group Revision 2

```
globalDbMgmtV1GroupR2 OBJECT-GROUP
  OBJECTS { dbCreateTransaction,
            dbVerifyStatus,
            dbVerifyError }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the database transaction
      feature of the device as revised in Amendment 1.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.5"
  ::= {globalGroups 5}
```

5.9.6 Global Time Management UTC Group

```
globalTimeMgmtUtcGroup OBJECT-GROUP
  OBJECTS { globalTime }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing UTC time.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.6"
  ::= {globalGroups 6}
```

5.9.7 Global Daylight Saving Time Group

```
globalTimeMgmtDstGroupR1 OBJECT-GROUP
  OBJECTS { globalDaylightSaving,
            globalLocalTimeDifferential }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the original DST logic.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.7"
  ::= {globalGroups 7}
```

5.9.8 Global Daylight Saving Time Group Revision 2

```
globalTimeMgmtDstGroupR2 OBJECT-GROUP
  OBJECTS { maxDaylightSavingEntries,
            dstEntryNumber,
            dstBeginMonth,
            dstBeginOccurrences,
            dstBeginDayOfWeek,
            dstBeginDayOfMonth,
            dstBeginSecondsToTransition,
            dstEndMonth,
            dstEndOccurrences,
            dstEndDayOfWeek,
            dstEndDayOfMonth,
            dstEndSecondsToTransition,
            dstSecondsToAdjust }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the DST logic according
      to a configurable table.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.8"
```



```
  ::= {globalGroups 8}
```

5.9.9 Global Local Time Group

```
globalTimeMgmtLocalGroup OBJECT-GROUP
  OBJECTS { controllerStandardTimeZone,
            controllerLocalTime }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The minimum objects necessary for managing local time.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.9"
  ::= {globalGroups 9}
```

5.9.10 Global Time Base Event Group

```
globalTimeBaseEventGroup OBJECT-GROUP
  OBJECTS { maxTimeBaseEventEntries,
            timeBaseScheduleNumber,
            timeBaseScheduleMonth,
            timeBaseScheduleDay,
            timeBaseScheduleDate,
            timeBaseScheduleDayPlan,
            maxDayPlans,
            maxDayPlanEvents,
            dayPlanNumber,
            dayPlanEventNumber
            dayPlanHour
            dayPlanMinute
            dayPlanAtionNuberOID,
            dayPlanStatus }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the original timebase
      schedule feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.10"
  ::= {globalGroups 10}
```

5.9.11 Global Time Base Event Group Extension

```
globalTimeBaseEventGroupExt OBJECT-GROUP
  OBJECTS { timeBaseScheduleTableStatus }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> Additional objects used to manage the status of the
      timebase schedule feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.11"
  ::= {globalGroups 11}
```

5.9.12 Global PMPP Group Revision 1

```
globalPmppGroupR1 OBJECT-GROUP
  OBJECTS { maxGroupAddresses,
            hdlcGroupAddressIndex,
            hdlcGroupAddress }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the PMPP group
     addresses.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.12"
```

```
  ::= {pmppGroups 12}
```

5.9.13 Global PMPP Group Revision 2

```
globalPmppGroupR2 OBJECT-GROUP
  OBJECTS { maxGroupAddresses,
            hdlcGroupAddressIndex,
            hdlcGroupAddressNumber }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the PMPP group
     addresses correcting for an ambiguity.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.8.127.2.13"
  ::= {pmppGroups 13}

END -- NTCIP1201-GlobalV1
```


Section 6
DEPRECATED AUXILIARY I/O V2 MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

The text provided from Section 6.1 through the end of Section 6 (except the headings) constitutes the standard NTCIP1201-AuxIOv2 MIB.

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1201 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP 1201 for the objects contained within this MIB.

Table 4: Compliance Statement for NTCIP1201-GlobalV1 MIB

Group	v01	v01A1	v02	v03
auxIOv2GroupR1			M	M

6.1 AUXILIARY I/O V2 HEADER

```
NTCIP1201-AuxIOv2 DEFINITIONS ::= BEGIN
IMPORTS
   MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Unsigned32
                                                        FROM SNMPv2-SMI
                                                          -- RFC 2578
   DisplayString
                                                         FROM SNMPv2-TC
                                                           -- RFC 2579
   global
                                                      FROM NTCIP8004v02

auxIOv2 MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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 20684-2.

     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

     Copyright (C) 2010, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.
```

```
                    Joint NEMA, AASHTO, and ITE
                NTCIP Management Information Base
                      DISTRIBUTION NOTICE

    To the extent and in the limited event these materials are distributed
    by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
    module, AASHTO/ITE/NEMA extends the permission for users to make and/or
    distribute unlimited copies of the unedited MIB module, including copies
    for commercial distribution, provided that each copy made and/or
    distributed contains this Notice.

    The MIB module may be distributed with a companion MIB module that
    contains one or more invocations of the AGENT-CAPABILITIES macro that
    specifies the capabilities of a specific implementation. By convention,
    such a companion MIB module might have the same module name followed by
    "-", followed the enterprise's Internet Assigned Number Authority
    (IANA)-assigned enterprise number.

    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 OR 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, OR 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 you.

    Use of these materials does not constitute an endorsement or affiliation
    by or between AASHTO, ITE, or NEMA and you, your company, or your
    products and services.

    NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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}
```

6.2 OBJECT IDENTITIES

6.2.1 AuxIOv2 Conformance Node

```
auxIOv2Conformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define conformance
      information for the auxIOv2 MIB.
```

```
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.127"
::= { auxIOv2 127 }

auxIOv2Compliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define compliance
      statements for the auxIOv2 MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.127.1"
::= { auxIOv2Conformance 1 }

auxIOv2Groups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define object groups for
      the auxIOv2 MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.127.2"
::= { auxIOv2Conformance 2 }
```

6.3 OBJECTS

6.3.1 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'.
       <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOTypeCount (ISO 20684-2)
       <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}
```

6.3.2 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'.
      <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOTypeCount (ISO 20684-2)
      <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}
```

6.3.3 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.
      <TableType> static
      <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOTable & FIELD-DEVICE-GPIO-
      MIB.fdGPIOPortTable (ISO 20684-2)
      <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.
      <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOEntry & FIELD-DEVICE-GPIO-
      MIB.fdGPIOPortEntry (ISO 20684-2)
      <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
   }
```

6.3.3.1 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.
      <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOType (ISO 20684-2)
      <Informative> The GPIO tables classify ports using a three-letter
      code. ISO 20684-2 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}
```

6.3.3.2 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.
      <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOPortNumber (ISO
       20684-2)
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.2"
::= {auxIOv2Entry 2}
```

6.3.3.3 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.
       <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOPortDescription (ISO
       20684-2)
       <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}
```

6.3.3.4 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.
       <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOPortMinValue &
       FIELD-DEVICE-GPIO-MIB.fdGPIOPortMaxValue (ISO 20684-2)
       <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}
```

6.3.3.5 Auxiliary Value Parameter

```
auxIOv2PortValue  OBJECT-TYPE
   SYNTAX       Unsigned32
   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.
      <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOPortValue (ISO
      20684-2)
      <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}
```

6.3.3.6 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).
       <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIOPortDirection (ISO
       20684-2)
       <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.3.1.6"
::= {auxIOv2Entry 6}
```

6.3.3.7 Auxiliary Port Last Commanded State Parameter

```
auxIOv2PortLastCommandedState OBJECT-TYPE
   SYNTAX       Unsigned32
   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).
       <Superseded by> FIELD-DEVICE-GPIO-MIB.fdGPIORequestedValue (ISO
       20684-2)
       <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}
```

6.4 COMPLIANCE GROUPS

6.4.1 Auxiliary I/O Version 2 Group

```
auxIOv2GroupR1 OBJECT-GROUP
  OBJECTS { maxAuxIOv2TableNumDigitalPorts,
            maxAuxIOv2TableNumAnalogPorts,
            auxIOv2PortType,
            auxIOv2PortNumber,
            auxIOv2PortDescription,
            auxIOv2PortResolution,
            auxIOv2PortValue,
            auxIOv2PortDirection,
            auxIOv2PortLastCommandedState }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the auxiliary IO ports
      according to the revised scheme under the global node.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.7.127.2.1"
  ::= {auxIOv2Groups 1}

END --NTCIP1201-AuxIOv2
```


Section 7
DEPRECATED AUXILIARY I/O MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

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

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1201 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP 1201 for the objects contained within this MIB.

Table 5: Compliance Statement for NTCIP1201-GlobalV1 MIB

Group	v01	v01A1	v02	v03
auxIOGroupR1		M		

7.1 AUXILIARY I/O HEADER

```
NTCIP1201-AuxIO DEFINITIONS ::= BEGIN
IMPORTS
   MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Unsigned32
                                                          FROM SNMPv2-SMI
                                                            -- RFC 2578
   DisplayString
                                                           FROM SNMPv2-TC
                                                             -- RFC 2579
   expGlobal
                                                       FROM NTCIP8004-NEMA


auxiliaryIO MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
```

```
  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

    Copyright (C) 2010, 2022 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 in whole or in part in any form, translation
    into other languages and display are reserved by the copyright owners
    under the laws of the United States of America, the Universal Copyright
    Convention, the Berne Convention, and the International and Pan American
    Copyright Conventions.

                          Joint NEMA, AASHTO, and ITE
                      NTCIP Management Information Base
                           DISTRIBUTION NOTICE

    To the extent and in the limited event these materials are distributed
    by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
    module, AASHTO/ITE/NEMA extends the permission for users to make and/or
    distribute unlimited copies of the unedited MIB module, including copies
    for commercial distribution, provided that each copy made and/or
    distributed contains this Notice.

    The MIB module may be distributed with a companion MIB module that
    contains one or more invocations of the AGENT-CAPABILITIES macro that
    specifies the capabilities of a specific implementation. By convention,
    such a companion MIB module might have the same module name followed by
    "-", followed the enterprise's Internet Assigned Number Authority
    (IANA)-assigned enterprise number.

    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 OR 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, OR 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 you.

    Use of these materials does not constitute an endorsement or affiliation
    by or between AASHTO, ITE, or NEMA and you, your company, or your
    products and services.

    NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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}
```

7.2 OBJECT IDENTITIES

7.2.1 AuxIO Configuration Node

```
auxIOConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define conformance
      information for the auxIO MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.2.1.127"
::= { auxiliaryIO 127 }

auxIOCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define compliance
      statements for the auxIO MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.2.1.127.1"
::= { auxIOConformance 1 }

auxIOGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define object groups for
      the auxIO MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.2.1.127.2"
::= { auxIOConformance 2 }
```

7.3 OBJECTS

7.3.1 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}
```

7.3.2 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}
```

7.3.3 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.
      <TableType> 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
   }
```

7.3.3.1 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}
```

7.3.3.2 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}
```

7.3.3.3 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}
```

7.3.3.4 Auxiliary Resolution Parameter

```
auxIOResolution OBJECT-TYPE
  SYNTAX       Integer32 (1..255)
  UNITS        "bits"
  MAX-ACCESS   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}
```

7.3.3.5 Auxiliary Value Parameter

```
auxIOValue OBJECT-TYPE
  SYNTAX       Unsigned32
  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}
```

7.3.3.6 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}
```

7.4 COMPLIANCE GROUPS

7.4.1 Auxiliary I/O Group

```
auxIOGroupR1 OBJECT-GROUP
  OBJECTS { maxAuxIODigital,
            maxAuxIOAnalog,
            auxIOPortType,
            auxIOPortNumber,
            auxIODescription,
            auxIOResolution,
            auxIOValue,
            auxIOv2PortDirection }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the auxiliary IO ports
      according to the revised scheme under the global node.
    <Object Identifier> 1.3.6.1.4.1.1206.2.2.1.127.2.1"
  ::= {auxIOGroups 1}

END  -- NTCIP1201-AuxIO
```

Section 8
DEPRECATED SNMP
MANAGEMENT INFORMATION BASE (MIB)
[NORMATIVE]

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

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP 1201 for the objects contained within this MIB.

Table 6: Compliance Statement for NTCIP1201-GlobalV1 MIB

Group	1103 v03
snmpConfigGroupR1	M

8.1 HEADER

```
NTCIP1201-SNMPConfig DEFINITIONS ::= BEGIN
IMPORTS
   MODULE-IDENTITY, OBJECT-TYPE
                                                FROM SNMPv2-SMI
                                                  -- RFC 2578
   application
                                                FROM NTCIP8004-Transportation

snmpConfig MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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
     Copyright © 1996, 2006, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                          Joint NEMA, AASHTO, and ITE
                    NTCIP Management Information Base
                          DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
     specifies the capabilities of a specific implementation. By convention,
     such a companion MIB module might have the same module name followed by
     "-", followed the enterprise's Internet Assigned Number Authority
     (IANA)-assigned enterprise number.

     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 OR 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, OR 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 you.
```

```
     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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}
```

8.2 OBJECT IDENTITIES

8.2.1 SNMP Conformance Node

```
snmpConfigConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define conformance
      information for the snmpConfig MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.1.127"
::= { snmpConfig 127 }

snmpConfigCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define compliance
      statements for the snmpConnfig MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.1.127.1"
::= { snmpConfigConformance 1 }

snmpConfigGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define object groups for
      the snmpConfig MIB.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.1.127.2"
::= { snmpConfigConformance 2 }
```

8.3 OBJECTS

8.3.1 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.
    <Superseded 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}
```

8.4 COMPLIANCE GROUPS

8.4.1 SNMP Configuration Group

```
snmpConfigGroupR1 OBJECT-GROUP
  OBJECTS { snmpMaxPacketSize }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the SNMP configuration.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.1.127.2.1"
  ::= {snmpConfigGroups 1}

END --NTCIP1201-SNMPConfig
```

Section 9
DEPRECATED SFMP
MANAGEMENT INFORMATION BASE (MIB)
[NORMATIVE]

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

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP 1103 for the objects contained within this MIB.

Table 7: Compliance Statement for NTCIP1201-SFMP MIB

Group	1103 v03
sfmpGroupR1	M

9.1 HEADER

```
NTCIP1201-SFMP DEFINITIONS ::= BEGIN
IMPORTS
   MODULE-IDENTITY, OBJECT-TYPE, Counter32
                                                    FROM SNMPv2-SMI
                                                      -- RFC 2578
   application
                                                    FROM NTCIP8004-Transportation

sfmp MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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
     Copyright © 1996, 2006, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                          Joint NEMA, AASHTO, and ITE
                     NTCIP Management Information Base
                          DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
     specifies the capabilities of a specific implementation. By convention,
     such a companion MIB module might have the same module name followed by
     "-", followed the enterprise's Internet Assigned Number Authority
     (IANA)-assigned enterprise number.

     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 OR 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, OR 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 you.
```

```
     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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}
```

9.2 OBJECT IDENTITIES

9.2.1 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 }
```

9.2.2 SFMP Conformance Node

```
sfmpConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define conformance
      clauses for SFMP.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.127"
::= { sfmp 127 }

sfmpCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define compliance
      statements for the SFMP MIB.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.127.1"
::= { sfmpConformance 1 }

sfmpGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define object groups for
      the SFMP MIB.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.127.2"
::= { sfmpConformance 2 }
```


9.3 OBJECTS

9.3.1 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 }
```

9.3.2 Number of Outgoing SFMP Packets

```
sfmpOutPkts  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.3 Number of Incoming SFMP Packets with Bad Version Numbers

```
sfmpInBadVersions OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.4 Number of Incoming SFMP Packets with Bad Community Names

```
sfmpInBadCommunityNames OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.5 Number of Incoming SFMP Packets with Bad Use of a Community Name

```
sfmpInBadCommunityUses OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.6 Number of Incoming SFMP Packets with Parsing Errors

```
sfmpInParseErrs  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.7 Reserved

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

9.3.8 Number of Incoming SFMP Packets indicating a Too Big Error

```
sfmpInTooBigs  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.9 Number of Incoming SFMP Packets indicating a No Such Name Error

```
sfmpInNoSuchNames  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.10 Number of Incoming SFMP Packets indicating a Bad Value Error

```
sfmpInBadValues  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.11 Number of Incoming SFMP Packets indicating a Read-Only Error

```
sfmpInReadOnlys  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.12 Number of Incoming SFMP Packets indicating a General Error

```
sfmpInGenErrs  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.13 Reserved

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

9.3.14 Reserved

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

9.3.15 Number of Incoming SFMP Get Requests

```
sfmpInGetRequests  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.16 Reserved

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

9.3.17 Number of Incoming SFMP Set Requests

```
sfmpInSetRequests  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.18 Number of Incoming SFMP Get Responses

```
sfmpInGetResponses  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.19 Reserved

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

9.3.20 Number of Outgoing SFMP Packets indicating a Too Big Error

```
sfmpOutTooBigs  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.21 Number of Outgoing SFMP Packets indicating a No Such Name Error

```
sfmpOutNoSuchNames  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.22 Number of Outgoing SFMP Packets indicating a Bad Value Error

```
sfmpOutBadValues  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.23 Number of Outgoing SFMP Packets indicating a Read-Only Error

```
sfmpOutReadOnly  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.24 Number of Outgoing SFMP Packets indicating a General Error

```
sfmpOutGenError  OBJECT-TYPE
  SYNTAX      Counter
```

```
  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 }
```

9.3.25 Number of Outgoing SFMP Get Requests

```
sfmpOutGetRequests  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.26 Reserved

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

9.3.27 Number of Outgoing SFMP Set Requests

```
sfmpOutSetRequests  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.28 Number of Outgoing SFMP Get Responses

```
sfmpOutGetResponses  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.29 Number of Outgoing SFMP Trap Messages

```
sfmpOutTrapMessages  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.30 Reserved

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

9.3.31 Number of Incoming SFMP Set Requests – No Replies

```
sfmpInSetRequestsNoReply  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.32 Number of Incoming SFMP Set Responses

```
sfmpInSetResponses  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.33 Number of Incoming SFMP Error Responses

```
sfmpInErrorResponses  OBJECT-TYPE
  SYNTAX      Counter
  UNITS       "packets"
  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 }
```

9.3.34 Number of Outgoing SFMP Set Requests – No Replies

```
sfmpOutSetRequestsNoReply  OBJECT-TYPE
  SYNTAX      Counter
  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 }
```

9.3.35 Number of Outgoing SFMP Set Responses

```
sfmpOutSetResponses  OBJECT-TYPE
  SYNTAX  Counter
  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 }
```

9.3.36 Number of Outgoing SFMP Error Responses

```
sfmpOutErrorResponses  OBJECT-TYPE
  SYNTAX      Counter
  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"
::= { sfmpStatistics 36 }
```

9.4 COMPLIANCE GROUPS

9.4.1 SFMP Group

```
sfmpGroupR1 OBJECT-GROUP
  OBJECTS { sfmpInPkts,
            sfmpOutPkts,
            sfmpInBadVersions,
```



```
            sfmpInBadCommunityNames,
            sfmpInBadCommunityUses,
            sfmpInParseErrors,
            sfmpInTooBigs,
            sfmpInNoSuchNames,
            sfmpInBadValues,
            sfmpInReadOnlys,
            sfmpInGenErrs,
            sfmpInGetRequests,
            sfmpInSetRequests,
            sfmpInGetResponses,
            sfmpOutTooBigs,
            sfmpOutNoSuchNames,
            sfmpOutBadValues,
            sfmpOutReadOnly,
            sfmpOutGenErr,
            sfmpOutGetRequests,
            sfmpOutSetRequests,
            sfmpOutGetResponses,
            sfmpOutTrapMessages,
            sfmpInSetRequestsNoReply,
            sfmpInSetResponses,
            sfmpInErrorResponses,
            sfmpOutSetRequestsNoReply,
            sfmpOutSetResponses,
            sfmpOutErrorResponses }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for monitoring SFMP statistics.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.2.127.2.1"
  ::= {sfmpGroups 1}

END  -- NTCIP1201-SFMP
```


Section 10
DEPRECATED STMP
MANAGEMENT INFORMATION BASE (MIB)
[NORMATIVE]

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

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP for the objects contained within this MIB.

Table 8: Compliance Statement for NTCIP1201-DynObjMgmt MIB

Group	1101 v01	1101 v01A1	1103 v01	1103 v02	1103 v03
dynObjMgmtGroupR1	M				
dynObjMgmtGroupR2		M	M	M	M
dynObjMgmtGroupR2Ext				M	M
dynObjDataGroupR1	M	M			

10.1 HEADER

```
NTCIP1201-DynObjMgmt DEFINITIONS ::= BEGIN
IMPORTS
   zeroDotZero, MODULE-IDENTITY, OBJECT-TYPE
                                                    FROM SNMPv2-SMI
                                                      -- RFC 2578
   protocols, OwnerString
                                                  FROM NTCIP8004v02

dynObjMgmt MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
```



```
  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

     Copyright © 1996, 2006, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                          Joint NEMA, AASHTO, and ITE
                     NTCIP Management Information Base
                          DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
     specifies the capabilities of a specific implementation. By convention,
     such a companion MIB module might have the same module name followed by
     "-", followed the enterprise's Internet Assigned Number Authority
     (IANA)-assigned enterprise number.

     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 OR 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, OR 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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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}
```

10.2 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) }
```

10.3 OBJECT IDENTITIES

10.3.1 Data Node

```
dynObjData   OBJECT-IDENTITY
  STATUS  current
  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 }
```

10.3.2 Dynamic Object Management Conformance Node

```
dynObjMgmtConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define conformance
      clauses for dynamic object management.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.127"
::= { dynObjMgmt 127 }

dynObjMgmtCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define compliance
      statements for ynamic object management.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.127.1"
::= { dynObjMgmtConformance 1 }

dynObjMgmtGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define object groups for
      dynamic object management.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.127.2"
::= { dynObjMgmtConformance 2 }
```

10.4 DYNAMIC OBJECT DEFINITION

10.4.1 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 }
```

10.4.2 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
    <TableType> static
    <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
     <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     OwnerString,       -- previously deprecated
      dynObjStatus    EntryStatus }      -- previously deprecated
```

10.4.2.1 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.
    <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 }
```

10.4.2.2 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.
    <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 }
```

10.4.2.3 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.
    <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 }
```

10.4.2.4 Dynamic Object Owner

```
dynObjOwner OBJECT-TYPE
  SYNTAX      OwnerString
  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 }
```

10.4.2.5 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.
    <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 }
```

10.5 DYNAMIC OBJECT DATA

```
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.
    <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 }

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
    <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 }

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
    <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 }

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
    <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 }
```



```
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
    <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 }

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
    <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 }

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
    <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 }

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
    <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 }

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
    <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 }

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
    <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 }

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
    <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 }

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
    <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 }

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
    <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 }
```

10.5.1 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.
    <TableType> static
    <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.
    <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   OwnerString,
   dynObjConfigStatus  ConfigEntryStatus }
```

10.5.1.1 Dynamic Object Configuration Owner

```
dynObjConfigOwner  OBJECT-TYPE
  SYNTAX      OwnerString
  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.
    <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}
```

10.5.1.2 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.
    <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}
```

10.6 COMPLIANCE GROUPS

10.6.1 Dynamic Object Definition Group

```
dynObjMgmtGroupR1 OBJECT-GROUP
  OBJECTS { dynObjNumber,
            dynObjIndex,
            dynObjVariable,
            dynObjOwner,
            dynObjStatus }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for defining dynamic objects
      according to the original standard.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.127.2.1"
  ::= {dynObjMgmtGroups 1}
```

10.6.2 Dynamic Object Definition Group 2

```
dynObjMgmtGroupR2 OBJECT-GROUP
  OBJECTS { dynObjNumber,
            dynObjIndex,
            dynObjVariable,
            dynObjConfigOwner,
            dynObjConfigStatus }
```



```
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for defining dynamic object based on
      the revised standard that incorporated field experience.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.127.2.2"
  ::= {dynObjMgmtGroups 2}
```

10.6.3 Dynamic Object Definition Group 2 Extension

```
dynObjMgmtGroupR2Ext OBJECT-GROUP
  OBJECTS { dynObjDefTableMaxEntries }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The object for determining the feature support within
     the dynamic object table.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.127.2.3"
  ::= {dynObjMgmtGroups 3}
```

10.6.4 Dynamic Object Data Group

```
dynObjDataGroupR1 OBJECT-GROUP
  OBJECTS { dynObj1,
            dynObj2,
            dynObj3,
            dynObj4,
            dynObj5,
            dynObj6,
            dynObj7,
            dynObj8,
            dynObj9,
            dynObj10,
            dynObj11,
            dynObj12,
            dynObj13 }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for monitoring dynamic objects as
      SNMP objects.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.3.127.2.4"
  ::= {dynObjMgmtGroups 4}

END  -- NTCIP1201-DynObjMgmt
```

Section 11
DEPRECATED STMP MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

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

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP for the objects contained within this MIB.

