TECHNICAL SPECIFICATION

ISO/TS 20684-7

First edition
2022-10

Intelligent transport systems — Roadside modules SNMP data interface —

Part 7: Support features

Systèmes de transport intelligents — Interface de données SNMP pour les modules en bord de route —

Partie 7: Service de support

Reference number
ISO/TS 20684-7:2022(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems*.

A list of all parts in the ISO 20684 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

0.1 Background

The need for standardized communication with ITS field devices is growing around the world. Several countries have adopted Simple Network Management Protocol (SNMP) based field device communication standards.

There is a growing view and empirical evidence that standardizing this activity will result in improved ITS performance, reduced cost, reduced deployment time, and improved maintainability. The ISO 20684 series extends ISO 15784-2 by defining the management information necessary to monitor, configure and control features of field devices. The data elements defined in all parts of ISO 20684 series may be used with any protocol but were designed with an expectation that they would be used with one of the ISO 15784-2 protocols.

By using this approach, agencies can specify open procurements and systems can be expanded geographically in an open and non-proprietary manner, which reduces costs, speeds up deployment, and simplifies integration.

0.2 Overview

SNMP is a collection of well-thought-out and well-proven concepts and principles. SNMP employs the sound principles of abstraction and standardization. This has led to SNMP being widely accepted as the prime choice for communication between management systems and devices on the internet and other communications networks.

The original implementation of SNMP was used to manage network devices such as routers and switches. Since then, the use of SNMP has grown into many areas of application on the internet and has also been used successfully over various serial communications networks.

This document defines management information for ITS field devices following the SNMP conventions.

0.3 Document approach and layout

This document defines:

a) the conformance requirements for this document (Clause 4);

b) a set of user needs for user-defined trigger conditions that can “fire” to initiate actions (Clause 5);

c) a set of detailed requirements for the identified user needs (Clause 6);

d) custom dialogues for the logging feature (Clause 7);

e) security considerations for the information defined in this document (Clause 8);

f) the management information bases that define the data for the defined requirements (Annex A);

g) the requirements traceability matrix (RTM) that traces the requirements to the design elements (Annex B).

Intelligent transport systems — Roadside modules SNMP data interface —

Part 7: Support features

1 Scope

Field devices are a key component in intelligent transport systems (ITS). Field devices include traffic signals, message signs, weather stations, traffic sensors, roadside equipment for connected ITS (C-ITS) environments, etc.

Field devices often need to exchange information with other external entities (managers). Field devices can be quite complex, necessitating the standardization of many data concepts for exchange. As such, the ISO 20684 series is divided several individual parts.

This document specifies user needs, requirements and design elements that are normatively used by other parts of the ISO 20684 series. Specifically, it defines an internal field device clock, a mechanism for grouping object values together to provide for more efficient transfer of data, and it provides formal requirements for the SNMP target and target parameters as defined in IETF RFC 3413.

NOTE 1 There are similarities between certain portions of NTCIP 1103 and NTCIP 1201 and this document.

NOTE 2 ISO 20684-1 provides additional details about how the ISO 20684 series relates to the overall ITS architecture.

2 Normative references

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

ISO/IEC 8825-1, *Information technology — ASN.1 encoding rules — Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*

ISO/IEC 8825-7, *Information technology — ASN.1 encoding rules — Part 7: Specification of Octet Encoding Rules (OER)*

ISO 20684-1:2021, *Intelligent transport systems — Roadside modules SNMP data interface — Part 1: Overview*

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 Frameworks, December 2002*.

IETF RFC 3413, *Simple Network Management Protocol (SNMP) Applications, December 2002*.

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 20684-1 apply.

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

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

4 Conformance

This clause follows the rules defined in ISO 20684-1. Table 1 traces each user need to a set of software features. Table 2 traces each feature to a set of requirements. Table 3 defines terms that are used as predicates in the conformance codes listed in Tables 1 and 2. For a full understanding of these tables and codes, see ISO 20684-1.

Table 1 — User need and feature conformance

Need	Requirement	Conformance
5.1: Efficient exchange of data		O
	6.4: Object group	M

Table 2 — Requirement conformance

Feature	Requirement	Conformance
6.1: Local clock		
	6.1.2.1: Configure local clock	M
	6.1.2.2: Confirm local clock configuration	M
	6.1.2.3: Determine local clock time	M
6.2: UTC clock		
	6.2.2.1: Discover UTC clock capabilities	M
	6.2.2.2: Configure UTC clock	M
	6.2.2.3: Confirm UTC clock configuration	M
	6.2.2.4: Determine UTC clock status	M
	6.2.2.5: Determine UTC time source status	M
	6.2.2.6: Set UTC time	snmp_time:M
	6.2.2.7: Retrieve UTC time	M
	6.2.2.8: Determine time discontinuity information	M
	6.2.3.1.1: Support for SNMP time configuration	O.1 (1..*)
	6.2.3.1.2: Support for network time protocol	O.1 (1..*)
	6.2.3.1.3: Support for radio time broadcast	O.1 (1..*)
	6.2.3.1.4: Support for satellite time signal	O.1 (1..*)
	6.2.3.2.1: Line frequency	O.2 (1..*)
	6.2.3.2.2: Crystal	O.2 (1..*)
6.3: Daylight saving time		
	6.3.2.1: Determine daylight saving time capabilities	M
	6.3.2.2: Configure daylight savings schedule	M
	6.3.2.3: Confirm daylight savings schedule configuration	M
	6.3.2.4: Determine daylight savings time active	M
	6.3.2.5: Determine daylight savings time adjustment	M

Table 2 *(continued)*

Feature	Requirement	Conformance
	6.3.3.1: Number of daylight saving entries	M
6.4: Object group		
	6.4.2.1: Determine object group capabilities	M
	6.4.2.2: Configure object group	M
	6.4.2.3: Confirm object group configuration	M
	6.4.2.4: Retrieve simple object group data	M
	6.4.2.5: Retrieve complex object group data	O
	6.4.2.6: Set new value for the object group	set:M
	6.4.2.7: Retrieve the last object group value set	set:M
	6.4.2.8: Retrieve object group statistics	M
	6.4.2.9: Clear object group	M
	6.4.2.10: Toggle object group	M
	6.4.2.11: Retrieve object group status	M
	6.4.3.1: Number of referenced objects	M
	6.4.3.2: Object group variable bindings size	M
	6.4.3.3: Object group basic encoding	M
	6.4.3.4: Object group octet encoding	O
	6.4.3.5: Support of set operations on object groups	O
	6.4.3.6: Support for complex object groups	O
6.5: SNMP target		
	6.5.2.1: Configure a target	M
	6.5.2.2: Confirm a target configuration	M
	6.5.2.3: Delete a target configuration	M
	6.5.2.4: Toggle a target	M
	6.5.2.5: Retrieve target row status	M
	6.5.2.6: Retrieve target statistics	M
	6.5.3.1.1: Support for SNMPv1 targets	O
	6.5.3.1.2: Support for SNMPv2c targets	O
	6.5.3.1.3: Support for SNMPv3 targets	M
	6.5.3.1.4: Support for STMP targets	O
	6.5.3.2.1: Support for UDP/IP targets	O
	6.5.3.2.2: Support for TCP/IP targets	O
	6.5.3.2.3: Support for DTLS/IP targets	M
	6.5.3.2.4: Support for TLS/IP targets	O
	6.5.3.3: Security profiles	M
6.6: SNMP target parameters		
	6.6.2.1: Configure SNMP target parameters	M
	6.6.2.2: Confirm configuration of SNMP target parameters	M
	6.6.2.3: Delete SNMP target parameters	M

Table 3 — External standard reference

Predicate	Clause
snmp_time	6.2.3.1.1
set	6.4.3.5

5 User needs

5.1 Efficient exchange of data

5.1.1 Object group user need

It is not uncommon for an SNMP manager to perform requests on the same group of managed object values from an SNMP agent multiple times. For example, an SNMP manager can potentially be configured to request a specific group of objects periodically to monitor the status of a device or when a certain set of predefined conditions occur. Normal SNMP operations require the complete object identifier (OID) of every object value to be specified in each request. These OIDs are often 10-20 octets in length, which can significantly increase communications overhead when the values being retrieved are small (e.g. one-octet INTEGERs). This document defines a mechanism to allow a manager to configure object groups, which can then be used to interact with multiple managed objects via a single OID.