Table 9: Compliance Statement for NTCIP1201-STMPStatistics MIB

Group	1103 v01	1103 v02	1103 v03
stmpStatisticsGroupR1	O	O	O

11.1 HEADER

```
NTCIP1201-STMP DEFINITIONS ::= BEGIN
IMPORTS
   zeroDotZero, MODULE-IDENTITY, OBJECT-TYPE, Counter32
                                                        FROM SNMPv2-SMI
                                                          -- RFC 2578
   application
                                                      FROM NTCIP8004-SMI;

stmp MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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
```

```
     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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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.
::= {stmp 1}
```

11.2 OBJECT IDENTITIES

11.2.1 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"
::= { stmpStatistics 127 }
```

11.2.2 STMP Statistics Conformance Node

```
stmpStatsConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define conformance
      clauses for STMP .
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.127"
::= { stmpStatistics 127 }

stmpStatsCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define compliance
      statements for STMP.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.127.1"
::= { stmpStatsConformance 1 }

stmpStatsGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to define groups
```



```
      for STMP.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.127.2"
::= { stmpStatsConformance 2 }
```

11.3 OBJECTS

11.3.1 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 }
```

11.3.2 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 }
```

11.3.3 Reserved

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

11.3.4 Reserved

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

11.3.5 Reserved

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

11.3.6 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 }
```

11.3.7 Reserved

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

11.3.8 Number of Incoming STMP Packets indicating a Too Big Error

```
stmpInTooBigs  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 }
```

11.3.9 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 }
```

11.3.10 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 }
```

11.3.11 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 }
```

11.3.12 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 }
```

11.3.13 Reserved

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

11.3.14 Reserved

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

11.3.15 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 }
```

11.3.16 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 }
```

11.3.17 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 }
```

11.3.18 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 }
```

11.3.19 Reserved

```
-- node 19 is reserved for in trap responses to parallel
-- SNMP, but it does not apply to STMP
```

11.3.20 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 }
```


11.3.21 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 }
```

11.3.22 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 }
```

11.3.23 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 }
```


11.3.24 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 }
```

11.3.25 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 }
```

11.3.26 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 }
```

11.3.27 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 }
```

11.3.28 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 }
```

11.3.29 Reserved

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

11.3.30 Reserved

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

11.3.31 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 }
```

11.3.32 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 }
```

11.3.33 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 }
```

11.3.34 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 }
```

11.3.35 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 }
```

11.3.36 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 }
```

11.4 COMPLIANCE GROUPS

11.4.1 Dynamic Object Definition Group

```
stmpStatisticsGroupR1 OBJECT-GROUP
  OBJECTS { stmpInPkts,
            stmpOutPkts,
```



```
            stmpInParseErrs,
            stmpInTooBigs,
            stmpInNoSuchNames,
            stmpInBadValues,
            stmpInReadOnlys,
            stmpInGenErrs,
            stmpInGetRequests,
            stmpInGetNexts,
            stmpInSetRequests,
            stmpInGetResponses,
            stmpOutTooBigs,
            stmpOutNoSuchNames,
            stmpOutBadValues,
            stmpOutReadOnly,
            stmpOutGenError,
            stmpOutGetRequests,
            stmpOutGetNexts,
            stmpOutSetRequests,
            stmpOutGetResponses,
            stmpInSetRequestsNoReply,
            stmpInSetResonses,
            stmpInErrorResponses,
            stmpOutSetRequestsNoReply,
            stmpOutSetResponses,
            stmpOutErrorResponses }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for monitoring STMP statistics.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.3.127.2.1"
  ::= {stmpStatsGroups 1}

END  -- NTCIP1201-STMPStatistics
```


Section 12
DEPRECATED STMP CONFIGURATION MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

The text provided from Section 12.1 through the end of Section 12 (except the headings) constitutes the standard NTCIP1201-ProfilesSTMP MIB.

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP for the objects contained within this MIB.

Table 10: Compliance Statement for NTCIP1201-ProfilesSTMP MIB

Group	1103 v01	1103 v02	1103 v03
profilesSTMPGroupR1	O	O	O

12.1 HEADER

```
NTCIP1201-ProfilesSTMP DEFINITIONS ::= BEGIN
IMPORTS
   MODULE-IDENTITY, OBJECT-TYPE
                                                        FROM SNMPv2-SMI
                                                          -- RFC 2578
   profiles
                                                      FROM NTCIP8004v02;

profilesSTMP MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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
     Copyright © 1996, 2006, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                          Joint NEMA, AASHTO, and ITE
                     NTCIP Management Information Base
                          DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
     specifies the capabilities of a specific implementation. By convention,
     such a companion MIB module might have the same module name followed by
     "-", followed the enterprise's Internet Assigned Number Authority
     (IANA)-assigned enterprise number.

     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 OR 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, OR 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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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 }
```

12.2 OBJECT IDENTITIES

12.2.1 Profile STMP Conformance Node

```
profilesSTMPConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the STMP profile.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.2.127"
::= { profilesSTMP 127 }

profilesSTMPCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the STMP profile.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.2.127.1"
::= { profilesSTMPConformance 1 }

profilesSTMPGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the STMP profile.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.2.127.2"
::= { profilesSTMPConformance 2 }
```

12.3 OBJECTS

12.3.1 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.
       <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 }
```

12.3.2 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.
    <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 }
```

12.4 COMPLIANCE GROUPS

12.4.1 Dynamic Object Definition Group

```
profilesSTMPGroup OBJECT-GROUP
  OBJECTS { dynamicObjectPersistence,
            dynamicObjectTableConfigID }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the STMP profile.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.2.2.127.2.1"
  ::= {profilesSTMPGroups 1}

END  -- NTCIP1201-ProfilesSTMP
```

Section 13
DEPRECATED LOGICAL NAMES MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

The text provided from Section 13.1 through the end of Section 13 (except the headings) constitutes the standard NTCIP1201-LogicalNames MIB.

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP for the objects contained within this MIB.

Table 11: Compliance Statement for NTCIP1201-LogicalNames MIB

Group	1103 v01	1103 v02	1103 v03
logicalNamesGroupR1	O	O	O

13.1 HEADER

```
NTCIP1201-LogicalNames DEFINITIONS ::= BEGIN
IMPORTS
   MODULE-IDENTITY, OBJECT-TYPE, IpAddress
                                                    FROM SNMPv2-SMI
                                                      -- RFC 2578
   application, RowStatusStatic
                                                  FROM NTCIP8004v02;

logicalNames MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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
```

```
     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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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 }
```

13.2 OBJECT IDENTITIES

13.2.1 Logical Names Conformance Node

```
logicalNamesConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage logical names.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.127"
::= { logicalNames 127 }

logicalNamesCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage logical names.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.127.1"
::= { logicalNamesConformance 1 }

logicalNamesGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage logical names.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.127.2"
::= { logicalNamesConformance 2 }
```

13.3 OBJECTS

13.3.1 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 }
```

13.3.2 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.
    <TableType> static
    <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.
    <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   NetworkAddress,
   logicalNameTranslationStatus          RowStatusStatic }
```

13.3.2.1 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.
    <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 }
```

13.3.2.2 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.
    <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 }
```

13.3.2.3 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.
    <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  { 0 }
::= { logicalNameTranslationEntry 3 }
```

13.3.2.4 Logical Name Translation Status

```
logicalNameTranslationStatus OBJECT-TYPE
  SYNTAX  RowStatusStatic
  MAX-ACCESS  read-write
  STATUS      deprecated
  DESCRIPTION
    "<Definition> This object allows for the management of rows
      within the 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> 1.3.6.1.4.1.1206.4.1.1.7.4.2.1.4"
```



```
   DEFVAL  { invalid }
::= { logicalNameTranslationEntry 4 }
```

13.4 COMPLIANCE GROUPS

13.4.1 Logical Names Group

```
logicalNamesGroupR1 OBJECT-GROUP
  OBJECTS { logicalNameTranslationTableMaxEntries,
            logicalNameTranslationIndex,
            logicalNameTranslationLogicalName,
            logicalNameTranslationNetworkAddress,
            logicalNameTranslationStatus }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing logical names.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.4.127.2.1"
  ::= {logicalNamesGroups 1}

END  -- NTCIP1201-LogicalNames
```

Section 14
DEPRECATED REPORT MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

The text provided from Section 14.1 through the end of Section 14 (except the headings) constitutes the standard NTCIP1201-Report MIB.

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP for the objects contained within this MIB.

Table 12: Compliance Statement for NTCIP1201-Report MIB

Group	1201 v01	1201 v02	1103 v01	1103 v02	1103 v03
reportGroupR1	O	O	O	O	O
reportGroupR1Ext					O

14.1 HEADER

```
NTCIP1201-Report DEFINITIONS ::= BEGIN
IMPORTS
   zeroDotZero, MODULE-IDENTITY, OBJECT-TYPE, Counter32, Opaque
                                                        FROM SNMPv2-SMI
                                                          -- RFC 2578
   global
                                                        FROM NTCIP1201-Global;

globalReport MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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 the
     ISO 20684 series.

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

     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4
```

```
     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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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 }

-- NOTE—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.
```

14.2 OBJECT IDENTITIES

14.2.1 Logical Names Conformance Node

```
reportConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the event logging
      feature.
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.127"
::= { globalReport 127 }

reportCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the event logging
      feature.
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.127.1"
::= { reportConformance 1 }

reportGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the event logging
      feature.
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.127.2"
::= { reportConformance 2 }
```


14.3 OBJECTS

14.3.1 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 }
```

14.3.2 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.
    <TableType> static
    <Superseded by> LOG-MIB.fdLogManagerTable (ISO 20684-5)
    <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
    <Superseded by> LOG-MIB.fdLogManagerEntry (ISO 20684-5)
    <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     Counter,
```

```
   eventClassDescription   OCTET STRING,
   eventClassNumRowsInLog  Integer32,
   eventClassNumEvents     Integer32}
```

14.3.2.1 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.
    <Superseded by> LOG-MIB.fdLogManagerName (ISO 20684-5)
    <Informative> The replacement object is a secondary index that uses an
      SnmpAdminString syntax; the primary index is LOG-MIB.fdLogManagerOwner
      which also uses an SnmpAdminSyntax.
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.6.1.1"
::= { eventClassEntry 1 }
```

14.3.2.2 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
      <Superseded by> LOG-MIB.fdLogManagerSizeLimit & LOG-
        MIB.fdLogManagerEntryLimit (ISO 20684-5)
      <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 }
```

14.3.2.3 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.
    <Superseded by> LOG-MIB.fdLogManagerClearDate & LOG-
      MIB.fdLogManagerClearTime (ISO 20684-5)
    <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 }
```

14.3.2.4 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.
    <Superseded by> LOG-MIB.fdLogManagerDescription (ISO 20684-5)
    <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 }
```

14.3.2.5 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 }
```

14.3.2.6 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.
    <Superseded by> LOG-MIB.fdLogManagerEventsLogged (ISO 20684-5)
    <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 }
```

14.3.3 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}
```

14.3.4 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.
    <TableType> static
    <Superseded by> LOG-MIB.fdLogFactoryTable (ISO 20684-5) & COND-
      TRIGGER-MIB.fdCondTriggerTable (ISO 20684-3)
    <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.
    <Superseded by> LOG-MIB.fdLogFactoryEntry (ISO 20684-5) & COND-
      TRIGGER-MIB.fdCondTriggerEntry (ISO 20684-3)
    <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 }
```

14.3.4.1 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.
```

```
    <Superseded by> LOG-MIB.fdLogManagerOwner, fdLogFactoryName (ISO
      20684-5), COND-TRIGGER-MIB.fdActionOwner & fdCondTriggerName (ISO
      20684-3)
    <Informative> The replacement object specifies an SnmpAdminString
      Syntax, which allows management systems to automatically recognize
      the entry as text in any language.
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.1"
::= { eventLogConfigEntry 1 }
```

14.3.4.2 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.

      NOTE-The event cannot be logged if the EventClass has an
      eventClassLimit of zero (0).
    <Superseded by> LOG-MIB.fdLogEventFactoryLogName (ISO 20684-5)
    <Informative> 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 }
```

14.3.4.3 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.

  <Superseded by> COND-TRIGGER-MIB.fdCondTriggerMode (ISO 20684-3)
  <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 }
```

14.3.4.4 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).
    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerValue &
      fdCondTriggerValueOctet (ISO 20684-3)
    <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 }
```

14.3.4.5 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.
    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerValue2 (ISO 20684-3)
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.5"
   DEFVAL  {0}
```



```
::= { eventLogConfigEntry 5 }
```

14.3.4.6 Event Log Configuration Compare Object Identifier Parameter

```
eventConfigCompareOID  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 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.
    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerObject (ISO 20684-3)
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.6"
  DEFVAL  { zeroDotZero }
::= { eventLogConfigEntry 6 }
```

14.3.4.7 Event Log Configuration Log Object Identifier Parameter

```
eventConfigLogOID  OBJECT-TYPE
  SYNTAX      OBJECT IDENTIFIER
  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 }

    <Superseded by> LOG-MIB.fdLogEventFactoryObjectID (ISO 20684-5)
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.7"
  DEFVAL  { zeroDotZero }
::= { eventLogConfigEntry 7 }
```

14.3.4.8 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.

    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerActionOwner,
      fdCondTriggerAction, fdCondTriggerActionOwner2, &
      fdCondTriggerAction2 (ISO 20684-3)
    <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 }
```

14.3.4.9 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
    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerRowStatus (ISO 20684-3)
      & LOG-MIB.fdLogEventFacoryRowStatus (ISO 20684-5)
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.2.1.9"
::= { eventLogConfigEntry 9 }
```

14.3.5 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.
    <Superseded by> LOG-MIB.fdLogsGlobalEntryLimit and
      fdLogsGlobalSizeLimit (ISO 20684-3)
    <Informative> The replacement objects are for all managers.
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.3"
::= { globalReport 3}
```

14.3.6 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.
    <TableType> dynamic
    <Superseded by> LOG-MIB.fdLogTable (ISO 20684-5)
    <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.
    <Superseded by> LOG-MIB.fdLogEntry (ISO 20684-5)
    <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 }
```

14.3.6.1 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.
    <Superseded by> LOG-MIB.fdLogManagerOwner & fdLogManagerName (ISO
      20684-5)
    <Informative> The replacement objects are SnmpAdminStrings
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1.1"
::= { eventLogEntry 1 }
```

14.3.6.2 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.
    <Superseded by> LOG-MIB.fdLogIndex (ISO 20684-5)
    <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 }
```

14.3.6.3 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.
    <Superseded by> LOG-MIB.fdLogFactoryName (ISO 20684-5)
```



```
    <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 }
```

14.3.6.4 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.
    <Superseded by> LOG-MIB.fdLogEventDate, fdLogEventTime, fdLogDate,
      fdLogTime (ISO 20684-5)
    <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 }
```

14.3.6.5 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.
    <Superseded by> LOG-MIB.fdLogValue (ISO 20684-5)
    <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 }
```

14.3.6.6 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).
    <Informative> This data content is covered by the new time format used
      by LOG-MIB.fdLogEventTime & fdLogTime (ISO 20684-5)
    <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.4.1.6"
::= { eventLogEntry 6 }
```

14.3.7 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.
    <Superseded by> LOG-MIB.fdLogsTotalLogged (ISO 20684-5)
    <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 }
```

14.3.8 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.
    <Superseded by> LOG-MIB.fdLogsRecordingLatency (ISO 20684-5)
    <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 }
```

14.4 COMPLIANCE GROUPS

14.4.1 Report Group

```
reportGroupR1 OBJECT-GROUP
  OBJECTS { maxEventClasses,
            eventClassNumber,
            eventClassLimit,
            eventClassClearTime,
            eventClassDescription,
            eventClassNumRowsInLog,
            eventClassNumEvents,
            maxEventLogConfigs,
            eventConfigID,
            eventConfigClass,
            eventConfigMode,
            eventConfigComapreValue,
            eventConfigCompareValue2,
            eventConfigLogOID,
            eventConfigAction,
            eventConfigStatus,
            maxEventLogSize,
            eventLogClass,
            eventLogNumber,
            eventLogID,
            eventLogTime,
            eventLogValue }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the event logging
      feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.127.2.1"
  ::= {reportGroups 1}
```

14.4.2 Report Group Extension

```
reportGroupR1Ext OBJECT-GROUP
  OBJECTS { eventLogTimeMilliseconds,
            numEvents,
            eventTimeLatency }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The extended objects necessary for managing the event
      logging feature.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.4.127.2.2"
  ::= {reportGroups 2}
```

```
END  -- NTCIP1201-Report
```

Section 15
DEPRECATED SECURITY MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

The text provided from Section 15.1 through the end of Section 15 (except the headings) constitutes the standard NTCIP1201-Security MIB.

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP for the objects contained within this MIB.

Table 13: Compliance Statement for NTCIP1201-Security MIB

Group	1201 v01	1201 v02	1103 v01	1103 v02	1103 v03
securityGroupR1	O	O	O	O	O

15.1 HEADER

```
NTCIP1201-Security DEFINITIONS ::= BEGIN
IMPORTS

MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Gauge32
                                                    FROM SNMPv2-SMI
                                                      -- RFC 2578
MODULE-COMPLIANCE, OBJECT-GROUP
                                                    FROM SNMPv2-CONF
                                                      -- RFC 2580
global
                                                    FROM NTCIP1201-Global;

security MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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
     Copyright © 1996, 2006, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                          Joint NEMA, AASHTO, and ITE
                     NTCIP Management Information Base
                          DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
     specifies the capabilities of a specific implementation. By convention,
     such a companion MIB module might have the same module name followed by
     "-", followed the enterprise's Internet Assigned Number Authority
     (IANA)-assigned enterprise number.

     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 OR 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, OR 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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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}
```

15.2 OBJECT IDENTITIES

15.2.1 Security Conformance Node

```
securityConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the community
      names for SNMPv1 deployments.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.127"
::= { security 127 }

securityCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the community
      names for SNMPv1 deployments.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.127.1"
::= { securityConformance 1 }

securityGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the community
      names for SNMPv1 deployments.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.127.2"
::= { securityConformance 2 }
```

15.3 OBJECTS

15.3.1 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 }
```

15.3.2 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 }
```

15.3.3 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.
    <TableType> 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  Gauge
     }
```

15.3.3.1 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 }
```

15.3.3.2 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 }
```

15.3.3.3 User Community Name Mask Parameter

```
communityNameAccessMask  OBJECT-TYPE
  SYNTAX  Gauge
  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 }
```

15.4 COMPLIANCE GROUPS

15.4.1 Security Group

```
securityGroupR1 OBJECT-GROUP
  OBJECTS { communityNameAdmin,
            communityNamesMax,
            communityNameIndex,
            communityNameUser,
            communityNameAccessMask }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the community names for
      SNMPv1 deployments.
     <Object Identifier> 1.3.6.1.4.1.1206.4.2.6.5.127.2.1"
  ::= {securityGroups 1}

END  -- NTCIP1201-Security
```


Section 16
DEPRECATED TRAP
MANAGEMENT INFORMATION BASE (MIB)
[NORMATIVE]

The text provided from Section 16.1 through the end of Section 16 (except the headings) constitutes the standard NTCIP1201-NtcipTraps MIB.

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1103. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP for the objects contained within this MIB.

Table 14: Compliance Statement for NTCIP1201-Security MIB

Group	1103 v03
watchBlockGroupR1	O
reportBlockGroupR1	O
trapMgmtGroupR1	M
trapClearGroupR1	O
trapGroupR1	M

16.1 HEADER

```
NTCIP1201-NtcipTraps DEFINITIONS ::= BEGIN
IMPORTS

zeroDotZero, MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
OBJECT-IDENTITY, Counter32, IpAddress
                                                FROM SNMPv2-SMI
                                                -- RFC 2578

DisplayString
                                                FROM SNMPv2-TC
                                                -- RFC 2579
```

```
MODULE-COMPLIANCE, OBJECT-GROUP
                                                    FROM SNMPv2-CONF
                                                    -- RFC 2580

ITSOerString
                                                    FROM FIELD-DEVICE-TC-MIB
                                                    -- ISO 20684-1 Annex A
application, protocols, RowStatusStatic
                                                    FROM NTCIP8004-Transportation
eventConfigID
                                                    FROM NTCIP1201-Report;

ntcipTraps MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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 the
     ISO 20684 series.

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

     <Object Identifier> 1.3.6.1.4.1.1206.4.1.4
     Copyright © 1996, 2006, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                         Joint NEMA, AASHTO, and ITE
                    NTCIP Management Information Base
                          DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.
```

```
    The MIB module may be distributed with a companion MIB module that
    contains one or more invocations of the AGENT-CAPABILITIES macro that
    specifies the capabilities of a specific implementation. By convention,
    such a companion MIB module might have the same module name followed by
    "-", followed the enterprise's Internet Assigned Number Authority
    (IANA)-assigned enterprise number.

    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 OR 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, OR 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 you.

    Use of these materials does not constitute an endorsement or affiliation
    by or between AASHTO, ITE, or NEMA and you, your company, or your
    products and services.

    NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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}
```

16.2 OBJECT IDENTITIES

16.2.1 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.
```

```
      Note that 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 }
```

16.2.2 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 }
```

16.2.3 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 }
```

16.2.4 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}
```

16.2.5 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}
```

16.2.6 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}
```

16.2.7 Trap Management Conformance Node

```
ntcipTrapConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage traps.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.127"
::= { ntcipTraps 127 }

ntcipTrapCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage traps.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.127.1"
::= { ntcipTrapConformance 1 }

ntcipTrapGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage traps.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.127.2"
::= { ntcipTrapConformance 2 }
```

16.3 WATCH BLOCKS

```
-- 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
```

16.3.1 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupsMaxObjects (ISO
      20684-7)
    <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 }
```

16.3.2 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 }
```

16.3.3 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.
    <TableType> static
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupFieldTable (ISO 20684-7)
    <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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupFieldEntry (ISO 20684-7)
    <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  RowStatusStatic,
   watchBlock   Integer32,
```



```
   watchOID     OBJECT IDENTIFIER }
```

16.3.3.1 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupFieldIndex (ISO 20684-7)
    <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 }
```

16.3.3.2 Watch Status Parameter

```
watchStatus  OBJECT-TYPE
  SYNTAX      RowStatusStatic
  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 }
```

16.3.3.3 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupName (ISO 20684-7)
    <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 }
```

16.3.3.4 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupFieldObject (ISO
      20684-7)
    <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 }
```

16.3.4 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.
    <TableType> static
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupTable (ISO 20684-7)
    <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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupEntry (ISO 20684-7)
    <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         RowStatusStatic,
    watchBlockDescription    OCTET STRING,
    watchBlockValue          OerString }
```

16.3.4.1 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupName (ISO 20684-7)
    <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 }
```

16.3.4.2 Watch Block Status

```
watchBlockStatus  OBJECT-TYPE
  SYNTAX      RowStatusStatic
  MAX-ACCESS  read-write
  STATUS      deprecated
  DESCRIPTION
    "<Definition>The value of this object indicates the current status of
      this row in the table.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupRowStatus (ISO 20684-7)
    <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 }
```

16.3.4.3 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupDescription (ISO 20684-7)
    <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 }
```

16.3.4.4 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupCurrentValue (ISO 20684-7)
    <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 20684-1. 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 { "" }
::= { watchBlockEntry 4 }
```

16.4 REPORT BLOCKS

```
-- The reportBlocks feature is replaced by the same fdObjectGroup feature
-- defined in ISO 20684-7 as used to replace the watchBlocks. The
-- <Informative> notes in this section are generally omitted as they would
-- repeat the content contained in the watchBlocks section of the MIB.
```

16.4.1 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupsMaxObjects (ISO 20684-7)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.1"
::= { reportBlocks 1 }
```

16.4.2 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 }
```

16.4.3 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.
    <TableType> static
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupFieldTable (ISO 20684-7)
    <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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupFieldEntry (ISO 20684-7)
    <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  RowStatusStatic,
   reportBlock   Integer32,
   reportOID     OBJECT IDENTIFIER }
```

16.4.3.1 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupFieldIndex (ISO 20684-7)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3.1.1"
::= { reportObjectDefinitionEntry 1 }
```

16.4.3.2 Report Status Parameter

```
reportStatus  OBJECT-TYPE
  SYNTAX      RowStatusStatic
  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 }
```

16.4.3.3 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupName (ISO 20684-7)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3.1.3"
   DEFVAL { 1 }
::= { reportObjectDefinitionEntry 3 }
```

16.4.3.4 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupFieldObject (ISO 20684-7)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.3.1.4"
   DEFVAL { zeroDotZero }
::= { reportObjectDefinitionEntry 4 }
```


16.4.4 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.
    <TableType> static
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupTable (ISO 20684-7)
    <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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupEntry (ISO 20684-7)
    <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       RowStatusStatic,
   reportBlockDescription  OCTET STRING,
   reportBlockValue        OerString }
```

16.4.4.1 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupName (ISO 20684-7)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1.1"
::= { reportBlockEntry 1 }
```

16.4.4.2 Report Block Status

```
reportBlockStatus  OBJECT-TYPE
  SYNTAX      RowStatusStatic
  MAX-ACCESS  read-write
  STATUS      deprecated
  DESCRIPTION
    "<Definition>The value of this object indicates the current status of
      this row in the table.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupRowStatus (ISO 20684-7)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1.2"
  DEFVAL { invalid }
::= { reportBlockEntry 2 }
```

16.4.4.3 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupDescription (ISO 20684-7)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1.3"
  DEFVAL { "" }
::= { reportBlockEntry 3 }
```

16.4.4.4 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.
    <Superseded by> OBJECT-GROUP-MIB.fdObjectGroupCurrentValue (ISO 20684-7)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.7.4.1.4"
   DEFVAL { "" }
::= { reportBlockEntry 4 }
```

16.5 TRAP MANAGEMENT

16.5.1 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.
    <Superseded by> NOTIFICATION-MIB.fdNotificationsEnabled (ISO 20684-4)
    <Informative> The replacement object is defined as a TruthValue (RFC
      2579), which is an enumeration where 1 = 'true' (in this case,
      enabled) and 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 }
```

16.5.2 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 183ventide (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).
    <Superseded by> NOTIFICATION-MIB.fdNotificationData (ISO 20684-4)
    <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 20684-1. 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 }
```

16.5.3 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 }
```

16.5.4 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 20684-4.
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.1.4"
   ::= { trapMgmt 4 }
```

16.5.5 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.
    <Superseded by> NOTIFICATION-MIB.fdNotificationsMaxSize (ISO 20684-4)
    <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 }
```

16.5.6 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.
    <TableType> static
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelTable (ISO 20684-4)
    <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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelEntry (ISO 20684-4)
    <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               Counter,
        trapMgmtRowStatus               RowStatusStatic,
        trapMgmtSeqNum                  Integer32,
        trapMgmtSeqNumAck               Integer32
             }
```

16.5.6.1 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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelID (ISO 20684-4)
    <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 }
```

16.5.6.2 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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelTarget (ISO 20684-4)
    <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 }
```

16.5.6.3 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.
    <Superseded 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 }
```

16.5.6.4 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
    <Superseded 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 }
```

16.5.6.5 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
    <Superseded 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 snmpDTLSUDPDomain 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 }
```

16.5.6.6 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).
    <Superseded 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 }
```

16.5.6.7 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.
      Note: A value of one indicates that the agent attempts a
      maximum of two transmissions.
    <Superseded by> SNMP-TARGET-MIB.snmpTargetAddrRetryCount (RFC 3413)
    <Informative> 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 }
```

16.5.6.8 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.
    <Superseded 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 }
```

16.5.6.9 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 }
```

16.5.6.10 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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelQueueDepth (ISO 20684-4)
    <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 }
```

16.5.6.11 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 }
```

16.5.6.12 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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelAntiStreamRate (ISO
      20684-4)
    <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 }
```

16.5.6.13 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 }
```

16.5.6.14 Trap Lost Trap Counter

```
trapMgmtLostTraps   OBJECT-TYPE
  SYNTAX       Counter
  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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelDroppedCount (ISO
      20684-4)
    <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 }
```

16.5.6.15 Trap Row Status

```
trapMgmtRowStatus   OBJECT-TYPE
  SYNTAX       RowStatusStatic
  MAX-ACCESS   read-write
  STATUS       deprecated
  DESCRIPTION
    "<Definition> This object allows for the management of rows
      within the table.
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelRowStatus (ISO 20684-4)
    <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 }
```

16.5.6.16 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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyChannelSeqNum (ISO 20684-4)
    <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 }
```


16.5.6.17 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 }
```

16.5.7 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.
    <TableType> static
    <Superseded by> NOTIFICATION-MIB.fdNotifyFactoryTable (ISO 20684-4)
    <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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyFactoryEntry (ISO 20684-4)
    <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          Counter
                 }
```

16.5.7.1 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.
      Note: 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.
    <Superseded by> NOTIFICATION-MIB.fdNotifyFactoryRowStatus (ISO 20684-4)
    <Informative> 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 }
```

16.5.7.2 Trap Mode

```
trapMode   OBJECT-TYPE
  SYNTAX       Integer32(0..255)
  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.
      8-255: reserved (not used in a proprietary manner)
    <Superseded by> NOTIFICATION-MIB.fdNotifyFactoryAckEnabled &
      fdNotifyFactoryQueueEnabled (ISO 20684-4)
    <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 }
```

16.5.7.3 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.
      (Note that this is larger than 8 bits to allow aggregation times
      to support 5 minutes and longer).
    <Superseded by> NOTIFICATION-MIB.fdNotifyFactoryAggregationTime
      (ISO 20684-4)
    <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 }
```

16.5.7.4 Trap Counter

```
trapCounter   OBJECT-TYPE
  SYNTAX       Counter
  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.
    <Superseded by> NOTIFICATION-MIB.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 }
```

16.6 NTCIP TRAP DATA

16.6.1 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.
    <Superseded by> NOTIFICATION-MIB.fdNotificationPacket
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.2.0.1"
  ::= { ntcipTrapNotifications 1 }
```

16.7 CLEAR EVENT DATA

16.7.1 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.
    <Superseded by> LOG-MIB.fdLogsDeleteAllConfiguration &
      fdLogManagerRowStatus (ISO 20684-5)
    <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 }
```

16.7.2 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).