6 Requirements

6.1 Local clock

6.1.1 Local clock definition

The local clock feature allows a field device to be aware of the local time as adjusted from the UTC clock by the local time zone and any daylight saving algorithms in effect, if the daylight saving feature is supported.

6.1.2 Local clock data exchange requirements

6.1.2.1 Configure local clock

The field device shall allow a manager to configure the local time zone for the clock.

6.1.2.2 Confirm local clock configuration

The field device shall allow a manager to confirm the configuration of the local clock.

6.1.2.3 Determine local clock time

The field device shall allow a manager to determine the local date and time.

6.2 UTC clock

6.2.1 UTC clock definition

The UTC clock feature allows a field device to maintain awareness of passing time; it represents the UTC time within the field device. However, there is no formal requirement as to the exact time source or accuracy. Ideally, it should be synchronized with a reliable and accurate time source in a secure manner.

6.2.2 UTC clock data exchange requirements

6.2.2.1 Discover UTC clock capabilities

The field device shall allow a manager to determine the capabilities of the UTC clock, including which time source and time-keeping options are available.

6.2.2.2 Configure UTC clock

The field device shall allow a manager to configure which available time source and time-keeping option to use as well as the frequency for synchronizing the clock and when to report discontinuities in time synchronization.

6.2.2.3 Confirm UTC clock configuration

The field device shall allow a manager to determine the configuration of the UTC clock.

6.2.2.4 Determine UTC clock status

The field device shall allow a manager to determine the time source and time-keeping option actually being used and the last time the clock was synchronized with the configured time source.

6.2.2.5 Determine UTC time source status

The field device shall allow a manager to determine the status of each supported time source.

6.2.2.6 Set UTC time

If a field device supports the configured time capability of the UTC clock feature, the field device shall allow a manager to set the current UTC time using SNMP.

NOTE This is perhaps the easiest time synchronization to implement, but also the least accurate. Variable multi-second communication network delays can occur between the time the manager initiates the set operation and the time the field device receives the command, especially when the command has to traverse a complex communications network.

6.2.2.7 Retrieve UTC time

The field device shall allow a manager to retrieve the current time in UTC.

NOTE Variable multi-second communication network delays can occur between the field device transmitting the time and the manager receiving it, especially when the command has to traverse a complex communications network.

6.2.2.8 Determine time discontinuity information

The field device shall allow a manager to determine information about discontinuities in its UTC time to enable to the detection of unusual behaviour in the clock.

NOTE The UTC time is expected to consistently increment its value. However, resynchronizing with an outside time source can result in an adjustment, known as a discontinuity.

6.2.3 UTC clock capabilities

6.2.3.1 Time source

6.2.3.1.1 Support for SNMP time configuration

The field device shall allow a manager to configure the time by SNMP.

6.2.3.1.2 Support for network time protocol

The field device allows the time to be set using the Network Time Protocol version 4 (NTPv4).

NOTE NTPv4 does not have any security. Devices that support NTPv4 are encouraged to provide mechanisms to physically disable access to this protocol and thereby close a potential security threat for systems that choose not to use this feature.

6.2.3.1.3 Support for radio time broadcast

The field device is able to set its time using a time signal broadcast over land-based radio waves.

6.2.3.1.4 Support for satellite time signal

The field device shall allow setting the time in the device based on satellite time signals.

6.2.3.2 Time-keeping

6.2.3.2.1 Line frequency

The field device shall allow time-keeping by monitoring the local electrical line frequency.

6.2.3.2.2 Crystal

The field device shall allow time-keeping with the use of an on-board time crystal.

6.3 Daylight saving time

6.3.1 Daylight saving time definition

The daylight saving time (DST) feature allows a manager to define when the local clock should spring forward and fall backwards to conform to local time customs. In addition to defining when the events occur, the design also allows the manager to specify how large of a shift is required. Finally, the design allows for multi-stage daylight savings adjustments (e.g. spring a half hour forward and then spring another half hour forward before falling back half an hour twice), but only requires support for a single stage. A procurement specification can require support for multi-stage changes, if deemed necessary.

6.3.2 Daylight saving time data exchange requirements

6.3.2.1 Determine daylight saving time capabilities

The field device shall allow a manager to determine the number of daylight saving time event pairs that the device supports.

NOTE A daylight saving time event pair consists of the spring forward and fall back events.

6.3.2.2 Configure daylight savings schedule

The field device shall allow a manager to configure the start and end times for daylight saving time as well as the amount of time that the local time should shift.

6.3.2.3 Confirm daylight savings schedule configuration

The field device shall allow a manager to confirm the configuration of the daylight savings schedule.

6.3.2.4 Determine daylight savings time active

The field device shall allow a manager to determine the status of each daylight savings time rule.

6.3.2.5 Determine daylight savings time adjustment

The field device shall allow a manager to determine the total of the current daylight savings time adjustments.

6.3.3 Daylight saving time capability requirements

6.3.3.1 Number of daylight saving entries

If a field device supports the local clock feature, the field device shall support at least one annual daylight saving adjustment.

6.4 Object group

6.4.1 Object group definition

An object group can minimize the amount of communications overhead for requests that will be regularly repeated. Each object group value is a user-configured sequence of data elements from the device. The data can be stored or retrieved without including the OID for each individual field within the object group while also using the more efficient octet encoding rules (OER) rather than the normal basic encoding rules.

6.4.2 Object group data exchange

6.4.2.1 Determine object group capabilities

6.4.2.2 Configure object group

A field device shall allow a manager to configure the objects to be contained within an object group.

6.4.2.3 Confirm object group configuration

A field device shall allow a manager to determine the configuration of an object group.

6.4.2.4 Retrieve simple object group data

A field device shall allow a manager to retrieve the values of each object within an object group through a simple request.

6.4.2.5 Retrieve complex object group data

A field device shall allow a manager to retrieve the values of each object within an object group using a two-step process, designed to prevent timeouts for requests that require significant processing time.

6.4.2.6 Set new value for the object group

A field device shall allow a manager to perform a set operation on the object group.

6.4.2.7 Retrieve the last object group value set

A field device shall allow a manager to retrieve the last value submitted in a set request for the object group.

6.4.2.8 Retrieve object group statistics

A field device shall allow a manager to retrieve statistics about when the last retrieval operation was made and the estimated duration for requests.

6.4.2.9 Clear object group

The field device shall allow a manager to clear the definition of an object group.

6.4.2.10 Toggle object group

The field device shall allow a manager to enable/disable an object group.

6.4.2.11 Retrieve object group status

The field device shall allow a manager to retrieve the status of an object group.

6.4.3 Object group capabilities

6.4.3.1 Number of referenced objects

The field device shall support at least two referenced objects for each object group.

NOTE An object group only offers a significant benefit if it compresses at least two objects. A device can perhaps want to establish limits on the number of objects within a group to prevent unreasonably complex requests.

6.4.3.2 Object group variable bindings size

The field device shall support values of the object group data up to a maximum size, which is required to be at least 400 octets.

NOTE The 400-octet limit is based on the standard conformant SNMP data packet size minus adequate room for an SNMP header.

6.4.3.3 Object group basic encoding

The field device shall support encoding object groups as per the basic encoding rules (ISO 8825-1).

6.4.3.4 Object group octet encoding

A field device shall support encoding object groups as per the octet encoding rules (ISO 8825-7).

6.4.3.5 Support of set operations on object groups

The field device shall support set operations on all object groups that only contain writable objects.

6.4.3.6 Support for complex object groups

The field device shall support a two-step request process.

NOTE Otherwise, any object group defined can potentially use the one-step process and complex definitions. If allowed, this can potentially require a significant amount of processing time before a response is generated.

6.5 SNMP target

6.5.1 Target definition

A target represents a remote SNMP entity that can receive messages from the field device. The remote SNMP entity can be a manager that can receive SNMP notifications or can be a remote field device that can receive SNMP requests.