      Note: 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.
```

```
    <Superseded by> LOG-MIB. fdLogEventFactoryRowStatus (ISO 20684-5) &
      COND-TRIGGER-MIB.fdCondTriggerRowStatus (ISO 20684-3)
    <Informative> 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 }
```

16.7.3 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).
    <Superseded by> LOG-MIB.fdLogsClearAllLogs
    <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 }
```

16.7.4 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 }
```

16.7.5 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 }
```

16.7.6 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 }
```

16.7.7 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 }
```

16.7.8 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 }
```

16.8 COMPLIANCE GROUPS

16.8.1 Watch Block Group

```
watchBlockGroupR1 OBJECT-GROUP
  OBJECTS { maxWatchObjects,
            maxWatchBlocks,
            watchID,
            watchStatus,
            watchBlock,
            watchOID,
            watchBlockNumber,
            watchBlockStatus,
            watchBlockDescription,
            watchBlockValue }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing watch blocks.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.127.2.1"
  ::= {ntcipTrapGroups 1}
```

16.8.2 Report Block Group

```
reportBlockGroupR1 OBJECT-GROUP
  OBJECTS { maxReportObjects,
            maxReportBlocks,
            reportID,
            reportStatus,
            reportBlock,
            reportOID,
            reportBlockNumber,
            reportBlockStatus,
            reportBlockDescription,
            reportBlockValue }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing watch blocks.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.127.2.2"
  ::= {ntcipTrapGroups 2}
```

16.8.3 Trap Management Group

```
trapMgmtGroupR1 OBJECT-GROUP
  OBJECTS { trapControl,
            trapData,
            trapMgmtMaxEntries,
            trapMaxAggregationEvents,
            trapMaxAggregationSize,
            trapMgmtManagerIndex,
            trapMgmtManagerPointer,
            trapMgmtCommunityNamePointer,
            trapMgmtApplicationProtocol,
            trapMgmtTransportProtocol,
            trapMgmtPortNum,
            trapMgmtMaxRetries,
            trapMgmtRepeatInterval,
            trapMgmtDelta,
            trapMgmtQueueDepth,
```



```
            trapMgmtLinkStateStatus,
            trapMgmtAntiStreamRate,
            trapMgmtErrStatus,
            trapMgmtLostTraps,
            trapMgmtRowStatus,
            trapMgmtSeqNum,
            trapMgmtSeqNumAck,
            trapDestEnable,
            trapMode,
            trapAggregationTime,
            trapCounter }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing watch blocks.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.127.2.3"
  ::= {ntcipTrapGroups 3}
```

16.8.4 Trap Clear Group

```
trapClearGroupR1 OBJECT-GROUP
  OBJECTS { eventClearClasses,
            eventClearConfiguration,
            eventClearLog,
            clearReportObjects,
            clearReportBlockTable,
            clearWatchObjects,
            clearWatchBlockTable,
            clearTrapMgmtTable }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for clearing trap-related
      information.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.127.2.4"
  ::= {ntcipTrapGroups 4}
```

16.8.5 Trap Group

```
trapGroupR1 NOTIFICATION-GROUP
  NOTIFICATIONS { trapEvent }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The notification necessary for managing traps.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.4.127.2.5"
  ::= {ntcipTrapGroups 5}

END  -- NTCIP1201-NtcipTraps
```

Section 17
DEPRECATED RECORDING MECHANISM MANAGEMENT INFORMATION BASE (MIB) [NORMATIVE]

The text provided from Section 17.1 through the end of Section 17 (except the headings) constitutes the standard NTCIP1201-RecMech MIB.

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

This MIB is provided to allow SNMPv3 implementations to support backward compatibility with prior versions of NTCIP 1201. Prior versions of NTCIP 1201 were based on SNMPv1, which used the SMIv1 MIB structure. SNMPv3 requires data to be defined in the SMIv2 format. This section defines the official translation of the NTCIP1201-v03 MIB into the SMIv2 format so that there is an unambiguous, standardized definition of how to represent NTCIP 1201 v03 data in an SNMPv3 data packet. For example, using this MIB, a 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 MIB contained in this section can be used with SNMPv3, doing so does not resolve all known security issues. As a result, the objects contained in this MIB are marked deprecated and notes have been added to indicate either:

- The object(s) that supersede the object with similar functionality, or
- An explanation as to why there is no superseding object.

SNMPv1 implementations should refer to prior versions of NTCIP 1103 for the formal MIB definition. Copies of all versions of NTCIP MIB files are maintained on the NTCIP website.

The following table indicates which object groups and capabilities are required for each version of NTCIP for the objects contained within this MIB.

Table 15: Compliance Statement for NTCIP1201-Security MIB

Group	1103 v03
recMechGroupR1	M

17.1 HEADER

```
NTCIP1201-RecMech DEFINITIONS ::= BEGIN
IMPORTS

zeroDotZero, MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Counter32,
IpAddress, Opaque
                                                FROM SNMPv2-SMI
                                                  -- RFC 2578
MODULE-COMPLIANCE, OBJECT-GROUP
                                                FROM SNMPv2-CONF
                                                  -- RFC 2580
ITSOerString
                                                FROM FIELD-DEVICE-TC-MIB
                                                  -- ISO 20684-1 Annex A
application
                                                FROM NTCIP8004-Transportation;
```



```
recMech MODULE-IDENTITY
  LAST-UPDATED "202210010000Z"
  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. ***

     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9
     Copyright © 1996, 2006, 2022 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 in whole or in part in any form, translation
     into other languages and display are reserved by the copyright owners
     under the laws of the United States of America, the Universal Copyright
     Convention, the Berne Convention, and the International and Pan American
     Copyright Conventions.

                          Joint NEMA, AASHTO, and ITE
                     NTCIP Management Information Base
                          DISTRIBUTION NOTICE

     To the extent and in the limited event these materials are distributed
     by AASHTO/ITE/NEMA in the form of a Management Information Base ("MIB")
     module, AASHTO/ITE/NEMA extends the permission for users to make and/or
     distribute unlimited copies of the unedited MIB module, including copies
     for commercial distribution, provided that each copy made and/or
     distributed contains this Notice.

     The MIB module may be distributed with a companion MIB module that
     contains one or more invocations of the AGENT-CAPABILITIES macro that
     specifies the capabilities of a specific implementation. By convention,
     such a companion MIB module might have the same module name followed by
     "-", followed the enterprise's Internet Assigned Number Authority
     (IANA)-assigned enterprise number.

     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 OR 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, OR 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 you.

     Use of these materials does not constitute an endorsement or affiliation
     by or between AASHTO, ITE, or NEMA and you, your company, or your
     products and services.

     NTCIP is a trademark of AASHTO/ITE/NEMA."
  REVISION "202210010000Z"
  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 }
```

17.2 OBJECT IDENTITIES

17.2.1 Recording Mechanism Conformance Node

```
recMechConformance OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the high-
      resolution recording mechanism.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.127"
::= { recMech 127 }

recMechCompliances OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the high-
      resolution recording mechanism.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.127.1"
::= { recMechConformance 1 }

recMechGroups OBJECT-IDENTITY
  STATUS deprecated
  DESCRIPTION
    "<Definition> This node is an identifier used to manage the high-
      resolution recording mechanism.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.127.2"
::= { recMechConformance 2 }
```

17.3 OBJECTS

```
-- NOTE—The recording mechanism class table is presented first to ease
-- the readability of the standard.
```


17.3.1 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 }
```

17.3.2 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.
    <TableType> static
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassTable
    <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
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassEntry
    <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 }
```

17.3.2.1 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassName
    <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 }
```

17.3.2.2 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).
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassSizeLimit
    <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 }
```

17.3.2.3 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassClearDate &
      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 }
```

17.3.2.4 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassDescription
    <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 }
```

17.3.2.5 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassNumRecordings
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.2.1.5"
::= { recClassEntry 5 }
```

17.3.2.6 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassRecordingCtr
    <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 }
```

17.3.3 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}
```

17.3.4 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechMinSamplePeriod
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.4"
::= { recMech 4}
```

17.3.5 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechMaxSamplePeriod
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.5"
::= { recMech 5}
```

17.3.6 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
    <Superseded by> NTCIP1201-RecMechV2.recMechV2SamplePeriodResolution
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.6"
::= { recMech 6}
```

17.3.7 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.
    <TableType> static
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactoryTable &
      COND-TRIGGER-MIB.fdCondTriggerTable (ISO 20684-3)
    <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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactoryEntry &
      COND-TRIGGER-MIB.fdCondTriggerEntry (ISO 20684-3)
    <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 }
```

17.3.7.1 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactoryName
    <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 }
```

17.3.7.2 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.

      NOTE-A recording cannot be recorded if the RecClass has an
      recClassLimit of zero (0).
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassName
    <Informative> 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 }
```


17.3.7.3 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.
    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerMode (ISO 20684-3)
    <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 }
```

17.3.7.4 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).
    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerValue (ISO 20684-3)
    <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 }
```

17.3.7.5 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.
    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerValue2 (ISO 20684-3)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.5"
   DEFVAL  {0}
::= { recConfigEntry 5 }
```

17.3.7.6 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.
    <Superseded by> COND-TRIGGER-MIB.fdCondTriggerObject (ISO 20684-3)
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.6"
   DEFVAL  { zeroDotZero }
::= { recConfigEntry 6 }
```

17.3.7.7 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 }

    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactorySampleOID
    <Informative> The replacement object does not contrain 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 }
```

17.3.7.8 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactoryPreSamples
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.8"
   DEFVAL  {80}
::= { recConfigEntry 8 }
```

17.3.7.9 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
            (recMinSamplePeriod + recSamplePeriodResolution * n)
      where  n is integer, 0 <= n, and
      n <= (recMaxSamplePeriod-recMinSamplePeriod)/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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactorySamplePeriod
```



```
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.9"
   DEFVAL  {1000}
::= { recConfigEntry 9 }
```

17.3.7.10 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactoryMonitorOID
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.7.1.10"
   DEFVAL  { zeroDotZero }
::= { recConfigEntry 10 }
```

17.3.7.11 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.

      Note: 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactorySampleLimit
    <Informative> 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 }
```

17.3.7.12 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactoryRowStatus
    <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 }
```

17.3.7.13 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
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactoryRowStatus
    <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 }
```

17.3.8 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}
```

17.3.9 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.
    <TableType> dynamic
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingTable
    <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
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingEntry
    <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 }
```

17.3.9.1 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassName
    <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 }
```

17.3.9.2 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingIndex
    <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 }
```

17.3.9.3 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 }
```

17.3.9.4 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingTrigger
    <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 }
```

17.3.9.5 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingDate &
      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 }
```

17.3.9.6 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingStatus
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.6"
::= { recRecordingEntry 6 }
```

17.3.9.7 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingTriggerSample
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.7"
::= { recRecordingEntry 7 }
```

17.3.9.8 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.

      Note: A completed recording does not always have recConfigNumEntries
      record entries in it.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingNumSamples
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.9.1.8"
::= { recRecordingEntry 8 }
```

17.3.10 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}
```

17.3.11 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.
    <TableType> dynamic
    <Superseded by> NTCIP1201-RecMechV2.recMechV2SampleTable
    <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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2SampleEntry
    <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  }
```

17.3.11.1 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).
    <Superseded by> NTCIP1201-RecMechV2.recMechV2SampleNum
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.11.1.1"
::= { recEntry 1 }
```

17.3.11.2 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
    <Superseded by> NTCIP1201-RecMechV2.recMechV2SampleTime
    <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 }
```

17.3.11.3 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.
      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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2SampleValue
    <Informative> 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 }
```

17.3.12 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.
    <Superseded by> NTCIP1201-RecMechV2.recMechV2OwnerRecordingCtr &
      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 }
```

17.3.13 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).
    <Superseded by> NTCIP1201-RecMechV2.recMechV2ClassRowStatus,
      recMechV2OwnerClearAllClasses, & adminRecMechClearAllClasses
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.13"
::= { recMech 13 }
```

17.3.14 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).
    <Superseded by> NTCIP1201-RecMechV2.recMechV2FactoryRowStatus,
      recMechV2OwnerClearAllFactories, adminRecMechClearAllFactories,
      COND_NOTIFICATIONS-MIB.fdCondTriggerRowStatus (ISO 20684-3)
    <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 }
```

17.3.15 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.

      Note: a device shall immediately start collecting new pre-event
      records for all active recording configurations.

      Upon performing the action requested, the device shall
      SET this object to zero (0). A GET shall always return zero (0).
    <Superseded by> NTCIP1201-RecMechV2.recMechV2RecordingDelete,
```

```
      recMechV2OwnerClearAllRecordings, & adminRecMechClearAllRecordings
    <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.15"
::= { recMech 15 }
```

17.4 COMPLIANCE GROUPS

17.4.1 Watch Block Group

```
recMechGroupR1 OBJECT-GROUP
  OBJECTS { maxRecClasses,
            recClassNumber,
            recClassLimit,
            recClassClearTime,
            recClassDescription,
            recClassNumRecordings,
            recClassRecordingCounter,
            maxRecConfigs,
            recMinSamplePeriod,
            recMaxSamplePeriod,
            recSamplePeriodResolution,
            recConfigID,
            recConfigClass,
            recConfigMode,
            recConfigCompareValue,
            recConfigCompareValue2,
            recConfigCompareOID,
            recConfigRecordOID,
            recConfigTriggerPoint,
            recConfigSamplePeriod,
            recConfigSampleOID,
            recConfigNumEntries,
            recConfigAction,
            recConfigStatus,
            maxRecordings,
            recordingClass,
            recordingNumber,
            recordingID,
            recordingConfigID,
            recordingTriggerTime,
            recordingStatus,
            recordingTriggerRecNum,
            recordingNumEntries,
            maxRecEntries,
            recEntryNumber,
            recSampleTime,
            recValue,
            numRecordings,
            recClearClasses,
            recClearConfigurations,
            recClearRecordingData }
  STATUS     deprecated
  DESCRIPTION
    "<Definition> The objects necessary for managing the high-resolution data
      recording mechanism.
     <Object Identifier> 1.3.6.1.4.1.1206.4.1.1.7.9.127.2.1"
  ::= {recMechGroups 1}
END -- NTCIP1201-RechMech
```


Annex A
CONCEPT OF OPERATIONS
[NORMATIVE]

Annex A provides examples of how a management station may interface with a device complying with NTCIP 1201. Any device claiming conformance to the current NTCIP 1201 features (download transaction or recording mechanism) shall support the exchanges as shown. However, the flexible design of the NTCIP protocols allows a large number of other possibilities, and these figures do not limit any other requirements of NTCIP standards. These diagrams promote a common understanding of how systems may be designed to increase the likelihood of interchangeability in deployed systems.

A.1 DOWNLOAD TRANSACTION FEATURE

The intent of the download transaction feature is that a management station has a need to download several inter-related parameters to the controller. Because the parameters are inter-related, the parameters need to be set simultaneously for the controller to validate the set operation (e.g., the download may consist of a set of parameters, whose sum shall equal the sum of another set of parameters. and the management station wishes to change the sum for both sets). While normal SNMP set operations simultaneously set all objects contained within the request, it does not normally perform consistency checks on the multiple values being set and has limited size constraints. The database transaction feature was designed to overcome these limitations.

The parameters that require the use of the transaction mode are device-specific. Some devices may not require support of the transaction feature, while other devices may require SET operations on any parameter object to be set within the transaction mode.

When used, the feature allows a device to buffer a series of set operations on database parameters and to implement all operations simultaneously to properly perform controller consistency checks.

The normal, fault-free process is shown in Figure 1.

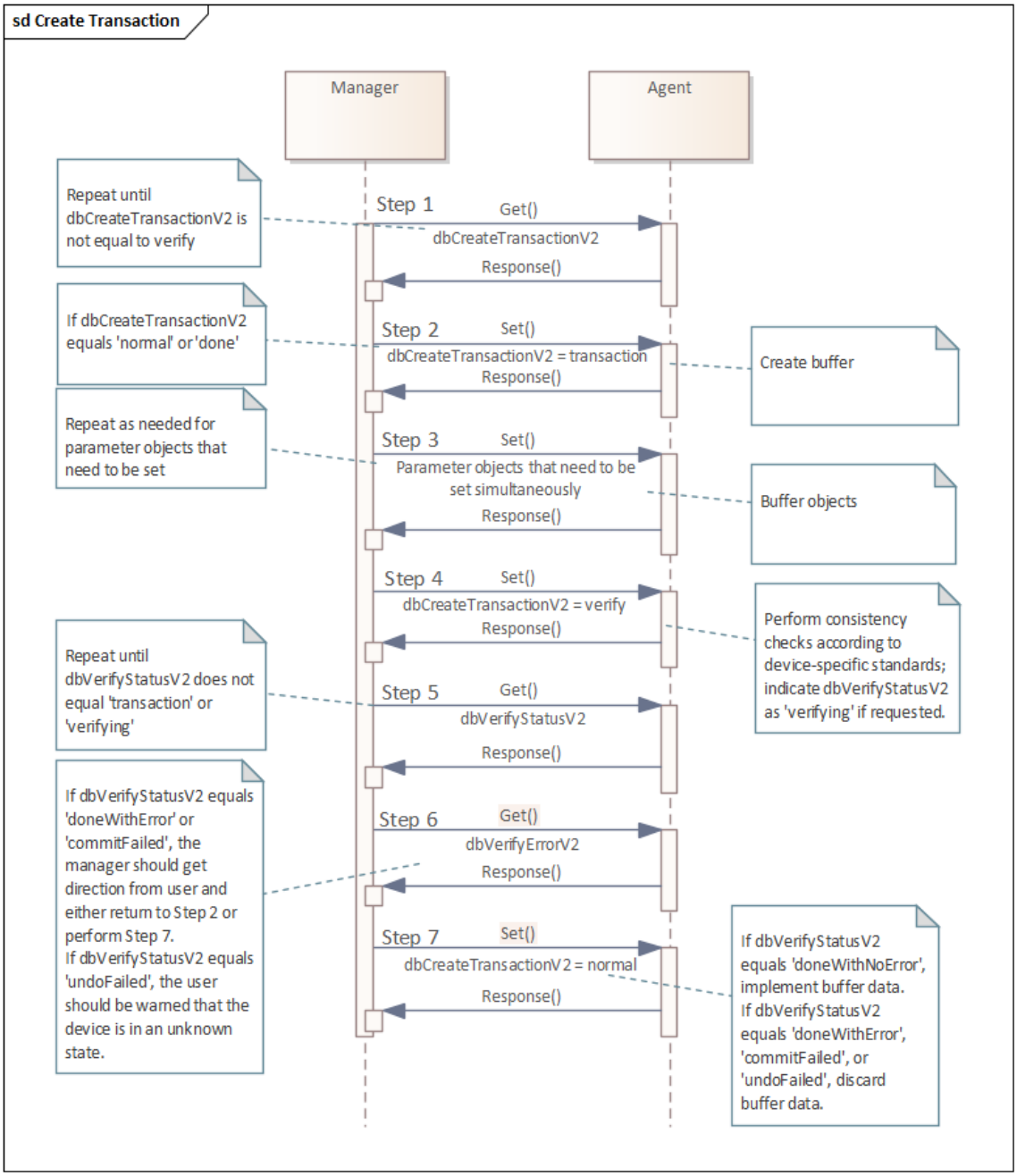

Figure 1 Download Transaction Process Sequence Diagram

Within this mode, the controller operates as a state machine as described in the definition of dbCreateTransactionV2 (see Section3.3.1). Figure 2 supplements this definition and is a state diagram that provides a formal Unified Modeling Language (UML) representation of the state machine. See Annex B.2.1 for a Controller Class Diagram in UML notation.

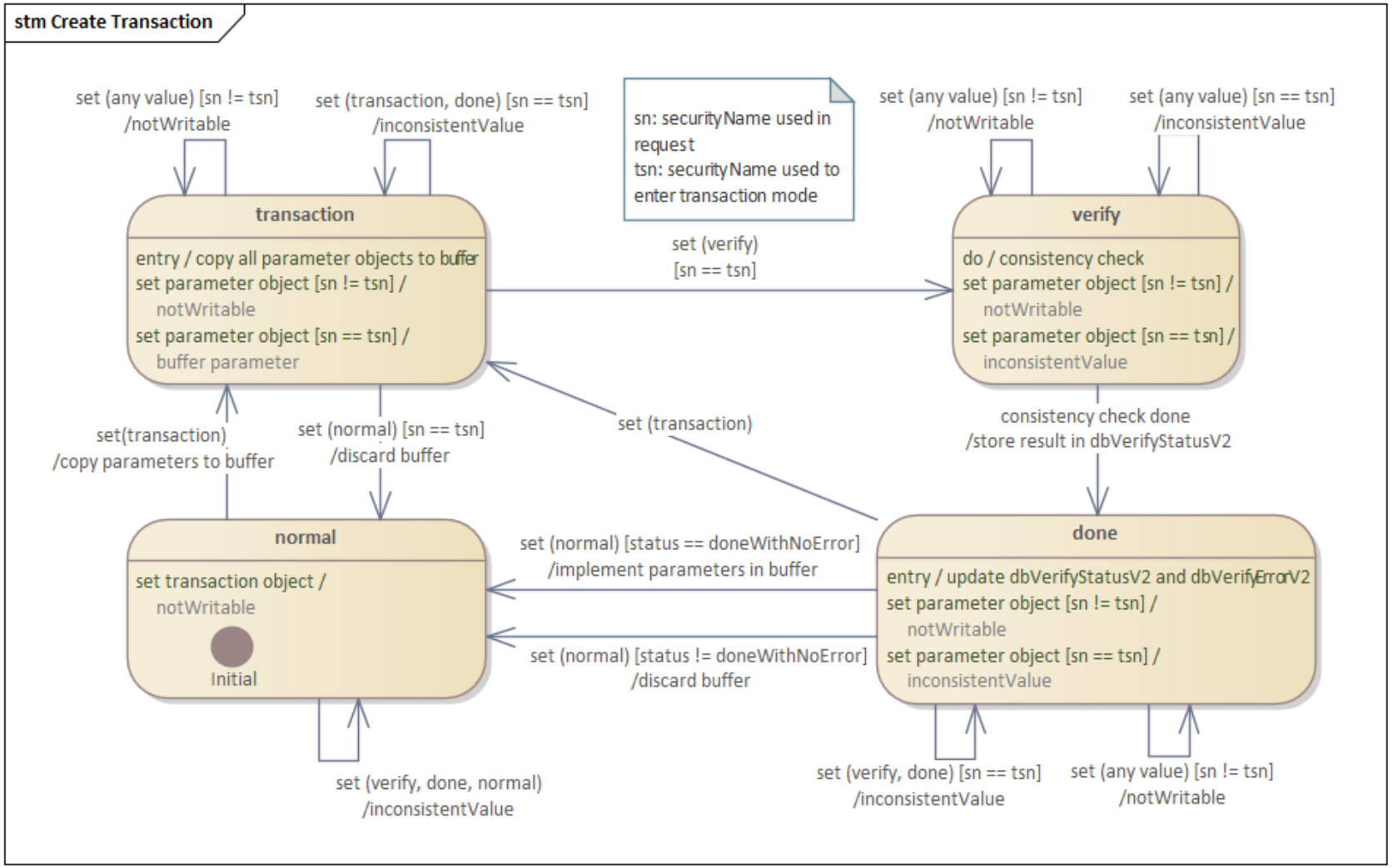

Figure 2 Controller State Diagram

A.2 HIGH-RESOLUTION DATA RECORDING

A.2.1 Overview

The high-resolution data recording facility (sometimes referred to simply as the *recording mechanism*) provides a programmable method of collecting high-resolution records of the history of values of selected objects. These records are collected and retained under user-specified conditions. While this facility may appear to be very similar to the logging feature defined in ISO 20684-5, it provides a distinct function. The logging feature captures a snapshot of information when a trigger fires and stores the information in a log. In contrast, the recording-mechanism feature captures multiple snapshots, potentially at very short intervals and stores a series of these snapshots when a trigger fires (potentially recording snapshots prior to the trigger firing, if so configured). Thus, whereas the logging feature produces a single snapshot for each event, the recording mechanism captures a short video for each event.

As with the logging feature, the manager is responsible for retrieving the data in a timely manner; however, since the recording mechanism can record a large amount of data in a short period of time, retrieving the data in a timely manner is even more important. To enable this, the ISO 20684-3 trigger used to initiate a recording can also be used to send a notification to the manager. Alternatively, a second trigger can be defined to create a notification when the status (recMechV2RecordingStatus) of any recording changes to 'complete'.

When a recording is active and the recording is configured to include pre-trigger samples, the device constantly samples the value of the data object to be recorded into a circular buffer, at a configurable sampling interval. When the trigger fires and the recording action is called, a user-specified fraction of the samples gathered prior to the trigger event are retained, and further samples are gathered until the recording is complete. Thus, the recording mechanism allows the management station to gather data sampled before, during, and after an event of interest. Multiple recordings may be active simultaneously and the memory available for recordings may be allocated flexibly to different classes under user control.

The high-resolution data recording mechanism is managed by the following MIB tables and associated objects that specify device-specific limits on the recording mechanism:

a. adminRecMechV2Table and associated objects – allows a manager with administrator rights to monitor the recording mechanism and configure rights for each owner of recordings. This includes the ability to set size limits on owners and to clear portions or all of the recording mechanism information.
b. recMechV2OwnerTable – allows an owner to monitor the number of recordings and to clear portions or all of the recording mechanism information.
c. recMechV2ClassTable – specifies the characteristics of the classes into which recordings may be classified. This includes properties such as the maximum number of recordings of each class that are allowed to exist.
d. ISO 20684-3 actionTable – table used to connect a trigger (e.g., ISO 20684-3 defines the conditional, day plan, and schedule triggers) firing to the desired action(s) (e.g., the ISO 20684 series includes notifying, logging, and commanding and this document defines recording)
e. recMechV2RecordingFactoryTable – specifies the data to be sampled for each recording and when to sample.
f. recMechV2RecordingTable – contains a row for each active recording, describing the state of the recording and indicating where the recorded data may be found in the recMechV2SampleTable.
g. recMechV2SampleTable – contains the data gathered for all active recordings that have been triggered and have not yet been overwritten.
h. Associated objects – identify the capabilities of the recording mechanism so that owners can properly configure recordings.

A.2.2 Administration

The administrator can:

1. Determine the number of recordings stored
2. Determine the total number of recordings completed since the last reboot

3. Clear all recording configuration information
4. Clear all recording classes (i.e., everything except the owner configuration)
5. Clear all recording factories (i.e., everything except the owner and classes)
6. Clear all recordings
7. Configure the amount of storage space to allot to any owner
8. Enable and diable the recording mechanism for owners

Clearing the recording mechanism is hierarchical in nature and this provides an efficient method for clearing lower levels. For example, clearing a recording clears the entry in the Recording Table and all associated entries in the Sample Table. Similarly, if one clears a recording factory, its information is cleared from the Recording Factory Table, the Recording Table, and the Sample Table. This same hierarchy logic applies to the recording class and overall information as well. Clearing a configuration shall also delete all associated "pretrigger" and "triggered" recordings.

Clearing a specific configuration at a particular level of hierarchy can be achieved by setting the appropriate RowStatus object. For example, setting recMechV2FactoryRowStatus object for the desired recording factory to 'destroy' shall cause the associated factory, recordings, and samples to be deleted.

A.2.3 Recording Owner

A recording owner can:

1. Determine the number of recordings stored for the owner
2. Determine the total number of recordings completed for the owner since the last reboot
3. Clear all recording information configured by the owner
4. Clear all recording factories for the owner
5. Clear all recordings for the owner

Clearing the recording mechanism using the owner objects is hierarchical in nature as described for the administrator, the only difference is that the scope is limited to the recording configuration owned by the owner. Likewise, the owner should have write access to the RowStatus object and can clear configuration on a detailed level as well.

A.2.4 Recording Classes

A recording owner can:

1. Create, delete, enable, and disable new recording classes
2. Define a description for a recording class
3. Define the maximum amount of memory to allow for the class
4. Clear recordings older than a specified date/time.
5. Determine the number of recordings stored for the class
6. Determine the number of recordings completed since the last reboot
7. Configure the type of memory to use for the recording class (e.g., volatile, non-volatile)

For the recording mechanism to function effectively, it is recommended that each class be configured with enough memory to support three full recordings (i.e., representing one that is about to be cleared, a newly completed recording, and a recording in progress). The counters can be used by a manager to determine when a new recording is available (complete) for a class and also enables it to determine if recordings have been lost (i.e. over-written due to class recording limit). A manager can clear recordings of a class by setting the clear date/time objects. Any completed recording that has a trigger date/time less than or equal to this value shall be removed from the recording table; however, incomplete recordings shall not be removed from the recording table. The date and time objects should be set within the same set operation and agent implementations should ensure that existing recordings are not cleared until the end of such a joint set operation.

A.2.5 Recording Factory

A recording owner can:

1. Create, delete, enable, and disable a recording factory, which is associated with a recording class

2. Assign a distinct name for each recording factory
3. Configure the object value to be recorded by the factory, including:
    a. The context of the object (i.e., if the controller manages more than one device)
    b. The identifier of the object
4. Configure the percentage of samples to be collected prior to the trigger event
5. Configure the when the object value should be sampled by specifying either
    a. the time interval at which to sample the value
    b. an object that will trigger a sampling when it changes
6. Configure the number of samples to be included within a recording
7. Configure the type of storage to be used for the factory configuration and its recordings

The recording samples are obtained using the same security parameters used to activate the recording factory. As such, these parameters are stored within the controller but are not accessible.

Once a recording is properly configured and enabled, it begins collecting records if any samples are requested prior to the trigger event. The pre-trigger samples are stored in a circular buffer until the trigger is activated; it is implementation dependent whether these samples are accessible from the sample table.

Once triggered, sample of the object value continues (or begins if only post-trigger samples are recorded) until the trigger point is at the user specified location in the recording or the maximum number of samples have been obtained.

A configuration cannot change if it is associated with completed recordings; however, clearing a configuration clears all of its recordings.

A.2.6 Recordings

A recording owner can:

1. Identify the trigger that caused the recording
2. Identify the date and time at which the recording was triggered
3. Identify the status of the recording
4. Identify the first sample within the recording that was retrieved after the trigger firing
5. Identify the number of samples within the recording
6. Delete the recording

Upon power restoration, any triggered but incomplete recording shall be marked complete. When a new recording is requested (e.g., by a trigger firing), the agent shall store any previously collected pre-trigger samples per its configuration settings and shall start collecting pre-trigger samples for a potential future recording. A recording factory shall be able to manage multiple "triggered" (i.e., not "complete") recordings simultaneously. The number of “triggered” recordings for a recording factory is only limited by the configured size limit defined for the class and the other triggered recordings by other recording factories for the same class.

A.2.7 Recording Samples

For each recording, a recording owner can:

1. Determine the time that each sample was taken
2. Determine the object value recorded at the time that the sample was taken

Upon power restoration, an agent shall delete any pre-trigger samples that are not associated with a recording that is either "triggered" or "complete". When a new sample causes the memory available for the recording class or owner to be exceeded, the agent shall delete the oldest recording of any class of the same owner from the recMechV2RecordingTable along with all associated entries in the recMechV2SampleTable, and then capture the latest sample.

A.2.8

A recording owner can:

1. Determine the minimum sample period that is allowed by the agent
2. Determine the maximum sample period that is allowed by the agent
3. Determine the sample period resolution supported by the agent.

Agents are required to support sample period resolutions of 50 milliseconds or less.

Managers should be aware of these object values so that they can properly configure their recordings. Allowable sample periods are restricted to the formula:

(recMinSamplePeriod + recSamplePeriodResolution * n)

where:
n is an integer,
0 <= n, and
n <= (recMaxSamplePeriod - recMinSamplePeriod) / recSamplePeriodResolution

Annex B
OBJECT TREE AND UML CLASS DIAGRAMS
[INFORMATIVE]

Annex B 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 B 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.

B.1 OBJECT IDENTIFIER NAMING TREE

Figure 3 depicts the location of nodes representing the MODULE-IDENTITY macros contained within this document.

Figure 4 through Figure 22 extend the tree shown in Figure 3 by showing the relevant nodes defined in each of the MIBs contained in Section 3 through Section 17.

In each diagram, gray boxes represent simple nodes, green boxes represent MIBs (i.e., defined with MODULE-IDENTITY), boxes shown in blue represent object types, boxes shown in orange represent notification types. An ellipsis (…) represents a repetition of the initial characters for the given context. For example, within the recMechV2 MIB, the ellipsis either represents the text "recMechV2" or, when occurring within a specific table, the root of the table name (e.g., "recMechV2Class").

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.

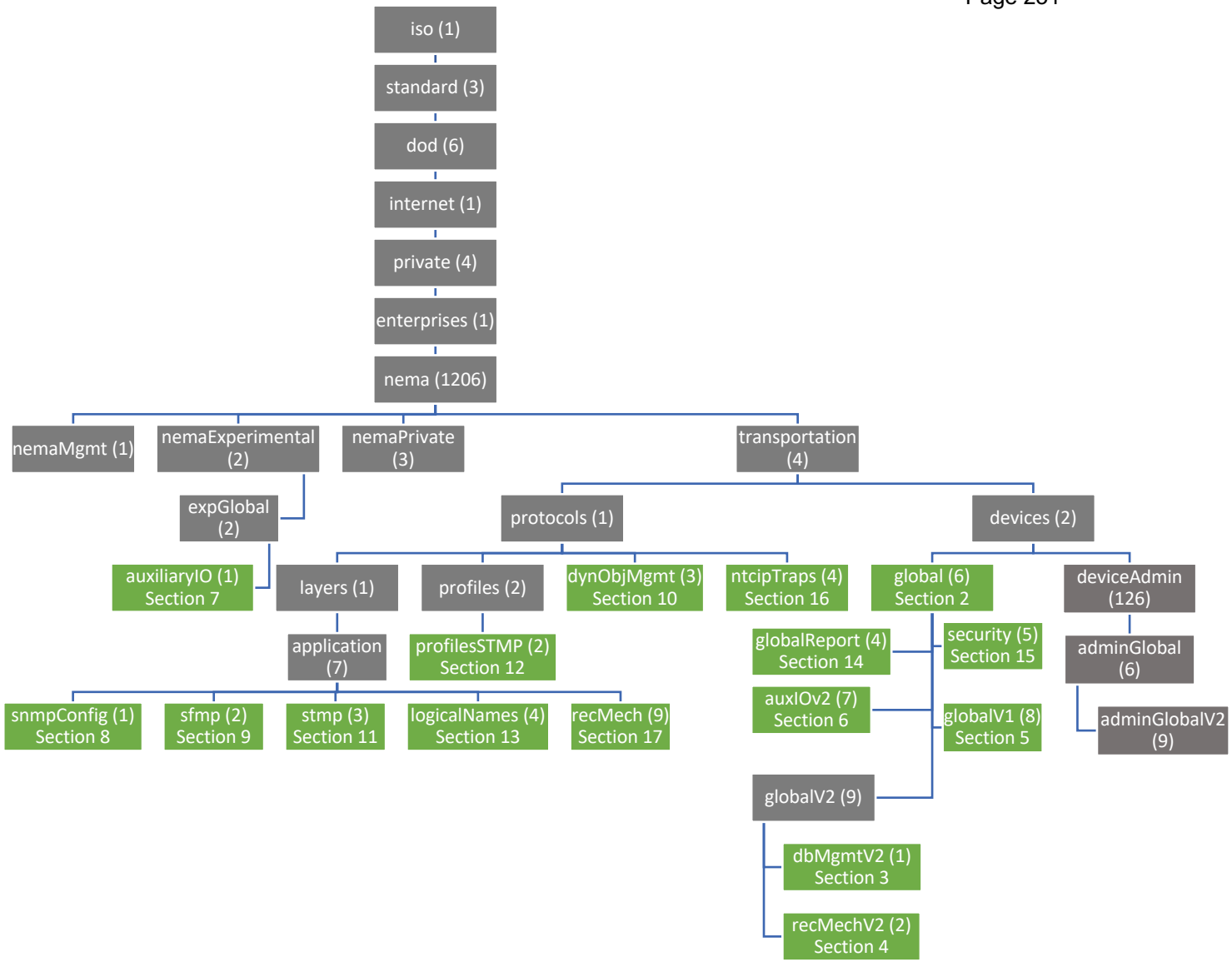

Figure 3 MODULE-IDENTITY nodes

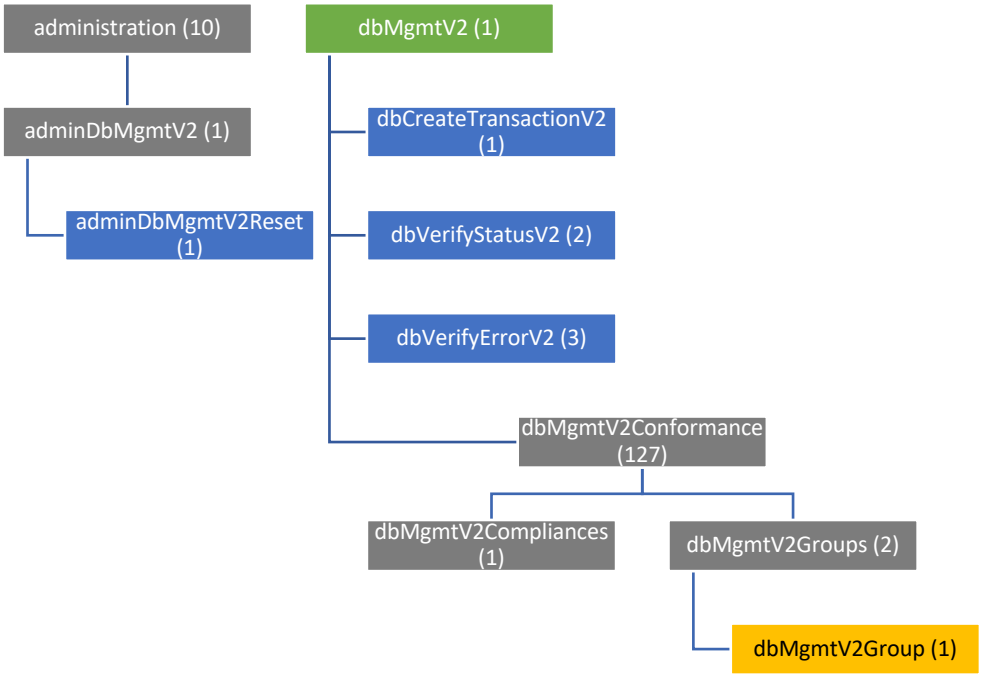

Figure 4 Nodes Defined in NTCIP1201-DbMgmtV2 MIB

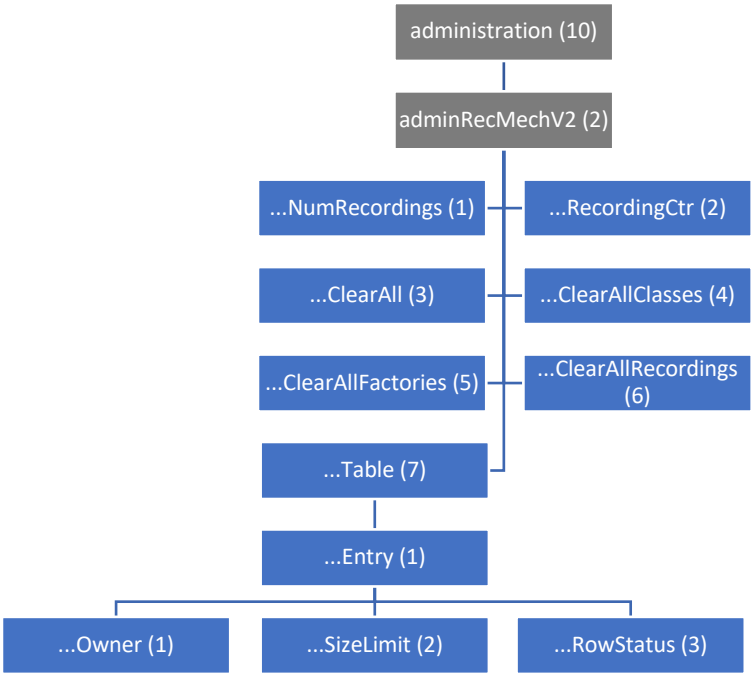

Figure 5 Nodes Defined in NTCIP1201-RecMechV2 MIB (Part 1 of 2: Administrative Assignments)

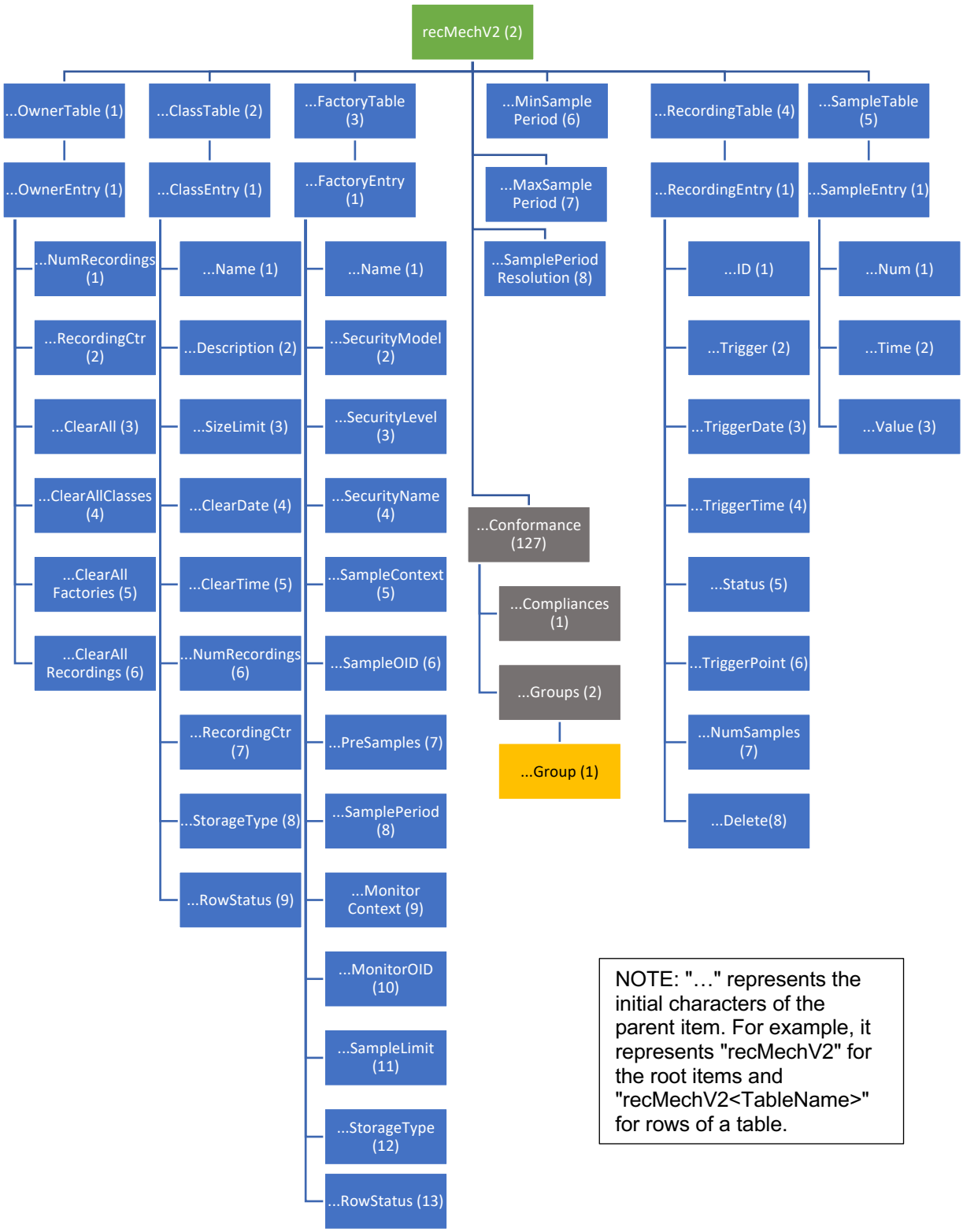

Figure 6 Nodes Defined in NTCIP1201-RecMechV2 MIB (Part 2 of 2: Core Functionality)

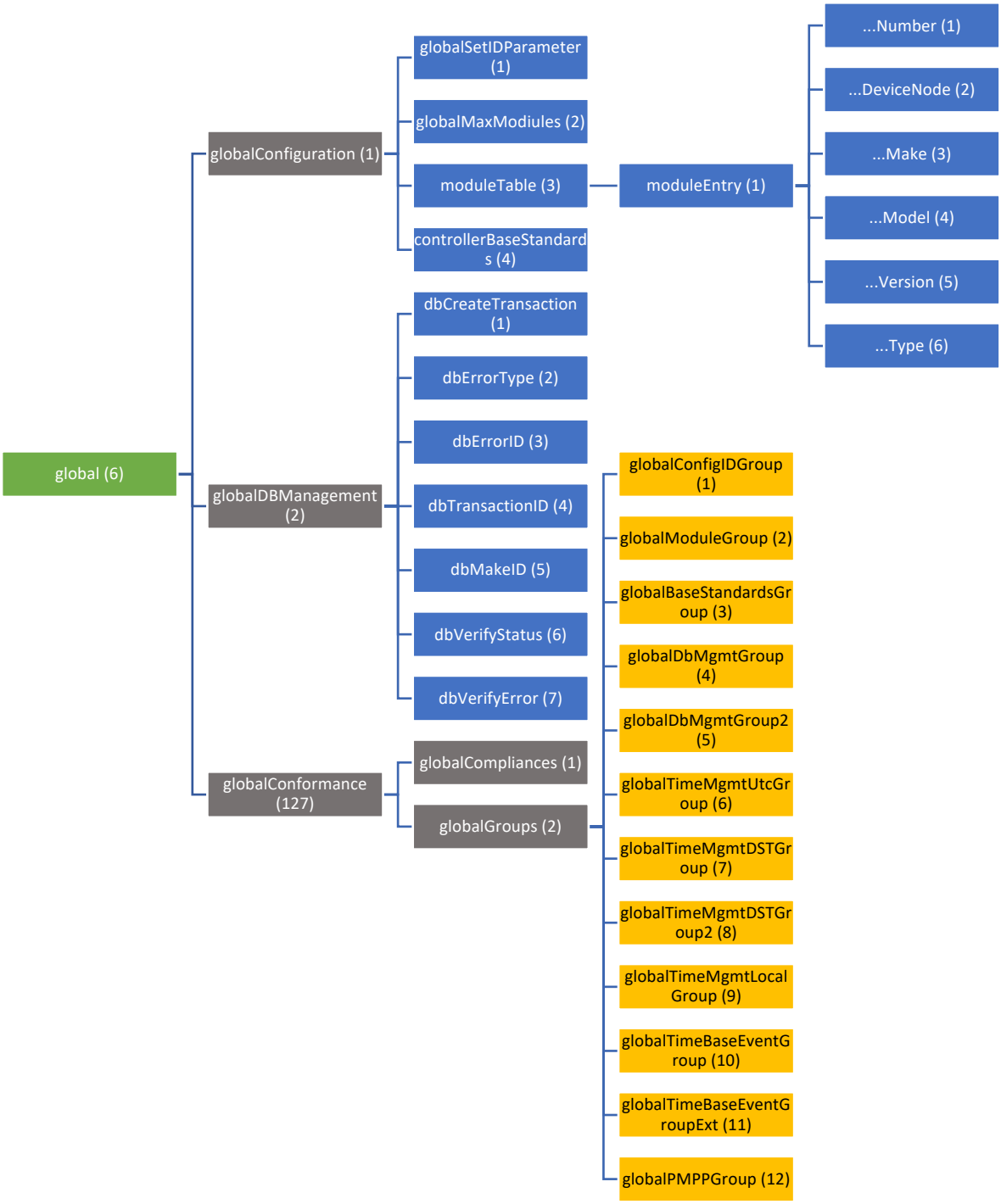

Figure 7 Nodes Defined in NTCIP 1201-Global MIB (Part 1 of 3: Base Objects)