6.5.2 Target data exchange requirements

6.5.2.1 Configure a target

The field device shall allow a manager to configure a target, including its address, security and communication parameters.

6.5.2.2 Confirm a target configuration

The field device shall allow a manager to determine the configuration of a target.

6.5.2.3 Delete a target configuration

The field device shall allow a manager to clear the configuration of a target.

6.5.2.4 Toggle a target

The field device shall allow a manager to enable/disable a target.

6.5.2.5 Retrieve target row status

The field device shall allow a manager to retrieve the status of a target entry row.

6.5.2.6 Retrieve target statistics

The field device shall allow a manager to retrieve statistics about the targets.

6.5.3 Target capabilities

6.5.3.1 Application profiles

6.5.3.1.1 Support for SNMPv1 targets

The field device shall support the SNMPv1 application profile when sending information to targets.

6.5.3.1.2 Support for SNMPv2c targets

The field device shall support the SNMPv2c application profile when sending information to targets.

6.5.3.1.3 Support for SNMPv3 targets

The field device shall support the SNMPv3 application profile when sending information to targets.

6.5.3.1.4 Support for STMP targets

The field device shall support the STMP application profile when sending information to targets.

6.5.3.2 Transport profiles

6.5.3.2.1 Support for UDP/IP targets

The field device shall support the UDP/IP transport profile when sending information to targets.

6.5.3.2.2 Support for TCP/IP targets

The field device shall support the TCP/IP transport profile when sending information to targets.

6.5.3.2.3 Support for DTLS/IP targets

The field device shall support the DTLS/UDP/IP transport profile when sending information to targets.

6.5.3.2.4 Support for TLS/IP targets

The field device shall support the TLS/TCP/IP transport profile when sending information to targets.

6.5.3.3 Security profiles

The field device shall support the following security profiles when sending messages to targets:

a) AES encryption,

b) SHA-2 Authentication.

NOTE The configuration of the SNMP Target Parameters table defines the rules on whether authentication and/or encryption is required to access data. For example, one set of access parameters can potentially be defined to access some subset of data without any authentication or encryption while another set of parameters can potentially provide full read-write access, but only with authentication and encryption.

6.6 SNMP target parameters

6.6.1 SNMP target parameters definition

The SNMP target parameters define a set of communication parameters for communicating with SNMP targets.

6.6.2 SNMP target parameters data exchange requirements

6.6.2.1 Configure SNMP target parameters

The field device shall allow a manager to configure an SNMP target parameters entry, including the message model and security settings.

6.6.2.2 Confirm configuration of SNMP target parameters

The field device shall allow a manager to confirm the configuration of an SNMP target parameters entry.

6.6.2.3 Delete SNMP target parameters

The field device shall allow a manager to delete an SNMP target parameters entry.

7 Dialogues

7.1 Retrieve a complex object group

The retrieval of a complex object group can require more time than a normal SNMP operation; this dialogue provides a two-step process by which data can be retrieved where each step can be more responsive. The standardized dialogue shall be as follows.

a) The dialogue shall be initiated by the manager. The logic used by the manager to initiate the dialogue is outside the scope of this document. Prior to initiating the dialogue, the manager shall be aware of which entry (x) in the table it wishes to retrieve.

b) The manager shall send a GetRequest-PDU for fdObjectGroupRowStatus.x and fdObjectGroupRefresh.x.

   1) If the value returned for fdObjectGroupRowStatus.x is not 'active', the dialogue shall terminate.

   2) If the value returned for fdObjectGroupRefesh.x is 'oneStep', the dialogue shall terminate and the manager shall use the one-step method for retrieving the object value.

   3) If the value returned for fdObjectGroupRefesh.x is 'ready', the dialogue shall continue to step c).

   4) If the value returned for fdObjectGroupRefesh.x is anything else, the dialogue shall terminate.

c) The manager shall send a SetRequest-PDU to set fdObjectGroupRefresh.x to 'refresh'.

d) The manager shall send a GetRequest-PDU for fdObjectGroupRefresh.x. If the value returned is 'pending', the manager will repeat this step (unless the manager decides to abort the operation). If the value returned is 'ready', the manager shall proceed to step e); if the value returned is anything else, the manager shall terminate the dialogue.

e) The manager shall send a GetRequest-PDU for fObjectGroupCurrentValue.x. The returned value represents the value of the object group.