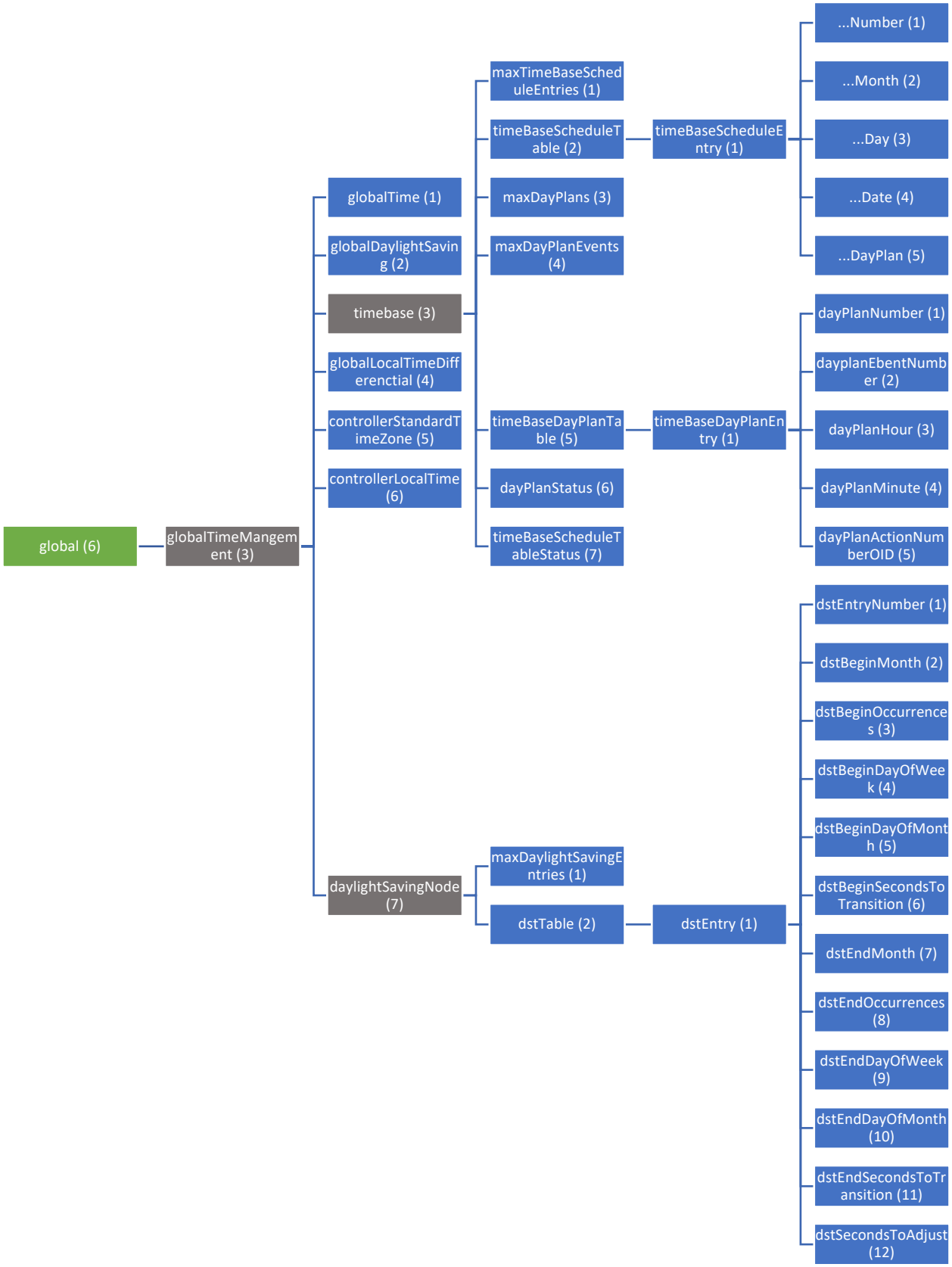

Figure 8 Nodes Defined in NTCIP 1201-Global MIB (Part 2 of 3: Time Management)

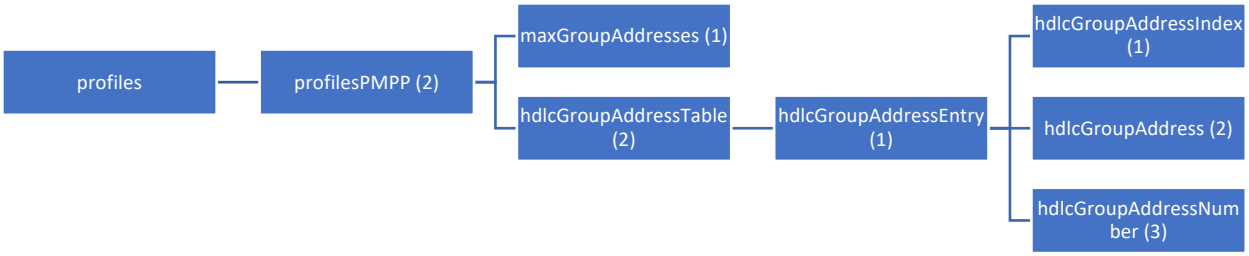

Figure 9 Nodes Defined in NTCIP 1201-Global MIB (Part 3 of 3: PMPP Objects)

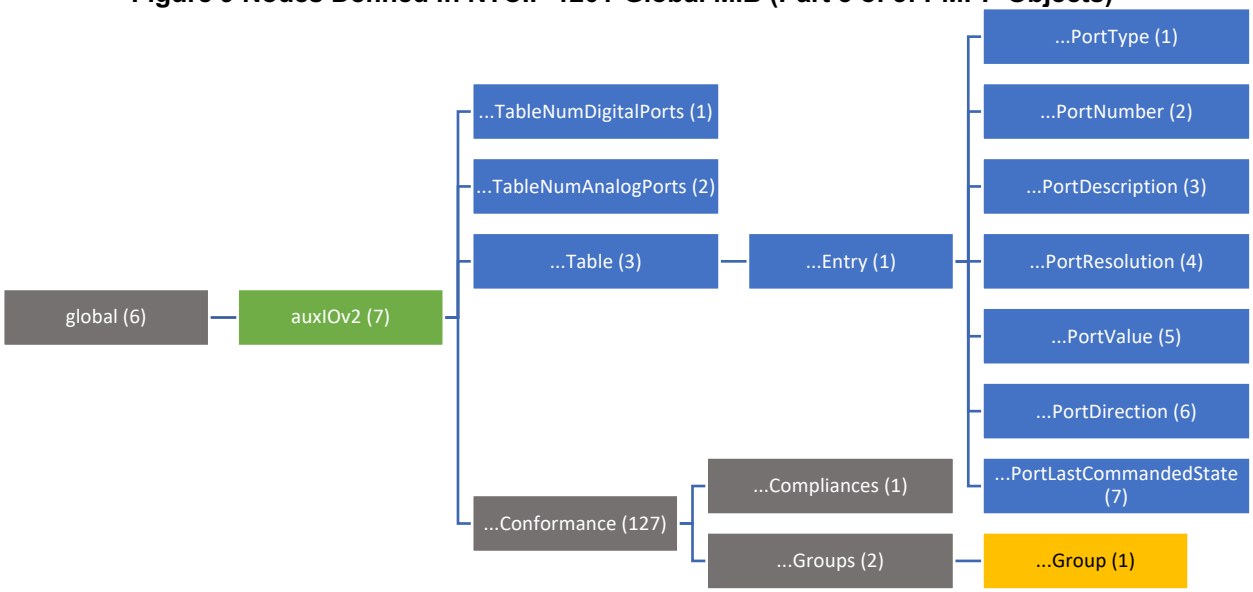

Figure 10 Nodes Defined in NTCIP1201-AuxIOv2 MIB

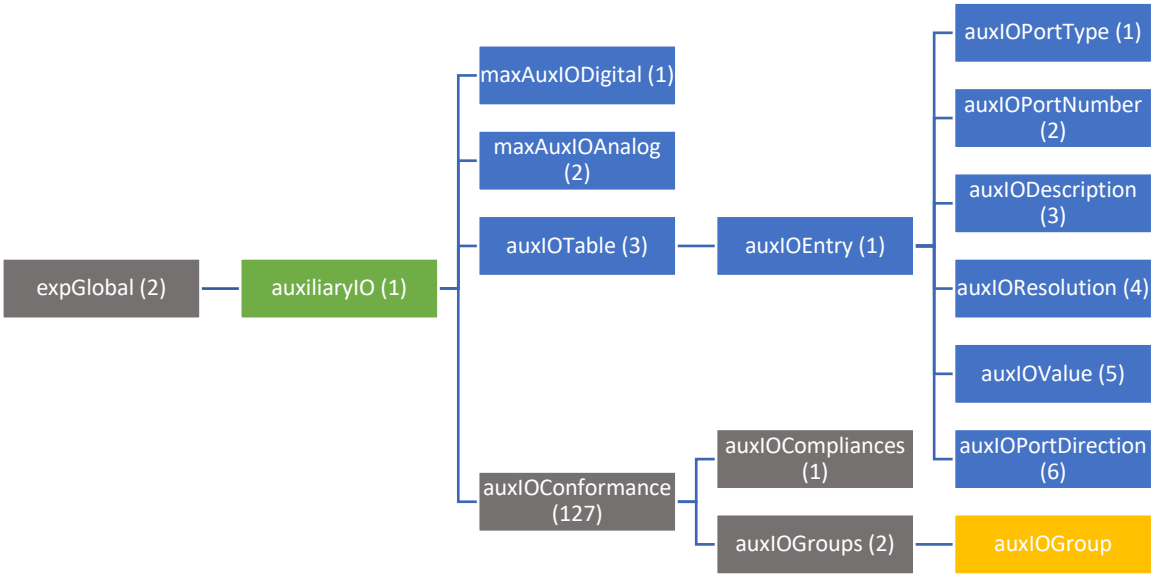

Figure 11 Nodes Defined in NTCIP1201-AuxIO MIB

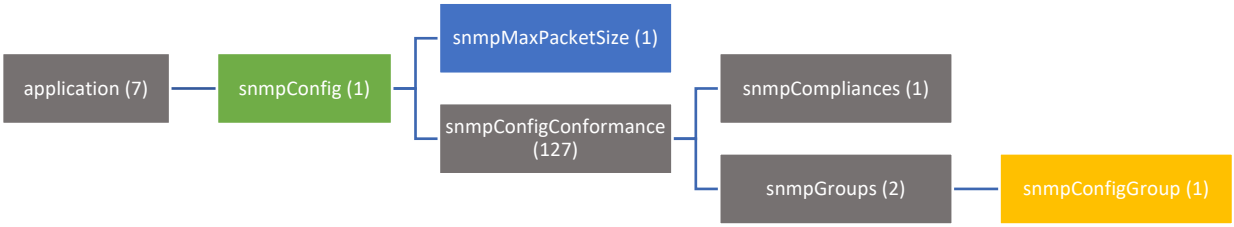

Figure 12 Nodes Defined in NTCIP1201-SNMPConfig MIB

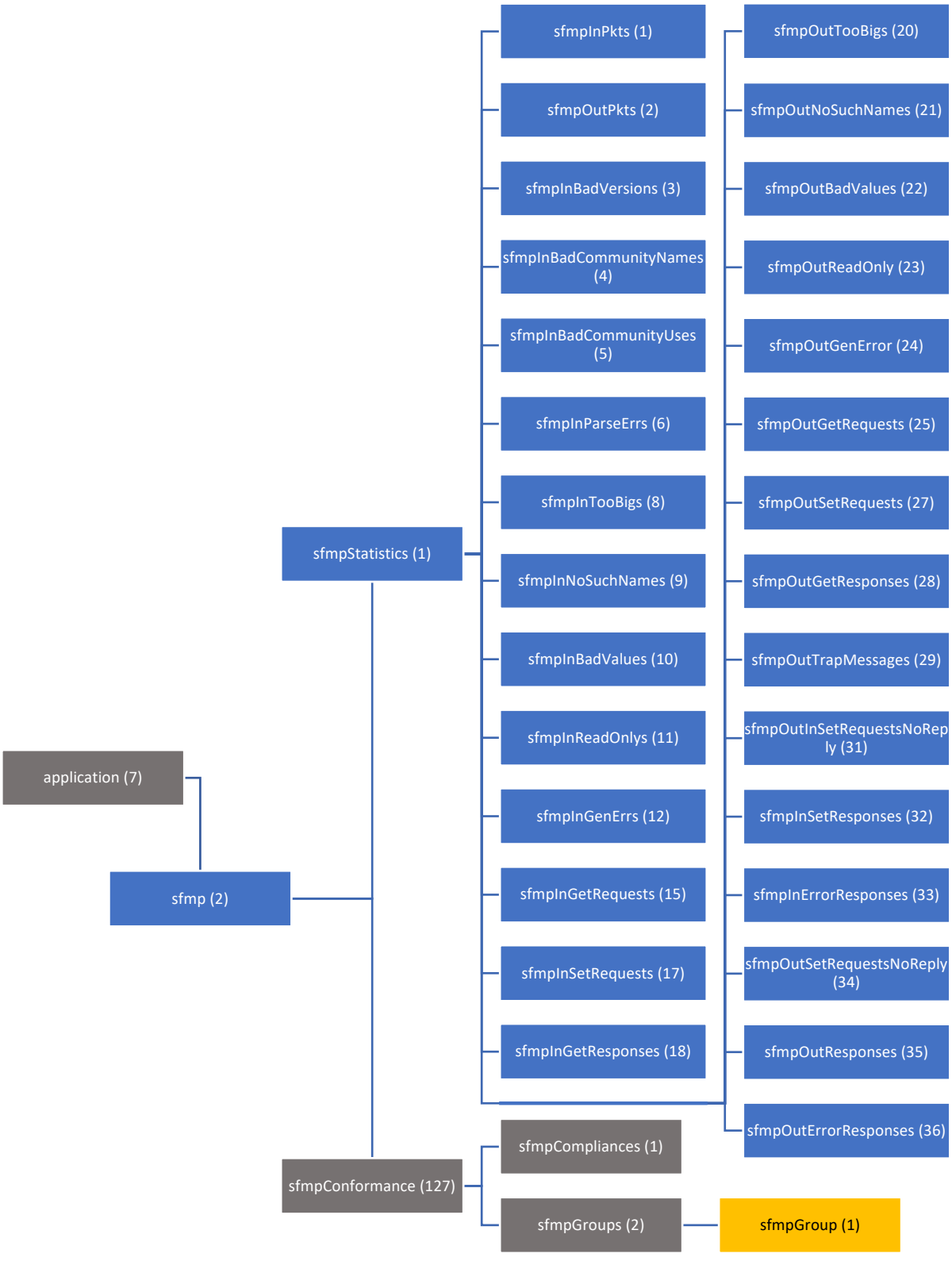

Figure 13 Nodes Defined in NTCIP1201-SFMP MIB

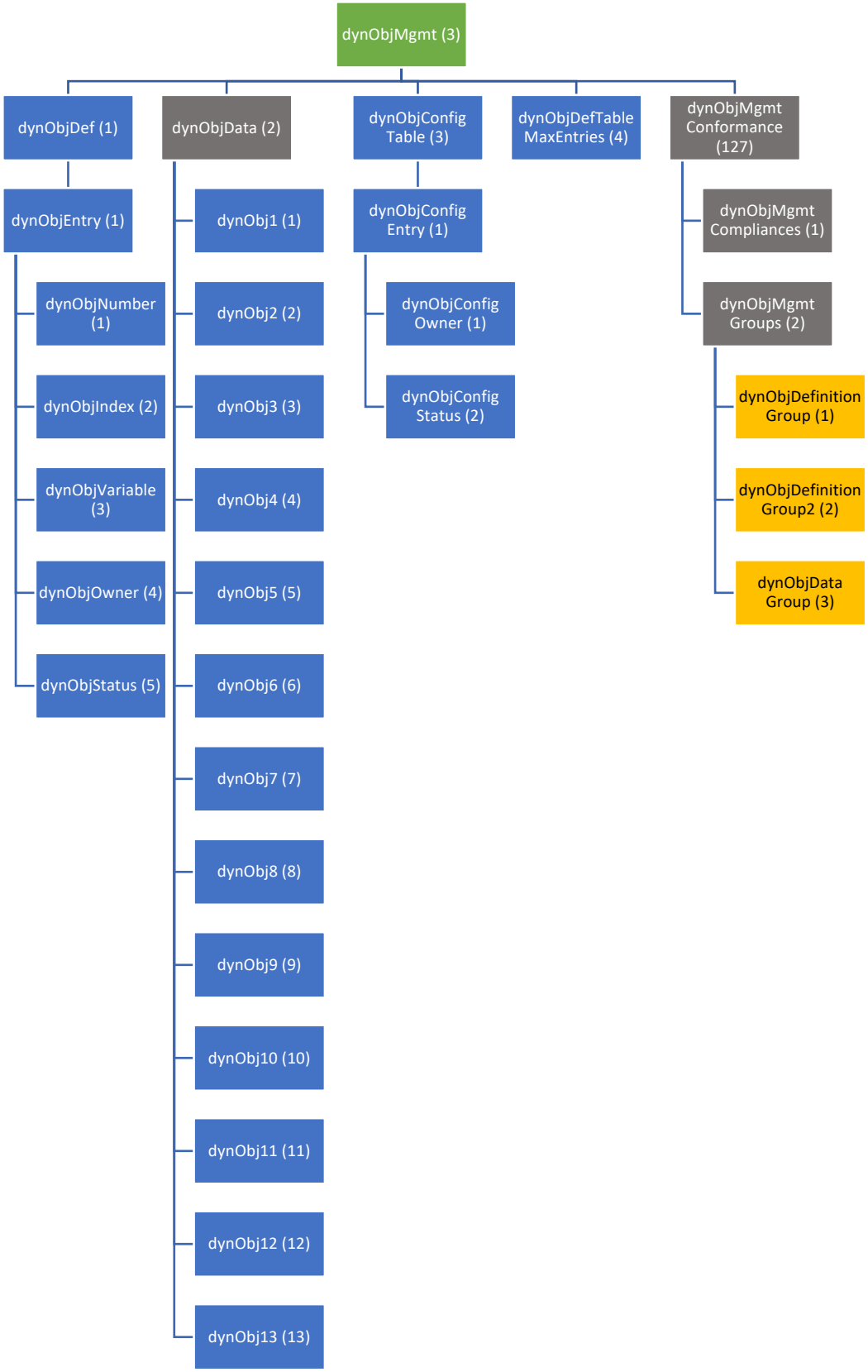

Figure 14 Nodes Defined in NTCIP1201-DynObjMgmt MIB

Figure 15 Nodes Under dbMgmtV2

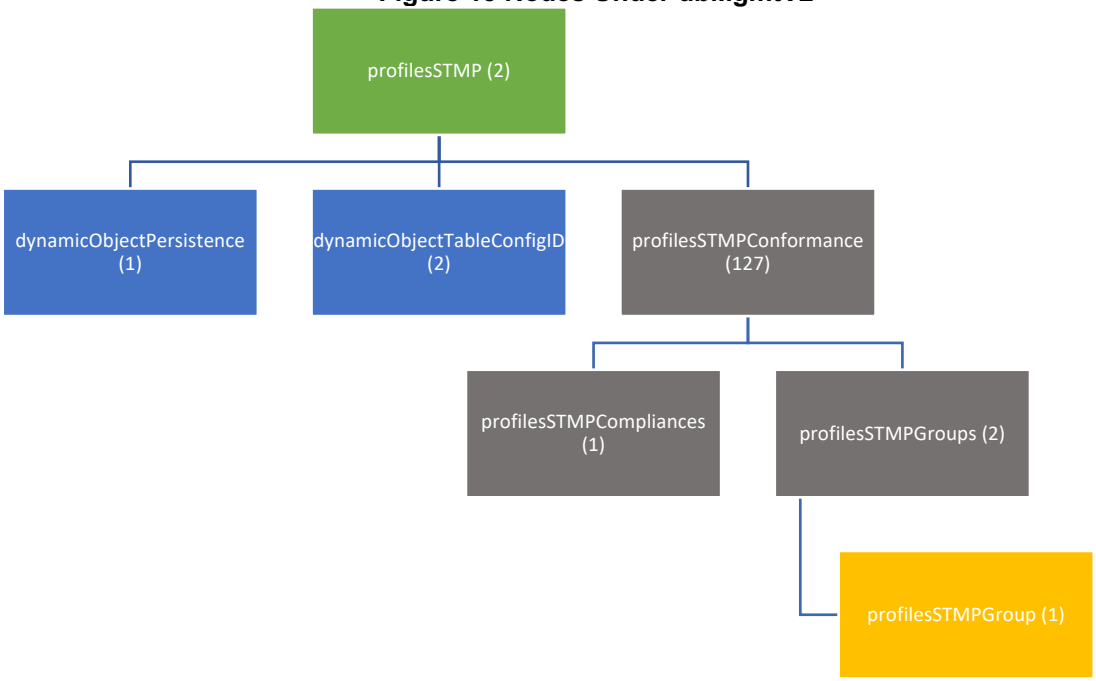

Figure 16 Nodes Under dbMgmtV2

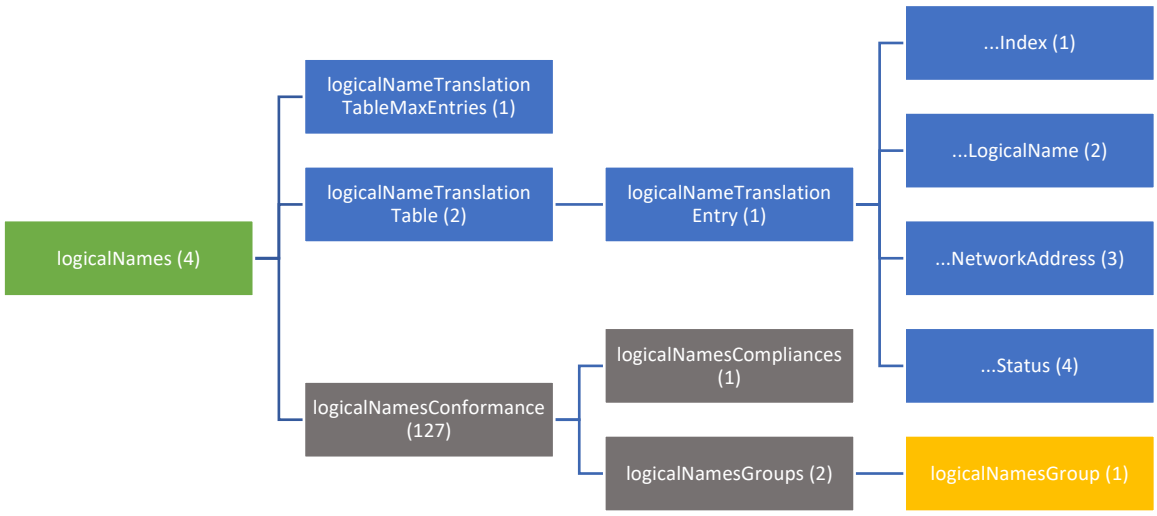

Figure 17 Nodes Under dbMgmtV2

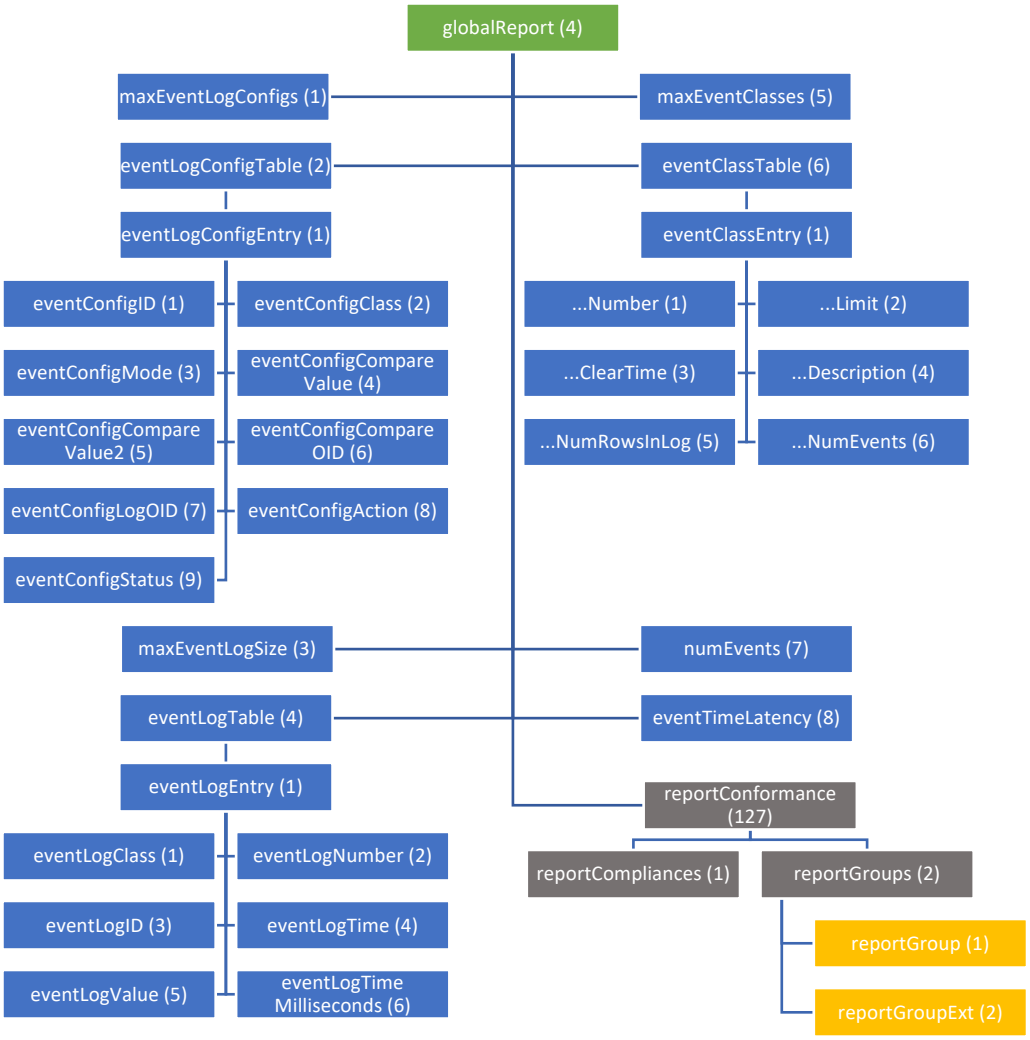

Figure 18 Nodes Defined in NTCIP1201-Report MIB

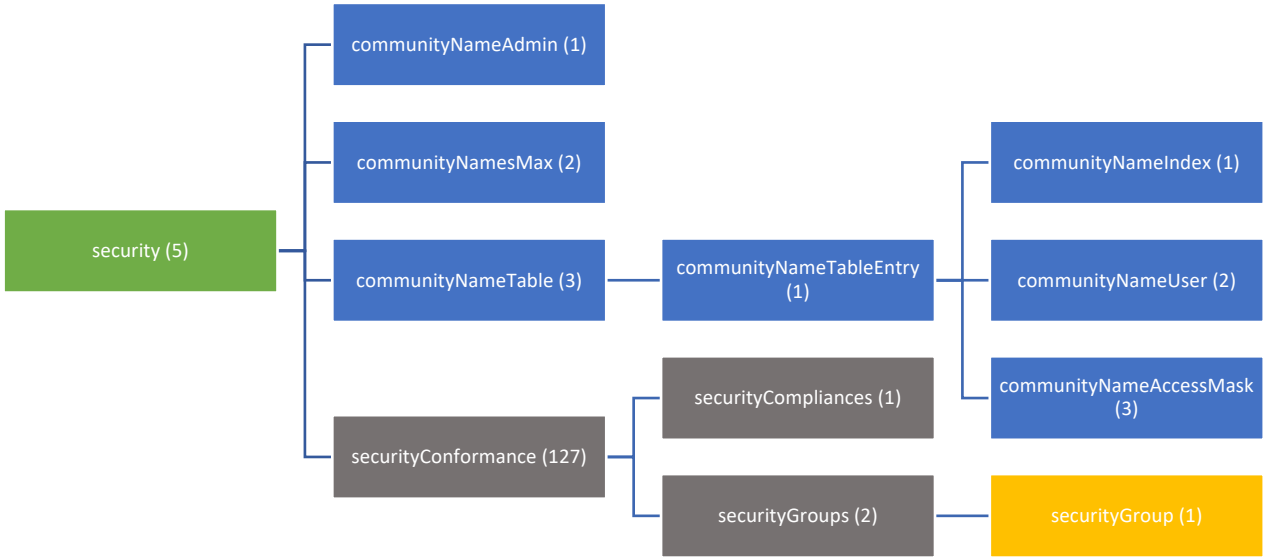

Figure 19 Nodes Defined in NTCIP1201-Security MIB

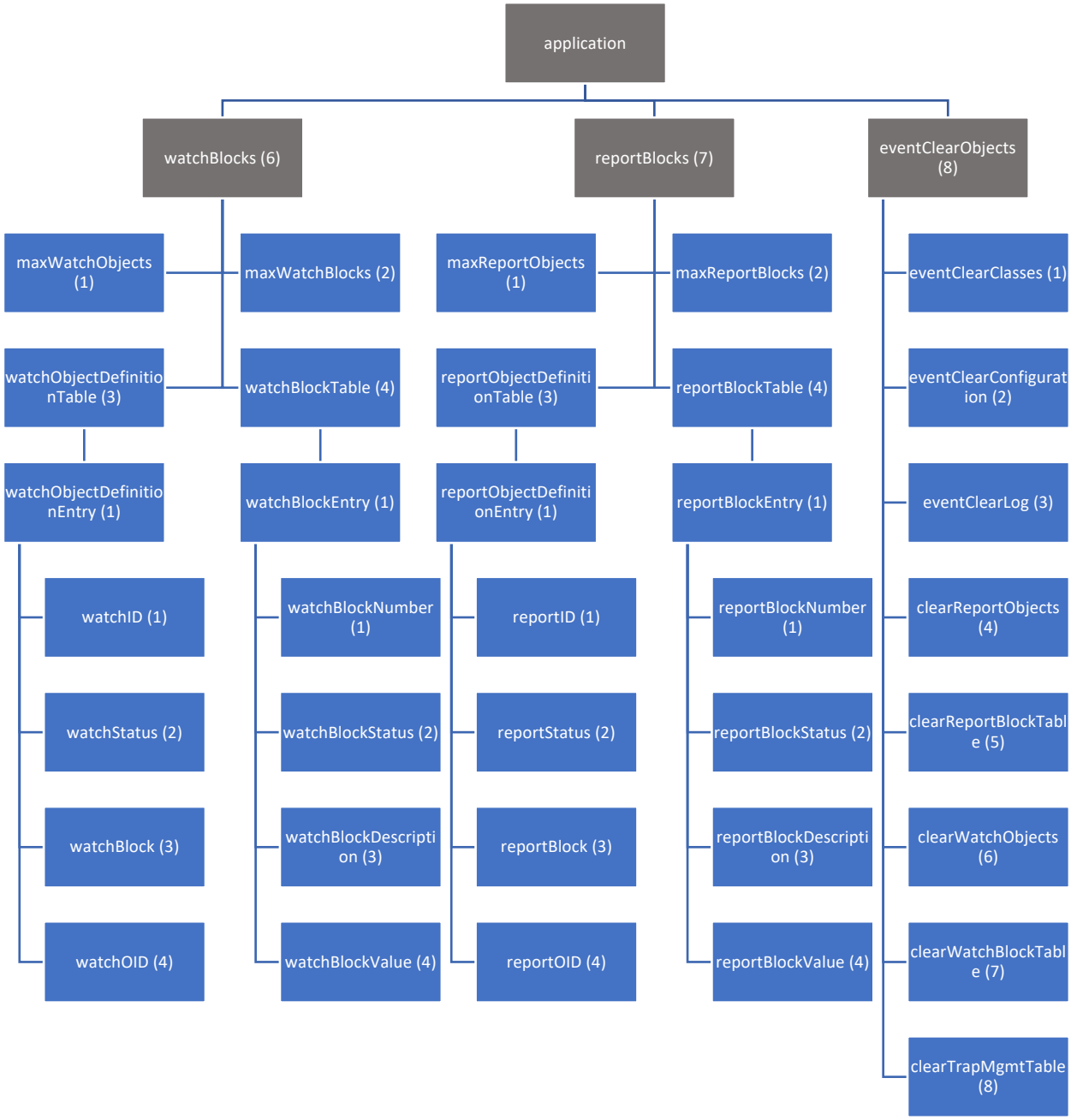

Figure 20 Nodes Defined in NTCIP1201-NtcipTraps MIB (Part 1 of 2: Application Objects)

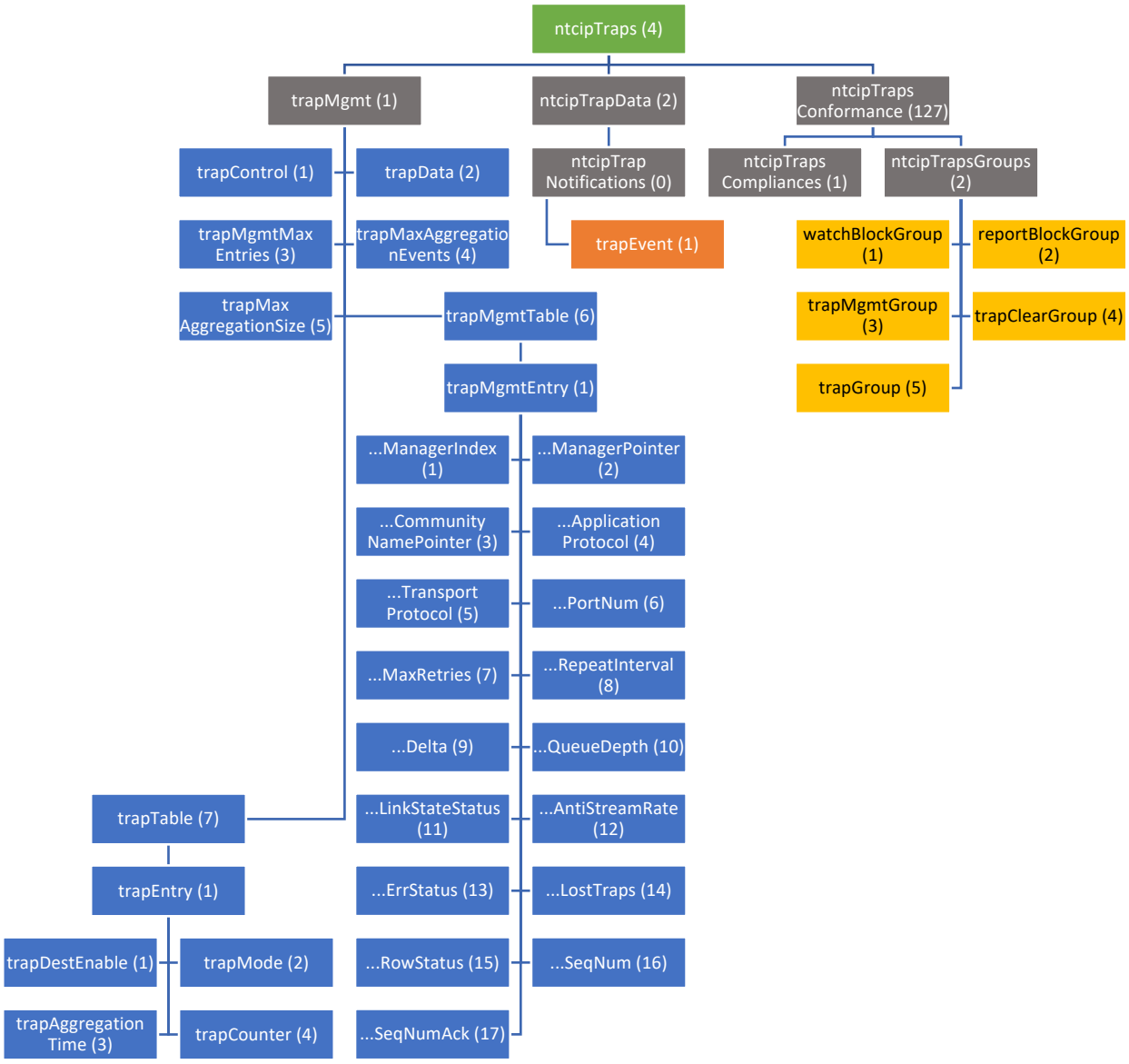

Figure 21 Nodes Defined in NTCIP1201-NtcipTraps MIB (Part 2 of 2: Trap Management)

Figure 22 Nodes Defined in NTCIP1201-RecMechV2 MIB

B.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.

B.2.1 Controller Class Diagram

Figure 23 depicts the high-level components of data stored within a controller in UML notation.

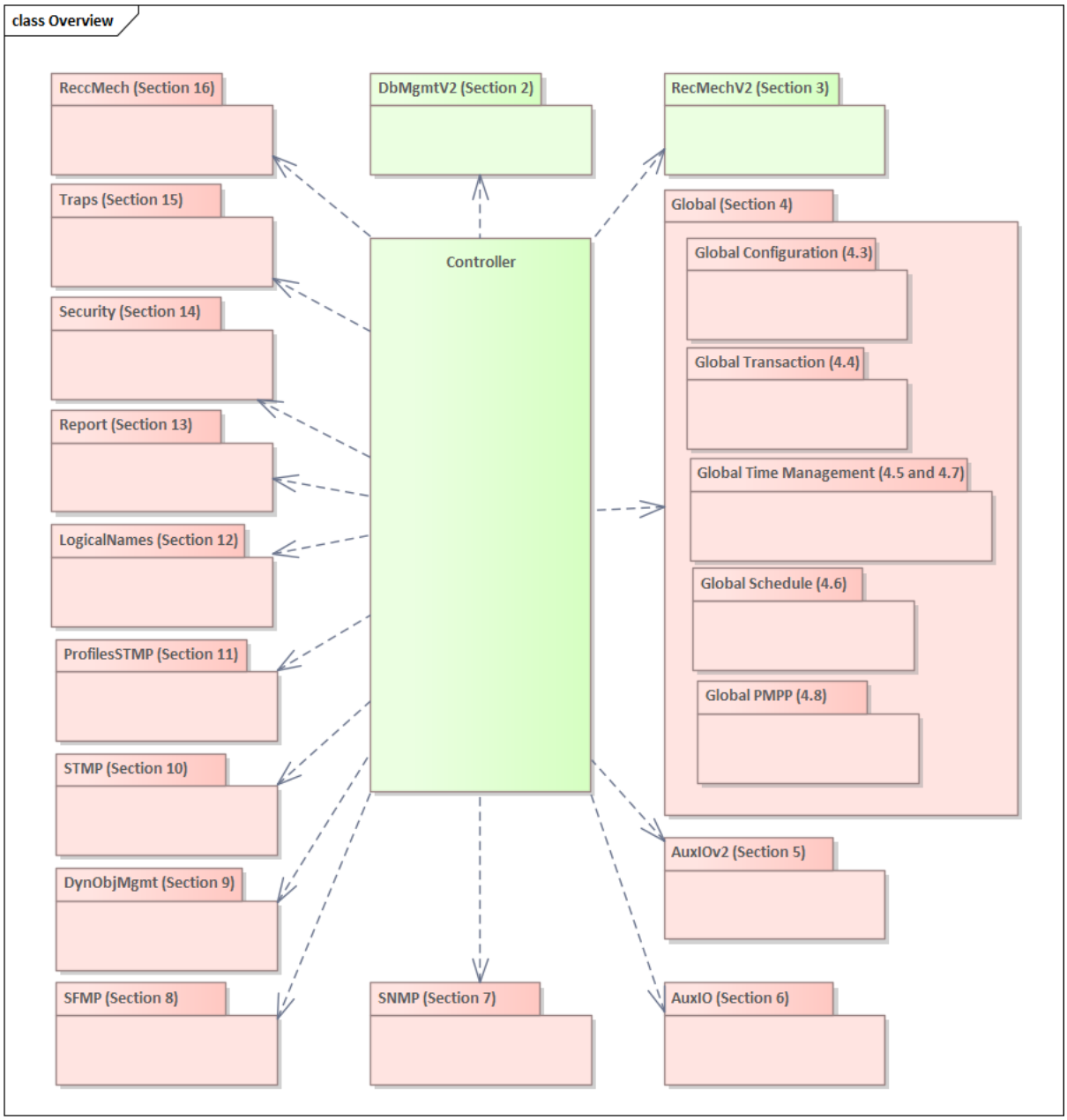

Figure 23 Controller Class Diagram

Figure 9 indicates that a Controller can include the following major components, which correspond to the sections of this document:

1) Database management version 2
2) Recording mechanism version 2
3) Global features, which include:
   a) Global configuration
   b) Global transaction
   c) Global time management
   d) Global schedule
   e) Global PMPP data
4) auxiliary input/output version 2
5) auxiliary input/output (version 1)
6) SNMP data
7) SFMP data
8) dynamic object management
9) STMP data
10) STMP profile data
11) Logical names
12) Event reporting
13) Security
14) Traps
15) Recording mechanism (version 1)

The first two of these major components are current while the others have been deprecated.

NOTE 1—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.

NOTE 2—Two different versions of the database management feature are defined. The first, which is deprecated, was originally defined for SNMPv1 and was defined in NTCIP 1201 v01 and later updated in NTCIP 1201 v02. The second was originally defined in NTCIP 1201 v04 to reflect characteristics of SNMPv3.

NOTE 3—Two different versions of the recording mechanism are defined. the first, which is deprecated, was defined in NTCIP 1103 v03 and the second was originally defined in NTCIP 1201 v04 to provide for a consistent linking to the trigger mechanisms defined in ISO 20684-3.

More detailed class diagrams for each feature are provided in Figure 24 through Figure 42.

B.2.2 Database Management Version 2

Figure 24 depicts the database management version 2 data stored by a controller.

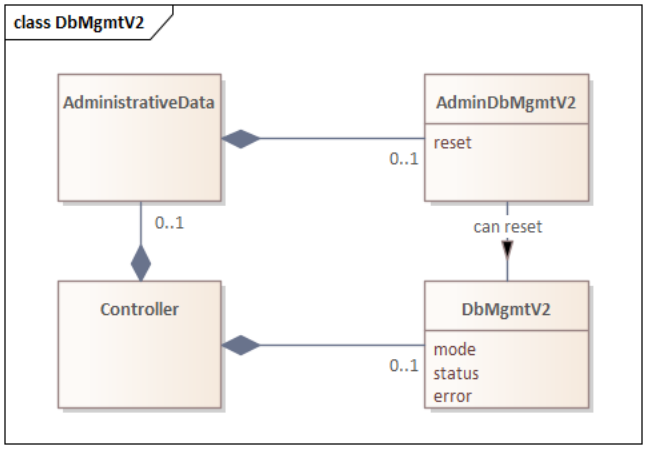

Figure 24 Class Diagram of the Database Management Version 2 Data

A controller can support the database management version 2 feature, which allows a manager to control the mode of database management while also monitoring its status and errors. In addition, a user with administrator rights can reset the database management feature. The mapping of each attribute in this class to object types are defined in Table 16.

Table 16: DbMgmtV2 Mapping to Class Diagram

Property	Object Type
DbMgmtV2.mode	dbMgmtV2Mode
DbMgmtV2.status	dbMgmtV2Status
DbMgmtV2.error	dbMgmtV2Error
AdminDbMgmtV2.reset	adminDbMgmtV2Reset

B.2.3 Recording Mechanism Version 2

Figure 25 depicts the recording mechanism version 2 data stored by a controller.

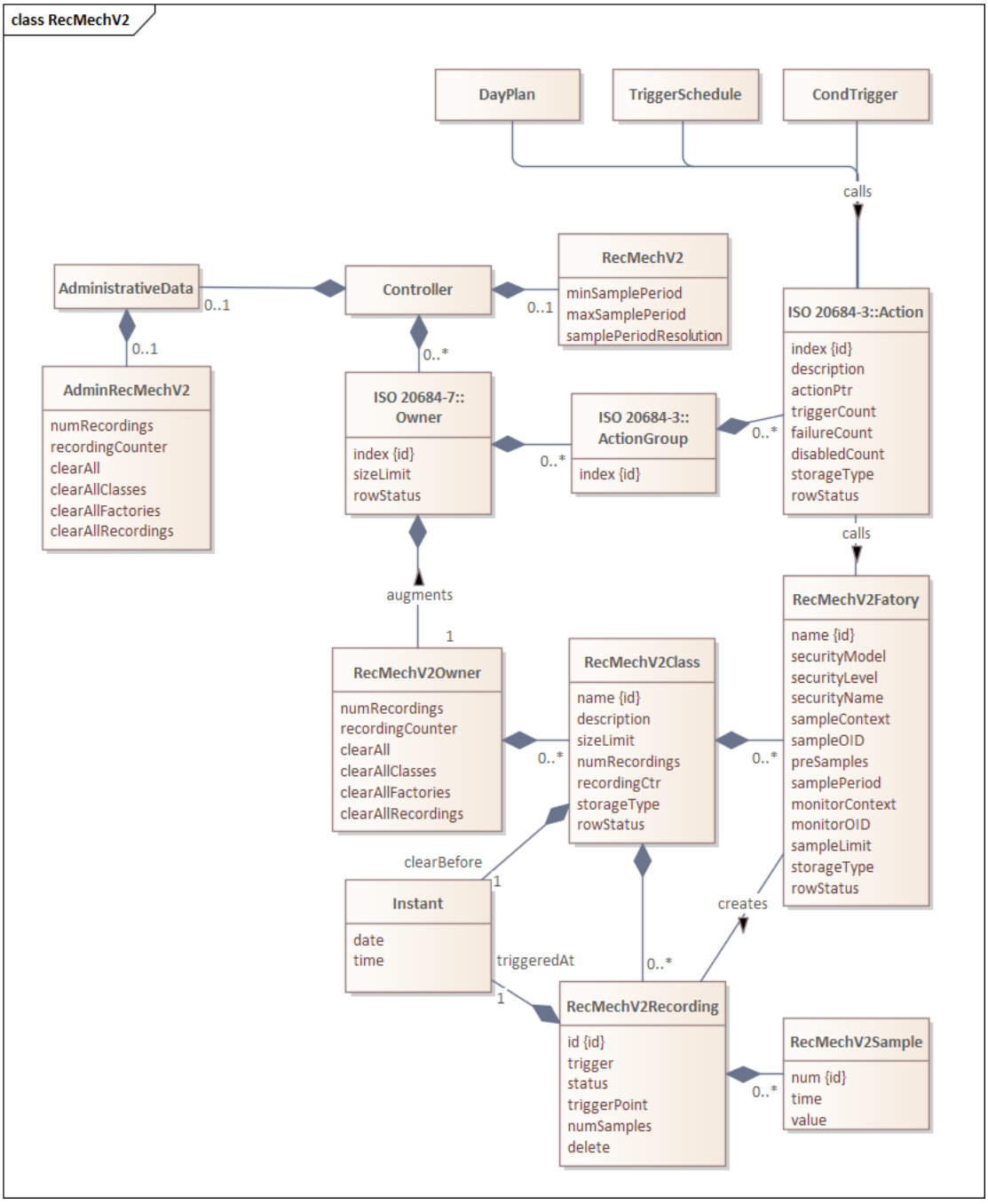

Figure 25 Class Diagram of the Recording Mechanism Version 2 Data

A controller can support the recording mechanism version 2 feature, which allows a manager to capture a series of data snapshots. The feature relies upon and extends the functionality of the Owner and Action features defined in ISO 20684-3.

A user with sufficient rights (e.g., an administrator) can monitor system-wide information, clear all information at different levels, and add new owners (i.e., as indicated with the AdminRecMechV2 and Owner classes). An Owner can then create their own action groups and actions (which are called by separately defined triggers, including those defined by the DayPlan, TriggerSchedule, and CondTrigger features of ISO 20684-3). An Action can be configured to call the RecMechV2Factory through its actionPtr. When called, the RecMechV2Factory creates a new RecMechV2Recording using the security

credentials that were used to configure the RecMechV2Factory. The recording consists of multiple (typically sampleLimit) samples of the sampleOID object, which are stored in instances of the RecMechV2Sample.

The mapping of each attribute in this class to object types are defined in Table 17.

Table 17: Recording Mechanism Version 2 Mapping to Class Diagram

Property	Object Type
AdminRecMechV2.numRecordings	adminRecMechV2NumRecordings
AdminRecMechV2.recordingCounter	adminRecMechV2RecordingCtr
AdminRecMechV2.clearAll	adminRecMechV2ClearAll
AdminRecMechV2.clearAllClasses	adminRecMechV2ClearAllClasses
AdminRecMechV2.clearAllFactories	adminRecMechV2ClearAllFactories
AdminRecMechV2.clearAllRecordings	adminRecMechV2ClearAllRecordings
RecMechV2Owner.numRecordings	recMechV2OwnerNumRecordings
RecMechV2Owner.recordingCounter	recMechV2OwnerRecordingCtr
RecMechV2Owner.clearAll	recMechV2OwnerClearAll
RecMechV2Owner.clearAllClasses	recMechV2OwnerClearAllClasses
RecMechV2Owner.clearAllFactories	recMechV2OwnerClearAllFactories
RecMechV2Owner.clearAllRecordings	recMechV2OwnerClearAllRecordings
RecMechV2Class.name	recMechV2ClassName
RecMechV2Class.description	recMechV2ClassDescription
RecMechV2Class.sizeLimit	recMechV2ClassSizeLimit
RecMechV2Class.clearBefore.date	recMechV2ClassClearDate
RecMechV2Class.clearBefore.time	recMechV2ClassClearTime
RecMechV2Class.numRecordings	recMechV2ClassNumRecordings
RecMechV2Class.recordingCtr	recMechV2ClassRecordingCtr
RecMechV2Class.storageType	recMechV2ClassStorageType
RecMechV2Class.rowStatus	recMechV2ClassRowStatus
RecMechV2Factory.name	recMechV2FactoryName
RecMechV2Factory.securityModel	recMechV2FactorySecurityModel
RecMechV2Factory.securityLevel	recMechV2FactorySecurityLevel
RecMechV2Factory.securityName	recMechV2FactorySecurityName
RecMechV2Factory.sampleContext	recMechV2FactorySampleContext
RecMechV2Factory.sampleOID	recMechV2FactorySampleOID
RecMechV2Factory.preSamples	recMechV2FactoryPreSamples
RecMechV2Factory.samplePeriod	recMechV2FactorySamplePeriod
RecMechV2Factory.monitorContext	recMechV2FactoryMonitorContext
RecMechV2Factory.monitorOID	recMechV2FactoryMonitorOID
RecMechV2Factory.sampleLimit	recMechV2FactorySampleLimit
RecMechV2Factory.storageType	recMechV2FactoryStorageType
RecMechV2Factory.rowStatus	recMechV2FactoryRowStatus
RecMechV2Recording.id	recMechV2RecordingID
RecMechV2Recording.trigger	recMechV2RecordingTrigger
RecMechV2Recording.triggeredAt.date	recMechV2RecordingTriggerDate
RecMechV2Recording.triggeredAt.time	recMechV2RecordingTriggerTime
RecMechV2Recording.status	recMechV2RecordingStatus
RecMechV2Recording.triggerPoint	recMechV2RecordingTriggerPoint
RecMechV2Recording.numSamples	recMechV2RecordingNumSamples
RecMechV2Recording.delete	recMechV2RecordingDelete
RecMechV2Sample.num	recMechV2SampleNum
RecMechV2Sample.time	recMechV2SampleTime
RecMechV2Sample.value	recMechV2SampleValue

Property	Object Type
RecMechV2.minSamplePeriod	recMechV2MinSamplePeriod
RecMechV2.maxSamplePeriod	recMechV2MaxSamplePeriod
RecMechV2.samplePeriodResolution	recMechV2SamplePeriodResolution
Owner.name	ISO 20684-7 fdOwnerName
Owner.sizeLimit	ISO 20684-7 fdOwnerSizeLimit
Owner.rowSatus	ISO 20684-7 fdOwnerRowStatus
ActionGroup.name	ISO 20684-3 fdActionName
Action.index	ISO 20684-3 fdActionIndex
Action.description	ISO 20684-3 fdActionDescription
Action.actionPtr	ISO 20684-3 fdActionPointer
Action.triggerCount	ISO 20684-3 fdActionTriggerCount
Action.failureCount	ISO 20684-3 fdActionFailureCount
Action.disabledCount	ISO 20684-3 fdActionDisabledCount
Action.storageType	ISO 20684-3 fdActionStorageType
Action.rowStatus	ISO 20684-3 fdActionRowStatus

B.2.4 Configuration Information

Figure 26 depicts the configuration data stored by a controller.