7.2 Set an object group value

This dialogue provides a process by which data values for multiple objects can be stored in the device using a single object group. The standardized dialogue shall be as follows.

a) The dialogue shall be initiated by the manager. The logic used by the manager to initiate the dialogue is outside the scope of this document. Prior to initiating the dialogue, the manager shall be aware of which entry (x) in the table it wishes to use.

b) The manager shall send a GetRequest-PDU for fdObjectGroupRowStatus.x, fdObjectGroupRefresh.x, and fdObjectGroupLastError.x.

   1) If the value returned for fdObjectGroupRowStatus.x is not 'active', the dialogue shall terminate.

   2) If the value returned for fdObjectGroupRefesh.x is not 'oneStep' or 'ready', the dialogue shall terminate.

   3) If the value returned for fdObjectGroupLastError is 'pending' the dialogue shall terminate.

   4) Otherwise, the dialogue shall continue to step c).

c) The manager shall send a SetRequest-PDU to set fdObjectGroupNewValue.x to the desired value.

   1) If the value returned for fdObjectGroupRefesh.x in step b) was 'oneStep', the dialogue has completed and the result of the set request is contained within the response to step c).

2) If the value returned for fdObjectGroupRefesh.x in step b) was 'ready' and the result from step c) indicates an error, the dialogue shall terminate. This condition indicates that the field device was unable to decode the object group as per the current configuration.

3) If the value returned for fdObjectGroupRefesh.x in step b) was 'ready' and the result from step c) did not indicate an error, proceed to step d).

d) The manager shall send a GetRequest-PDU for fdObjectGroupLastError.x and fdObjectGroupLastErrorIndex.x. These objects will indicate the success or failure of the request with an indication of which field within the object group caused any error.

7.3 Clear an object group list

This dialogue provides a process by which the fields within an object group can be cleared. The standardized dialogue shall be as follows.

a) The dialogue shall be initiated by the manager. The logic used by the manager to initiate the dialogue is outside the scope of this document. Prior to initiating the dialogue, the manager shall be aware of which entry (x) in the table it wishes to use.

b) The manager shall send a GetRequest-PDU for fdObjectGroupRowStatus.x.

1) If the response indicates that fdObjectGroupRowStatus.x does not exist, the dialogue shall terminate (i.e. the object group is already cleared).

2) If the response value for fdObjectGroupRowStatus.x is 'active', the manager shall send a SetRequest-PDU to set fdObjectGroupRowStatus.x to 'notInService'. If this results in an error, the dialogue shall terminate.

c) The manager shall send a SetRequest-PDU to set fdObjectGroupClear.x to 'true'.

NOTE At the end of this process, the object group entry will be in the 'notReady' state since the entry will not contain valid information to be used.

8 Security vulnerabilities

There are data elements defined in this document with a MAX-ACCESS clause of read-write and/or read-create. These and other data elements are sensitive and need to be protected from malicious and inadvertent manipulation and/or disclosure. The support for requests in a non-secure environment without proper protection can have a negative effect on network operations. A sampling of the vulnerabilities includes:

a) the ability to change the time, information within object groups, and/or the remote agents that the field device will connect to;

b) the ability to delete object groups, or remote devices;

c) the ability to create additional object groups and other remote devices that can receive information; and

d) the ability to monitor current configurations.

To overcome these vulnerabilities, it is highly recommended that SNMPv3 with TLS support, as defined in RFC 6353, is used to exchange the data.

Annex A
(normative)

Management information base (MIB)

This annex provides definitions which it is useful to import into other management information base (MIB) modules of this document.