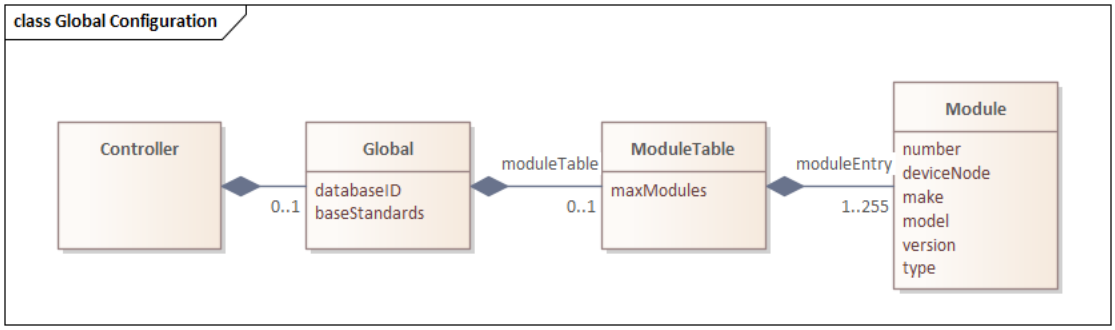

Figure 26 Class Diagram of the Configuration Information

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:

a) The module number;
b) The device node to which the module relates;
c) The make of the module;
d) The model of the module;
e) The version of the module; and
f) The type of module.

The mapping of each attribute in this class to object types are defined in Table 18.

Table 18: Configuration Information Mapping to Class Diagram

Property	Object Type
Global.databaseID	globalSetIDParameter
Global.baseStandards	controllerBaseStandards
ModuleTable.maxModules	globalMaxModules

Property	Object Type
Module.number	moduleNumber
Module.deviceNode	moduleDeviceNode
Module.make	moduleMake
Module.model	moduleModel
Module.version	modelVersion
Module.type	moduleType

B.2.5 Transaction Information

Figure 27 depicts the transaction state data stored by a controller.

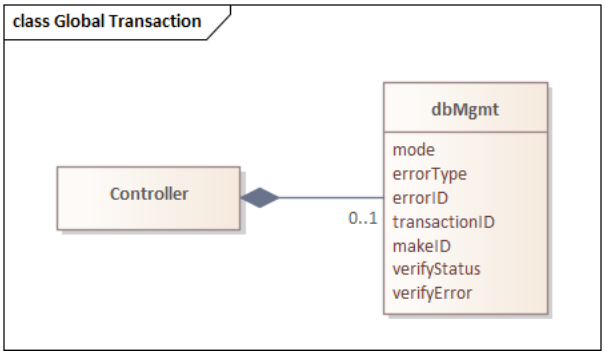

Figure 27 Class Diagram of the Transaction Service

A controller may support a transaction feature (see Figure 27). The following information characterizes the feature as defined in NTCIP 1103 v03:

a) A mode;
b) A verify status; and
c) 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 19.

Table 19: 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

B.2.6 Time and Daylight Saving Time (DST) Information

Figure 28 depicts the time-related and DST-related data stored by a controller.

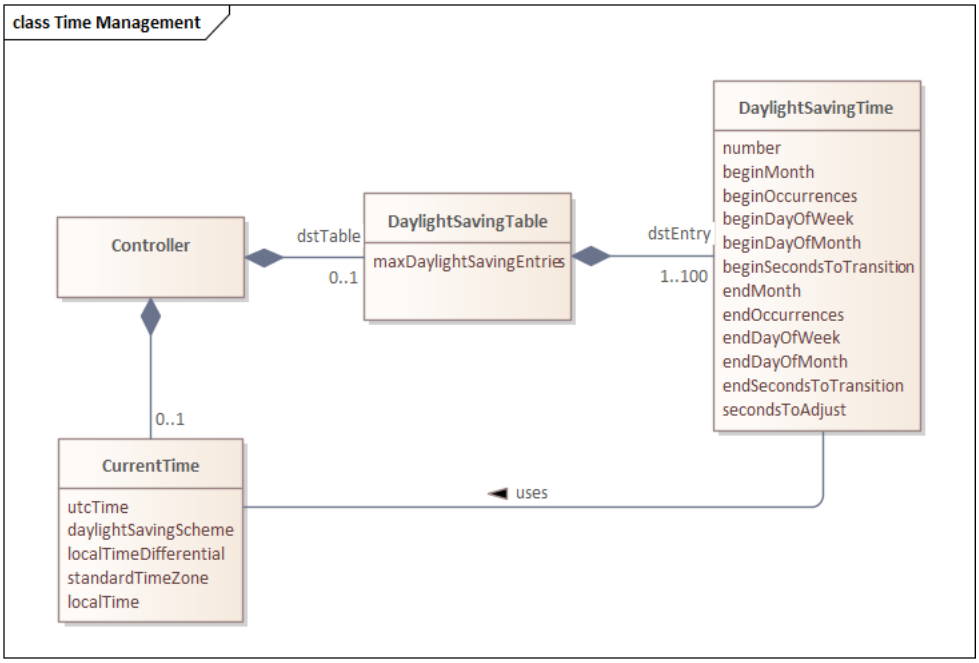

Figure 28 Class Diagram of Time/DST Information

A controller may store time-related information, including:

a) The current time in UTC
b) An indication of the daylight saving mode
c) An indication of the time zone when in standard time
d) 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:

a) A DST number
b) A begin DST month indicating in which month the DST may begin
c) 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—"beginOccurances" (sic) is misspelled in the figure, but is spelled correctly in the MIB object definitions.]
d) A begin DST day of week indicating on which day of week the DST may begin
e) A begin DST day of month indicating on which day of month the DST may begin
f) A begin DST Seconds to Transition parameter indicating after how many seconds after midnight of a particular day the DST may begin
g) An end DST month indicating in which month the DST may begin
h) 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
i) An end DST day of week indicating on which day of week the DST may end
j) An end DST day of month indicating on which day of month the DST may end
k) An end DST Seconds to Transition parameter indicating after how many seconds after midnight of a particular day the DST may end
l) 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 20.

Table 20: 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

B.2.7 Generic Schedule Information

Figure 29 depicts the generic schedule-related data stored by a controller.

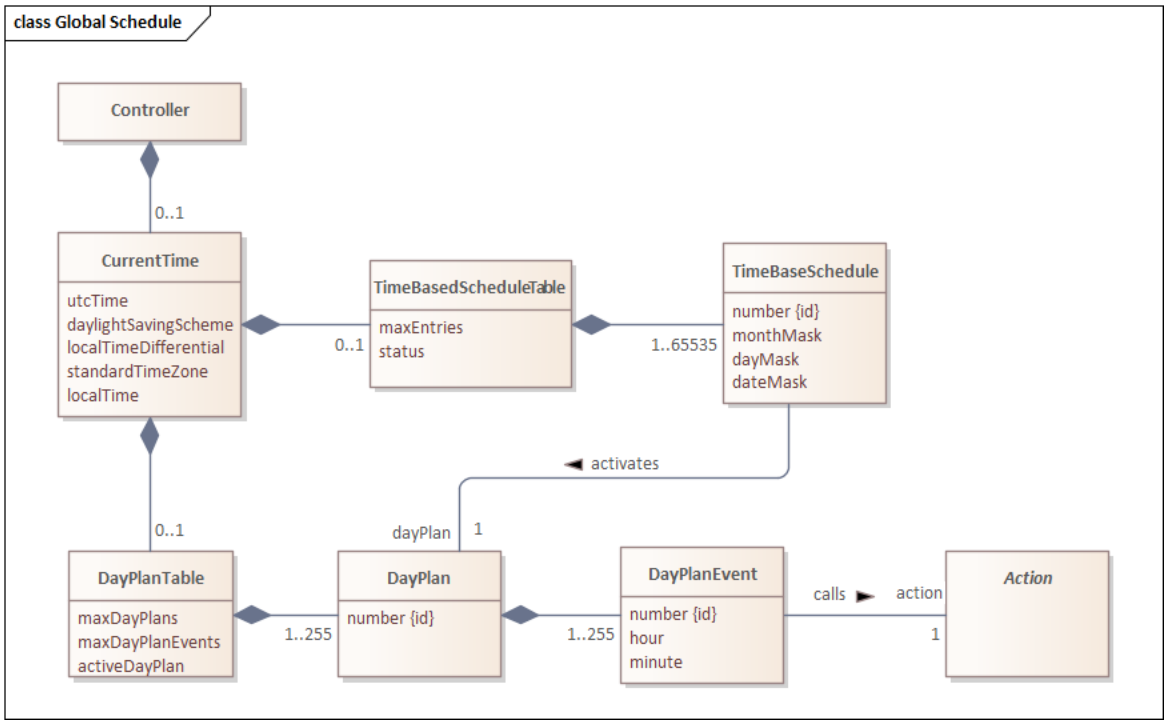

Figure 29 Class Diagram of Generic Schedule-Related Information

Figure 29 indicates a controller may store schedule information, including:

a) The current time in UTC
b) An indication of the time zone when in standard time
c) 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:

a) A schedule number
b) A month mask indicating which months the schedule may be active
c) A day mask indicating which days of the week the schedule may be valid
d) A date mask indicating which dates of the month the schedule may be active
e) 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:

a) The maximum number of day plans that may be stored, which must be between one and 255
b) The maximum number of events that may occur during a day, which must be between one and 255
c) 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:

a) A number;
b) The hour during which the event occurs;
c) The minute during which the event occurs;
d) The status of the action; and
e) 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 21.

Table 21: Generic Schedule 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

B.2.8 PMPP Information

Figure 30 depicts the database management version 2 data stored by a controller.

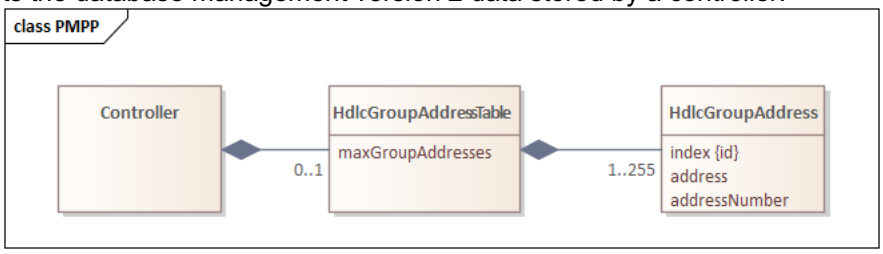

Figure 30 Class Diagram of the PMPP Data

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 22.

Table 22: PMPP Mapping to Class Diagram

Property	Object Type
HdlcGroupAddressTable.maxGroupAddresses	maxGroupAddresses
HdlcGroupAddress.index	hdlcGroupAddressIndex
HdlcGroupAddress.address	hdlcGroupAddress
HdlcGroupAddress.addressNumber	hdlcGroupAddressNumber

B.2.9 Auxiliary Input/Output Information

Figure 31 and Figure 32 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.

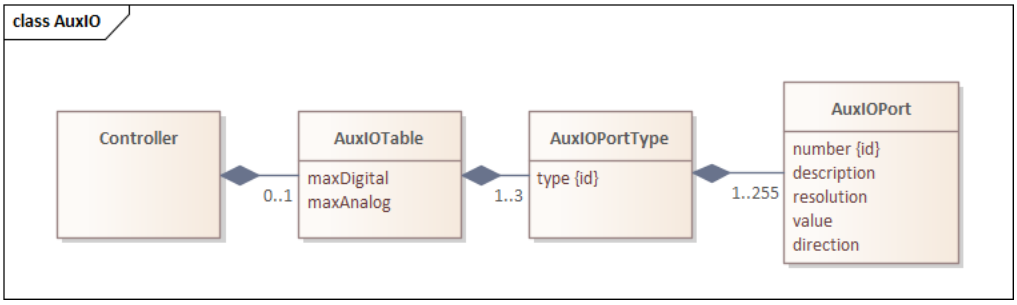

Figure 31 Class Diagram for Auxiliary Input/Output (NTCIP 1203 v01)

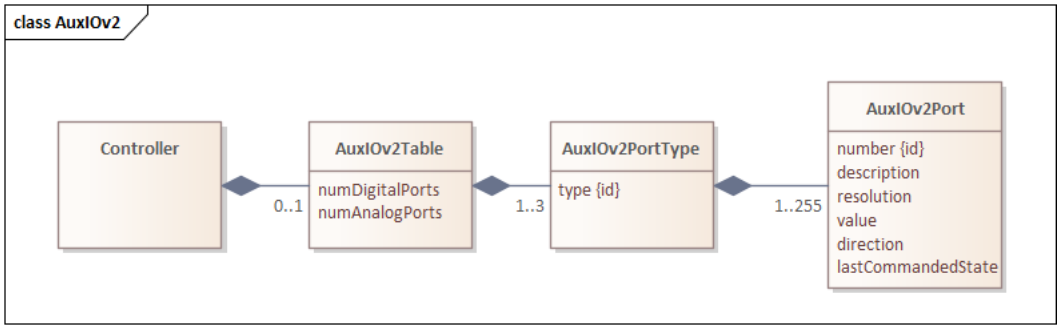

Figure 32 Class Diagram for Auxiliary Input/Output Version 2 (NTCIP 1201 v02)

Figure 31 and Figure 32 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:

a) A number
b) A description
c) A resolution of the data supported by the port
d) A value
e) A direction
f) The last commanded state (only in NTCIP 1201 v02)

The mapping of each attribute in this class to object types are defined in Table 23.

Table 23: 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

B.2.10 SNMP Information

Figure 33 depicts the NTCIP-defined SNMP data stored by a controller.

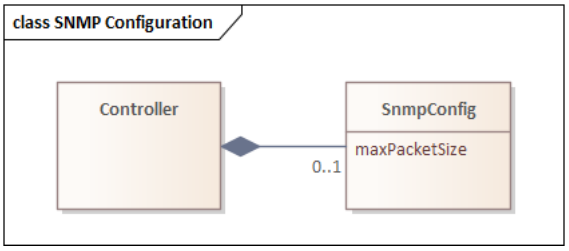

Figure 33 Class Diagram of the NTCIP-Defined SNMP Data

A controller can indicate the maximum SNMP packet size. The mapping of each attribute in this class to object types are defined in Table 24.

Table 24: NTCIP-Defined SNMP Data Mapping to Class Diagram

Property	Object Type
SnmpConfig.maxPacketSize	snmpMaxPacketSize

B.2.11 SFMP Information

Figure 34 depicts the SFMP data stored by a controller.

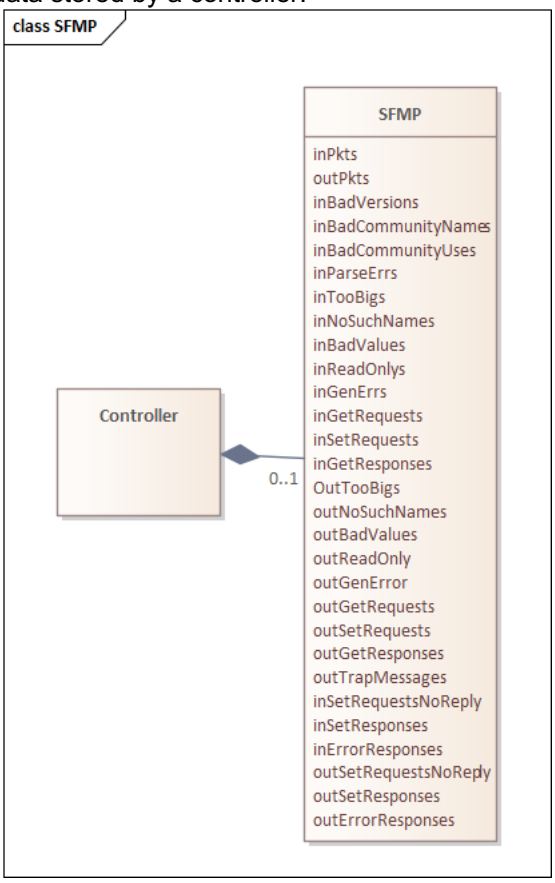

Figure 34 Class Diagram of the SFMP Data

A controller can indicate various statistics for SFMP. The mapping of each attribute in this class to object types are defined in Table 25.

Table 25: 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.inParsErrs	sfmpInParsErrs
SFMP.inTooBigs	sfmpInTooBigs
SFMP.inNoSuchNames	sfmpInNoSuchNames
SFMP.inBadValues	sfmpInBadValues
SFMP.inReadOnlys	sfmpInReadOnlys
SFMP.inGenErrs	sfmpInGenErrs
SFMP.inGetRequests	sfmpInGetRequests
SFMP.inSetRequests	sfmpInSetRequests
SFMP.inGetResponses	sfmpInGetResponses
SFMP.outTooBigs	sfmpOutTooBigs
SFMP.outNoSuchNames	sfmpOutNoSuchNames
SFMP.outBadValues	sfmpOutBadValues
SFMP.outReadOnlys	sfmpOutReadOnlys
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

B.2.12 Dynamic Object Management Information

Figure 35 depicts the data stored by a controller to manage dynamic objects.

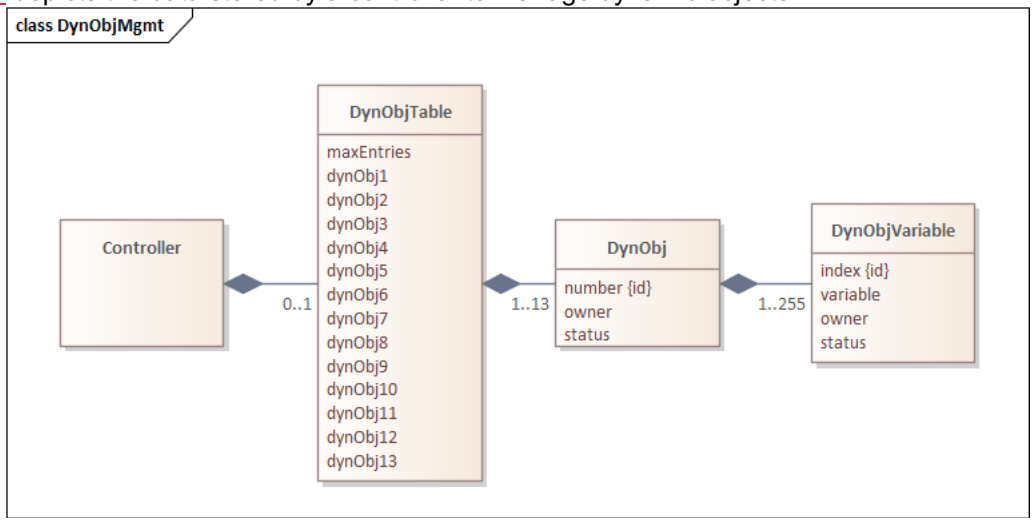

Figure 35 Class Diagram for Dynamic Object Management

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 26.

Table 26: 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

B.2.13 STMP Information

Figure 36 depicts the STMP data stored by a controller.

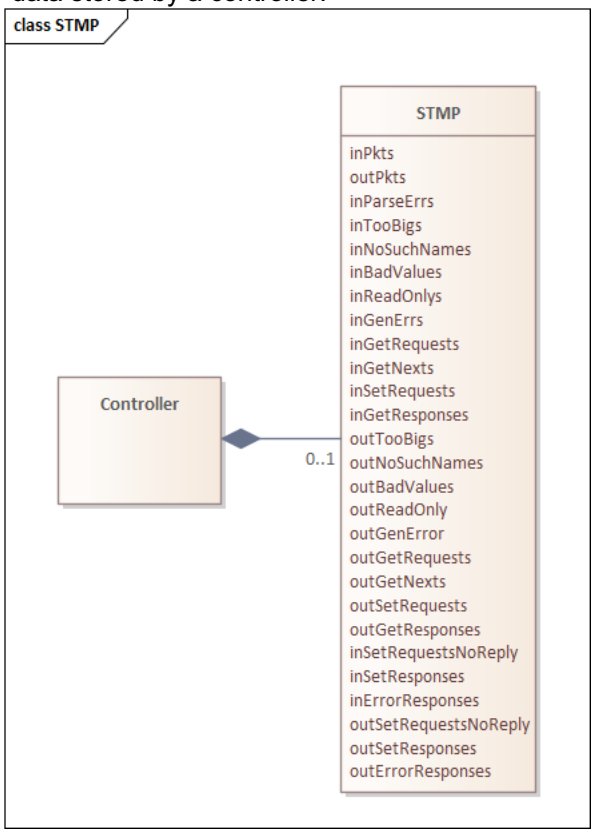

Figure 36 Class Diagram of STMP Data

A controller can support various STMP statistics. The mapping of each attribute in this class to object types are defined in Table 27.

Table 27: 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

B.2.14 Profiles STMP

Figure 37 depicts the STMP profile data stored by a controller.

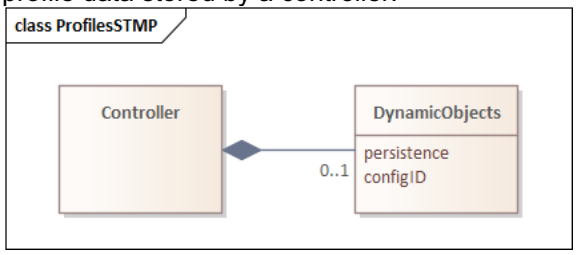

Figure 37 Class Diagram of the STMP Profile Data

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 28.

Table 28: STMP Profile Data Mapping to Class Diagram

Property	Object Type
DynamicObjects.persistence	dynamicObjectPersistence
DynamicObjects.configID	dynamicObjectTableConfigID

B.2.15 Logical Names Information

Figure 38 depicts the logical names data stored by a controller.

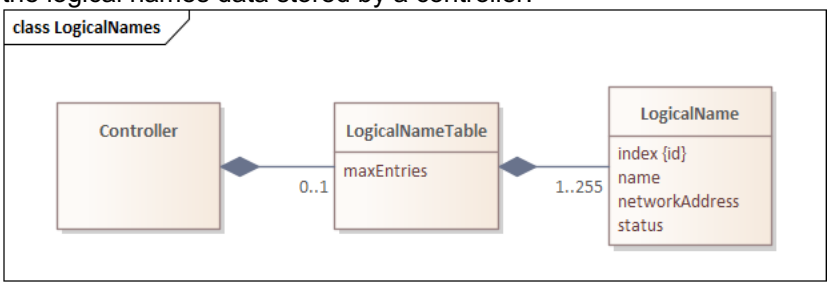

Figure 38 Class Diagram of the Logical Names Data

A controller can support a logical names table. The mapping of each attribute in this class to object types are defined in Table 29.

Table 29: 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

B.2.16 Event Report Information

Figure 39 depicts the event report data stored by a controller.

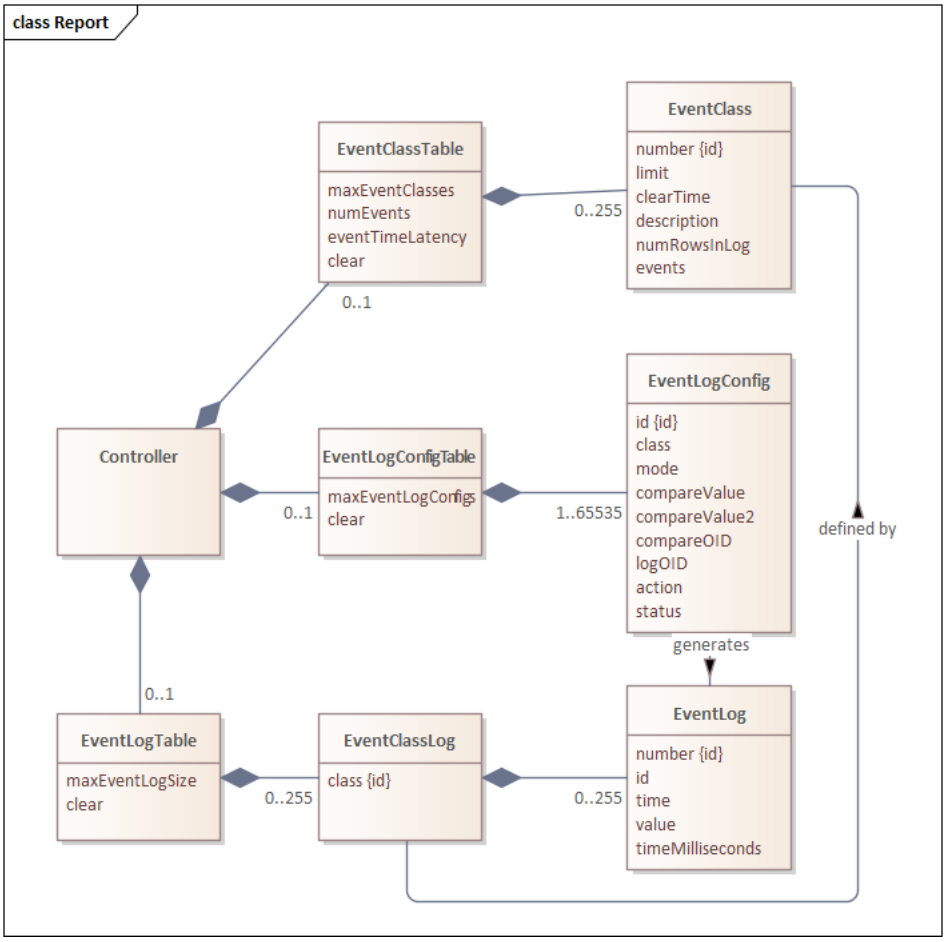

Figure 39 Class Diagram of the Event Report Data

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 30.

Table 30: 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

B.2.17 Community Name Security Information

Figure 40 depicts the community name management data stored by a controller.

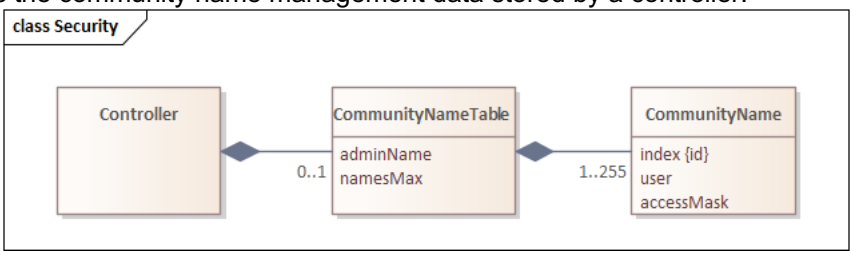

Figure 40 Class Diagram of the Community Name Security Data

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 31.

Table 31: 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

B.2.18 Trap Management Information

Figure 41 depicts the trap management data stored by a controller.

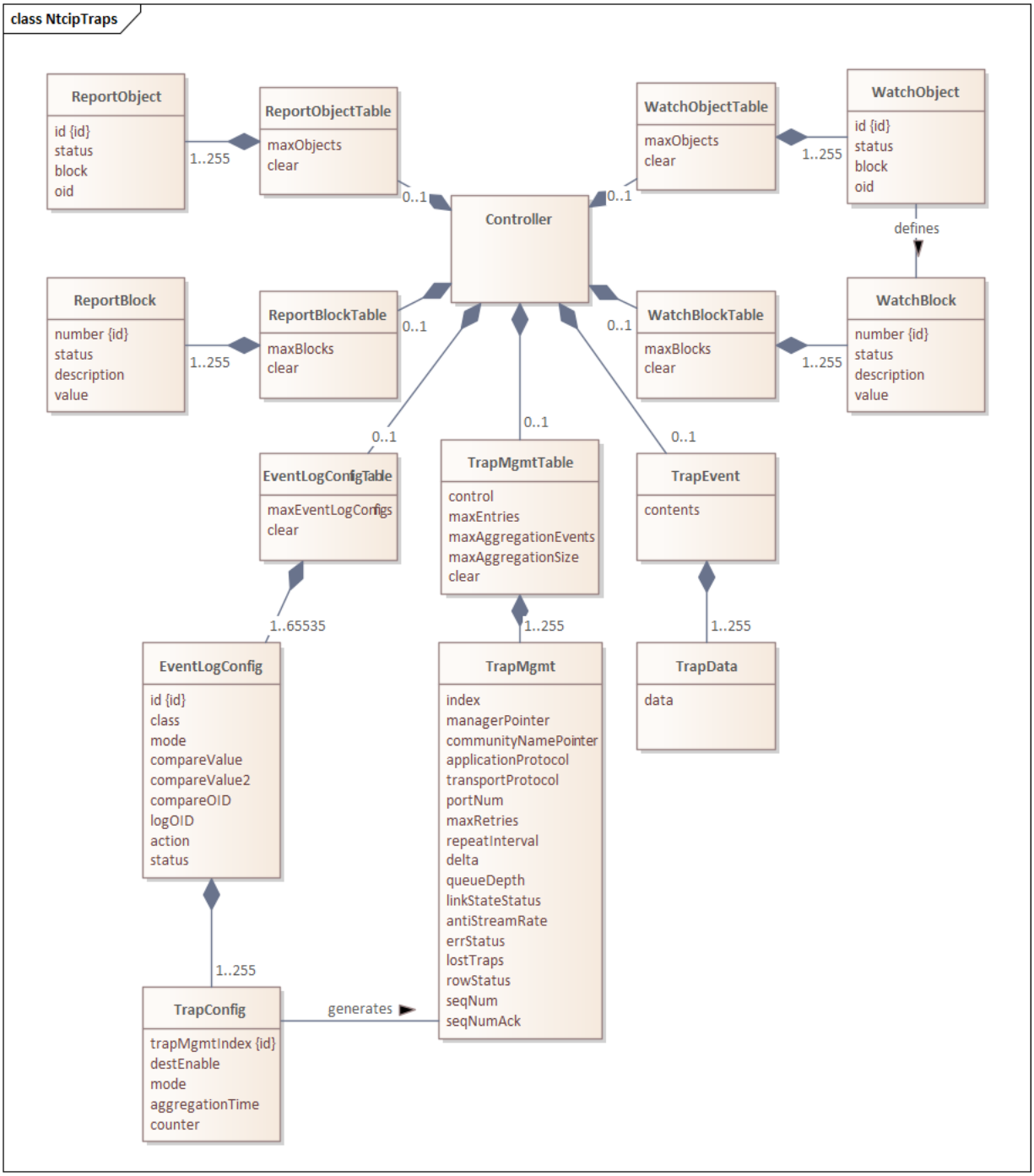

Figure 41 Class Diagram of the Trap Management Data

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 32.

Table 32: Trap Management Data Mapping to Class Diagram

Property	Object Type
WatchObjectTable.maxObjects	maxWatchObjects
WatchObjectTable.clear	clearWatchObjects

Property	Object Type
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
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

Property	Object Type
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

B.2.19 Recording Mechanism (Version 1) Information

Figure 42 depicts the data stored by a controller for the recording mechanism version 1 feature as used prior to NTCIP 1201 v04.

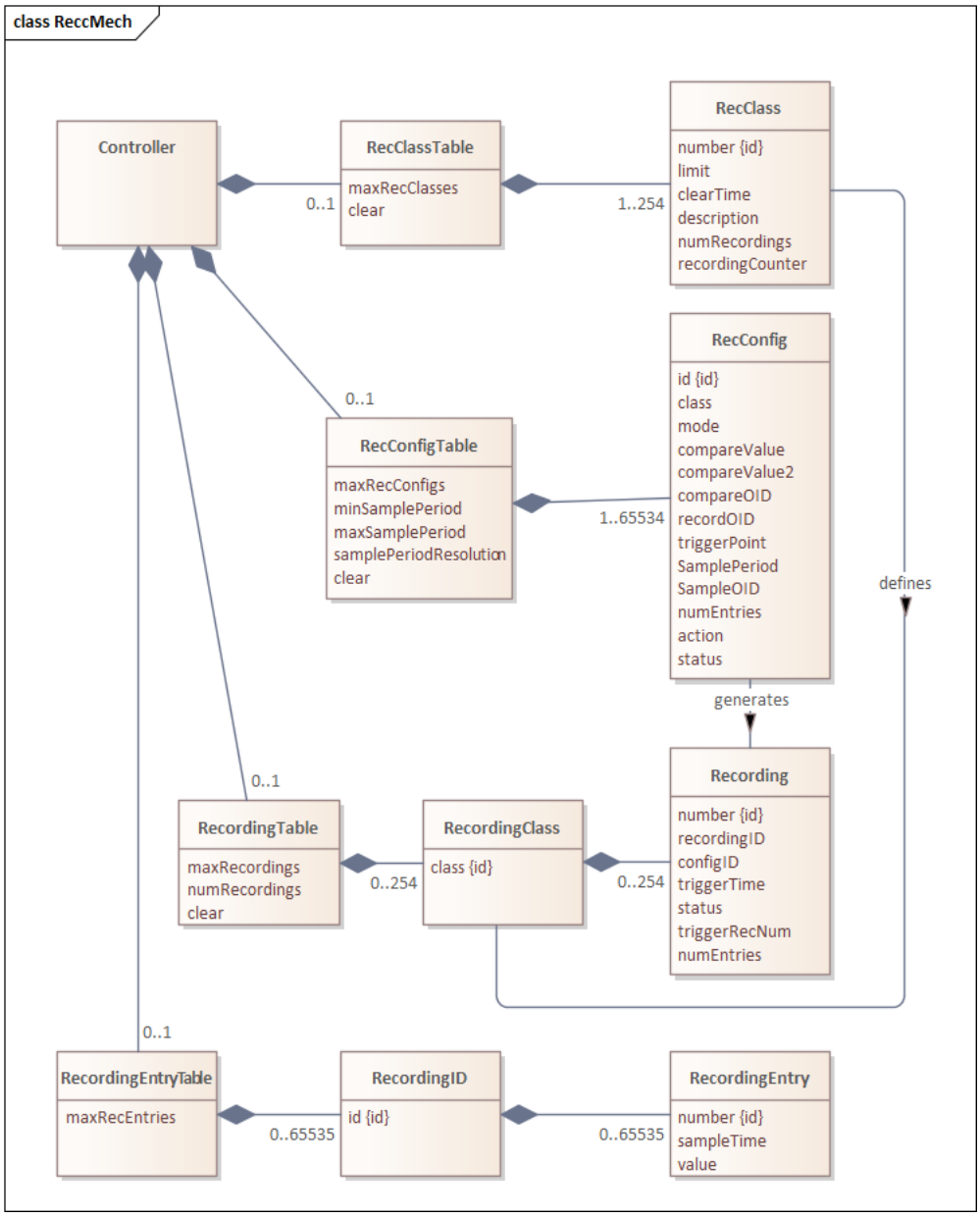

Figure 42 Class Diagram of the Recording Mechanism (Version 1) Data

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 33.

Table 33: 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 C
TEST PROCEDURES
[INFORMATIVE]

NOTE—It is anticipated that a future version of NTCIP 1201 may include test procedures, and Annex C serves as a placeholder. If test procedures are added, they will be normative.

Annex D
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 D explains the issue identified resulting in significant revisions, provides a description of the revision made, and a brief analysis of the impact of each revision on existing implementations.

NOTE—NTCIP 1201:2005 is referenced as NTCIP 1201 v02. NTCIP 1201:1996 (also referred to as NTCIP 1201:1997) is referenced as NTCIP 1201 v01.

D.1 REVISIONS FROM NTCIP 1201 V01 TO NTCIP 1201 V02

D.1.1 Updated Object Tree

Because of the various other revisions in NTCIP 1201 v02, the (ISO) Object Tree was updated to properly reflect content. Interoperability issues are not anticipated.

D.1.2 Updated to Conform with NTCIP 8004 v02

Data stored in field devices are often retrieved by a central system and then may be exchanged with other centers as a part of regional communications. These center-to-center communications use protocols other than SNMP and require the data to be defined according to either IEEE 1489 (or its recently approved update known as ISO 14817). NTCIP v01 standards did not define data in this format, creating ambiguities for center-to-center implementations.

To ensure that a single definition for all NTCIP data, regardless of its use context (e.g., center-to-center vs. center-to-field), NTCIP defined an enhanced MIB format, as defined in NTCIP 8004 v02, to be used for all new and updated NTCIP standards.

Interoperability issues are not anticipated because of the additions that this update creates (e.g., the <DEFINITION> tags, etc.).

D.1.3 Updated Name of the MIB

Changes to a MIB can affect the way other MIBs import data. Thus, when a MIB imports data from another MIB, it should be able to unambiguously reference the specific version of the MIB that it wants to import. Therefore, every update to an NTCIP standard results in an update to the name of the MIB according to the rules in NTCIP 8004 v02.

Additionally, because there are now two versions of the auxiliary input/output object definitions, two additional MIBs were created, each of which contains one of the AuxIO definitions.

The update to the MIB name should prevent ambiguity as to which version of a MIB may be referenced from another MIB.

D.1.4 Added Default Value Statements

Interoperability problems can arise when different controllers initialize differently. As a result, this standard has standardized the default initialization value of several configuration and control parameters.

This NTCIP 1201 v02 revision may result in some NTCIP 1201 v01 devices performing slightly differently than NTCIP 1201 v02 devices. However, this should *reduce* interoperability issues overall. Current implementations operate differently from one another, and any central system is customized to handle

this uniqueness for each manufacturer. By defining the default value, this customization can be avoided in the future.

D.1.5 [Section Deleted]

NOTE—Annex D.5, renumbered as Annex D.1.5 in NTCIP 1201 v03, was not included in NTCIP 1201 v02.

D.1.6 Enhanced Module Version Definition

The module table provides basic information about the make, model, and version of the controller. However, NTCIP 1201 v01 provided a generic format for the version that did not adequately allow for proper configuration management of software. NTCIP 1201 v02 defined a detailed format for the presentation of version information.

While some NTCIP 1201 v01 devices may not conform, this revision should not present any real interoperability issues between NTCIP 1201 v01 and NTCIP 1201 v02 devices.

D.1.7 Added an Object to Identify Supported Standards

Several integrators have expressed concerns over the ability to be able to quickly determine to which NTCIP standards and which versions of NTCIP standards a device claims conformance. By being able to query the device to determine which standards it supports, a central system can quickly determine how to manage the device. Therefore, an object providing this information in a standard format was added.

Interoperability issues are not anticipated. A central system can readily identify any NTCIP v01 device, since the device returns a noSuchName error.

D.1.8 Corrected the Database Transaction Feature

The transaction mode process was modified by NTCIP 1201:1996 Amendment 1. Implementations discovered that the original process did not provide for the desired operation in the presence of multiple management stations (e.g., a central and a local laptop). Specifically, there were issues with the second management station overriding the first operation to issue a control command. The solution deprecated dbErrorID, dbTransactionID, and dbMakeID; revised the definition of dbCreateTransaction; and created two new objects labeled dbVerifyStatus and dbVerifyError.

This revision resolves pre-existing interoperability issues. While NTCIP 1201 v01 implementations require modification to conform to NTCIP 1201 v02, the NTCIP 1201 v01 feature did not work as intended.

NOTE—For the dbVerifyStatus object, to align NTCIP 1201 v02 with other NTCIP and several Internet standards, the object definitions in NTCIP 1201 v01 that had enumerated values starting with a value of (0) have been changed in NTCIP 1201 v02 to start with a value of (1). The changed definition in NTCIP 1201 v02 makes this object incompatible with NTCIP 1201 v01.

D.2 REVISIONS FROM NTCIP 1201 V02 TO NTCIP 1201 V03

D.2.1 Added Support for Additional Daylight Saving Modes

The *globalDaylightSaving* object was deprecated after NTCIP 1201 v02; therefore, the following paragraphs are no longer relevant. See Annex D.2.2 for more information.

Several parties located outside of the U.S. are now deploying NTCIP for various devices and have pointed out that the NTCIP should support all of the various daylight saving plans. Thus, these have been added to the daylight saving object.

This addition is fully backwards compatible and should not cause any interoperability problems. It will have no effect on systems in the US; NTCIP 1201 v01 systems outside of the US have not had a way to offer support of other daylight saving modes in a standard way, but with the NTCIP 1201 v02 enhancement, this feature is offered.

D.2.2 Added New Objects to Address US Daylight Saving Time (DST) Modifications

The globalDaylightSaving object was deprecated and a table allowing definition of the beginning and ending of DST was added (see dstBeginMonth). The definition of these parameters in a table allows definition of one or more sets of DST. Additionally, a new enumerated value was added to globalDaylightSaving object to support cases where a device supports both mechanisms and backward compatibility is required.

These changes were due to new rules enacted by the U. S. Congress to take effect in 2007. Since the DST rules have changed about every 10 years, the new set of DST objects allows users to define the start and end of DST.

D.2.3 Added a Schedule Status Object

Some agencies wanted to monitor the logic of the timebase schedule more closely and, as a result, a status object is added to the timebase schedule table. This extra feature is anticipated to be backward compatible.

D.2.4 Clarified Definitions of Day Plan Objects

Various questions were raised about the precise meaning of the object definitions for the day plan table. NTCIP 1201 v03 clarifies these definitions. However, the clarifications reflect actual implementations and interoperability issues are not anticipated.

D.2.5 Corrected Problems with the Local Time Logic

A problem was discovered with the time differential logic in that if the globalTime was set during the one-hour fall-back period of the DST logic; there was an ambiguity as to what time was intended. Manufacturers overcame this ambiguity via unique implementations, many of which created interoperability issues with other manufacturers. Several options were considered to correct this flawed logic, but all resulted in some level of interoperability issues. The best solution was to produce the cleanest design, which required deprecating the global time differential object (globalLocalTimeDifferential) and adding new objects for local time (controllerLocalTime) and time zone (controllerStandardTimeZone).

NTCIP 1201 v02 corrects an existing interoperability problem. Minor compatibility issues between the variety of NTCIP 1201 v01 interpretations and NTCIP 1201 v02 were noted, but an alternative solution that adequately corrected these issues without presenting new issues was not identified. By deprecating objects and creating new objects, any central system can quickly discover (by receiving a noSuchName error) when the wrong version is used.

D.2.6 Clarified Definitions Related to the Event Log

The exact definitions used for objects in the event log yielded a variety of detailed comments. Several clarifications resulted, but in all cases, these merely clarified the text and explained how manufacturers had implemented the features. Interoperability issues are not anticipated.

D.2.7 Reordered Sections for the Event Log

The order of the sections related to the event log proved confusing to some users, and sections were re-ordered. However, the OBJECT IDENTIFIERs for the objects have not changed. As such, this is an editorial change; interoperability issues are not anticipated.

After the above changes were made, the report node and event related objects were moved to NTCIP 1103 v02.

D.2.8 Added Support for Another Mode to Event Log and Moved to NTCIP 1103 v02

Based on requests from implementers, a new mode was added for the event log configuration table (andedWithValue) and better explanations of the definitions of each mode were provided.

This is an editorial clarification; interoperability issues are not anticipated.

NOTE—After the above changes were made, the report node and even-related objects were moved to NTCIP 1103 v02.

D.2.9 Added Error Value to the Event Configuration Status

Based on requests from implementers, an error code to the status object of the event configuration table was added to ensure that the controller is not programmed to repeatedly check an invalid condition. Logic was also added to the object that requires a consistency check whenever the configuration of the row changes.

Minor backward compatibility challenges may be anticipated; however, issues associated with some manufacturers' use of the 'other' code to mean error are avoided.

After the above revisions were made, the report node and event related objects were moved to NTCIP 1103 v02.

D.2.10 Corrected Syntax of Event Log Size Object

NTCIP 1201 v01 indicated that the lower bound of the event log size was zero; however, if the size was zero, there would be no table and this object would not be supported. Thus, to avoid this contradiction, the lower limit was re-defined to be one.

Interoperability issues are not anticipated, since any implementation supporting this feature is expected to have a value greater than one.

After the above revisions were made, the report node and event related objects were moved to NTCIP 1103 v02.

D.2.11 Replaced the Group Address Object

NTCIP 1201 v01 had an object defining the PMPP group address to which the device belonged; however, the meaning of the value had been interpreted in two different ways. One group held that the value was supposed to be the group address number that was encoded in the PMPP address field. The other group held that the value was the encoded PMPP address field. Because of this conflict and resulting non-interoperability in deployed systems, the existing object (hdlcGroupAddress) was deprecated, and a new object was defined (hdlcGroupAddressNumber) to resolve the issue.

While replacing the existing object presents a minor interoperability issue, the solution does provide an unambiguous definition of the object, and any central system can readily identify NTCIP 1201 v01 implementations since these do not support the NTCIP 1201 v02 object.

D.2.12 Added Generic Auxiliary I/O Objects

The development of the NTCIP DMS standard (NTCIP 1203 v01) identified the need to support auxiliary I/O ports. This need was later identified by other NTCIP device-specific standards development groups, including the ESS WG. As a result, the auxiliary objects defined in NTCIP 1203 were moved to NTCIP 1201 v02. An effort to just refine, enhance, and rename the object definitions was anticipated to lead to interoperability issues. Therefore, the original object definitions moved from NTCIP 1203 v01 to NTCIP 1201 v01 were retained (see Section 0) but deprecated, and it is anticipated that these will be removed from a successor of NTCIP 1201 v03. The renamed and redefined object definitions are now also defined (see Section 0).

The provision of both sets of Auxiliary I/O object definitions, minimizes interoperability issues with either NTCIP 1201 v01 or NTCIP v02 implementations.

D.2.13 Removed Conformance Statements

Deployments using the NTCIP v01 ('version 01') standards highlighted problems with using conformance groups in agency specifications. Issues also arose as NTCIP standards were updated to include systems engineering information (such as user needs and requirements), since user needs and requirements for each feature were typically not defined in a clear fashion within the NTCIP v01 standard.

Subsequent NTCIP standards follow an outline that is based on a systems engineering process (SEP). As a result, conformance groups were eliminated and replaced with a Protocol Requirements List (PRL). In the case of NTCIP 1201 v03, the PRL is located in the subject device standard that references NTCIP 1201 v03. The PRL, combined with the Requirements Traceability Matrix (RTM), now defines the conformance requirements rather than the conformance groups and conformance statement used in NTCIP v01 standards.

Interoperability issues are not anticipated.

D.2.14 Added a Concept of Operations

The intended operation of some features was unclear to NTCIP 1201 v01 implementers. To address this issue, Annex A is included to explain how various features were intended to operate.

Although this is normative text, these are intended to clarify the text that already existed in the version 01 standard and are therefore not expected to produce any interoperability problems.

D.2.15 Prepared Communication Objects Moved to NTCIP 1103 v02

The security objects and the event log objects moved to NTCIP 1103 v02, because they relate more to application layer issues than to the end-application.

D.2.16 Deleted Annex B to Document Deprecated Objects

NTCIP 1201 v02 contained an Annex B that documented all objects that were deprecated between NTCIP 1201 v01 and NTCIP 1201 v02. Annex B provided future developers with an understanding of objects that may exist in or are used by legacy equipment. However, Annex D now serves this purpose.

Including this information may assist in newer systems being able to communicate with NTCIP 1201 v01 devices.

D.2.17 Added Class Diagrams

Many users of NTCIP 1201 v01 found it difficult to understand the context of the various objects defined in the Management Information Base (MIB). While various object definitions provided the detailed definition of each object, it was difficult to readily understand, at a high-level, how the data worked together.

Annex B provides high-level graphical images that depict the various relationships among all data defined in NTCIP 1201 v03, including the rules on multiplicity (i.e., how many of one object might exist for a given instance of another object). While all of this information was (and still is) recorded within the textual definition of the objects, providing high-level graphical depictions of these relationships facilitate this understanding.

Including this information may assist in clarification for users, without (of itself) contributing to backward compatibility issues.

D.2.18 Added Generic SNMP Interface Definitions

NTCIP 1203 v02 contained the definitions for a generic Simple Network Management Protocol (SNMP) interface. This definition continues to appear in NTCIP 1201 v03 **Error! Reference source not found.** because these operations are generic to all field devices.

D.3 REVISIONS FROM NTCIP 1201 V03 TO NTCIP 1201 V04

D.3.1 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.

D.3.2 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.

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 due 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. Rather than showing all deprecated objects along with current objects in a single MIB, the NTCIP community decided to:

1. upgrade the existing NTCIP 1103 v03 and NTCIP 1201 v03 MIBs to SMIv2 and show all contained objects as deprecated; the "deprecated MIBs" can be used to provide an official SMIv2 interpretation of the prior design (e.g., for a device that supports the prior data but has been upgraded to support SNMPv3)
2. create new MIBs that capture the current recommended design

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 old and new 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 keep the NTCIP 1103 v03 and NTCIP 1201 v03 MIBs intact and create replacement objects as needed in the current design MIB.

D.3.3 Incorporation of Features from ISO and IETF

Many of the features contained in NTCIP 1103 v03 and NTCIP 1201 v03 have parallels within ISO standards and IETF RFCs that are already defined in SMIv2 format. In fact, some of the features have been incorporated into the core of the SNMPv3 RFCs. Rather than duplicating this functionality, NTCIP 1201 v04 recommends importing the objects defined by these external standards as appropriate.

D.3.4 Summary

The result of the three significant changes identified in D.3.1 through D.3.3 effort was to deprecate all NTCIP 1103 v03 and NTCIP 1201 v03 MIBs (as shown in Sections 4-15) and to create two new MIBs: a replacement for database transactions (in Section 2) and a replacement for the high-resolution data

recording mechanism (in Section 3). Each object definition contained in Sections 4 through 15 provides a recommendation for what object(s) should supersede the object or why no superseding object is identified.

§