A.1 Clock MIB

```
-- ****************************************************************************
-- A.1.1 Clock Header
-- ****************************************************************************

-- ASN1START
CLOCK-MIB DEFINITIONS ::= BEGIN
IMPORTS

MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Integer32, TimeTicks,
Unsigned32
                                                    FROM SNMPv2-SMI
                                                      -- RFC 2578

TruthValue, StorageType, RowStatus
                                                    FROM SNMPv2-TC
                                                      -- RFC 2579

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

ITSUnsigned8, ITSInteger16, ITSUnsigned16, ITSDayOfWeek, ITSDayOfMonth,
ITSMonth, ITSDailyTimeStamp, ITSDateStamp, fieldDevice, iso20684p7
                                                    FROM FIELD-DEVICE-TC-MIB
                                                      -- ISO 20684-1 Annex A

;
fdClockMIB MODULE-IDENTITY
  LAST-UPDATED "202001082121Z"
  ORGANIZATION "ISO TC 204 WG 9"
  CONTACT-INFO
   "name: Kenneth Vaughn
    phone: +1-571-331-5670
    email: kvaughn@trevilon.com
    postal: 6606 FM 1488 RD STE 148-503
            Magnolia, TX 77354
            USA"
  DESCRIPTION
    "This MIB defines a generalized clock that can be used within a field device
    to track time, including an optional local time with a time zone and
    daylight saving time (a.k.a., 'summer time') offsets."

  REVISION "202001082121Z"
  DESCRIPTION
    "Initial revision of the document as proposed for CD ballot"

  ::= {iso20684p7 1}


-- ****************************************************************************
-- A.1.2 Node Definitions
-- ****************************************************************************
```

```
fdClockConformance OBJECT-IDENTITY
  STATUS      current
  DESCRIPTION
    "A node containing conformance statements related to the fdClockMIB, as
    defined in ISO/TS 20684-7."
  ::= {fdClockMIB 2}

fdClockCompliances OBJECT-IDENTITY
  STATUS      current
  DESCRIPTION
    "A node for compliance statements for the fdClockMIB."
  ::= {fdClockConformance 1}

fdClockGroups OBJECT-IDENTITY
  STATUS      current
  DESCRIPTION
    "A node for group definitions related to fdClockMIB."
  ::= {fdClockConformance 2}

fdClock OBJECT-IDENTITY
  STATUS      current
  DESCRIPTION
    "A node defining management information related to the field device's
    Clock."
  ::= {fieldDevice 9}

fdClockUtcTime OBJECT-TYPE
  SYNTAX      ITSDailyTimeStamp
  MAX-ACCESS read-write
  STATUS      current
  DESCRIPTION
    "The number of milliseconds that have elapsed since 00:00:00.000 (midnight)
    Coordinated Universal Time (UTC). Exactly one second past midnight UTC
    would be represented as 1000."
  REFERENCE  "NTCIP 1201 Clause 2.4.1"

  DEFVAL      {0}
  ::= {fdClock 1}
-- NTCIP violated standard SNMP conventions when it defined time as a
-- read-writable Counter; Counters are supposed to be read-only. The
-- NTCIP object also does not support the millisecond resolution that
-- is increasingly needed for field systems. To provide millisecond
-- resolution with a viable time range, the time object would need to
-- be more than a 4-byte integer, but SNMP does not natively support
-- such values. Thus, another mechanism is needed. The standard SNMPv3
-- TimeStamp textual Convention also fails to provide the necessary
-- resolution (it only supports deciseconds and is represented in ASCII
-- and requires much more space). Thus, this draft proposes to divide
-- the timestamp into two components, a time of day object and a date
-- object.

fdClockUtcDate OBJECT-TYPE
  SYNTAX      ITSDateStamp
  MAX-ACCESS read-write
  STATUS      current
  DESCRIPTION
    "The Gregorian calendar date represented as the OER-encoded structure of:

      SEQUENCE {
        year INTEGER (0..65535),
        month INTEGER (1..12),
        date INTEGER (1..31) }

    An attempt to SET this object to a value of a non-existent date (e.g., 29
    February 2019), shall cause the the field device to respond with a
    'wrongValue' error."
  REFERENCE  "NTCIP 1201 Clause 2.4.1"
  DEFVAL      {'07D00101'H}
  ::= {fdClock 2}
-- Using a structure rather than a sequential integer makes it easier
-- to read and is consistent with DateCompact as defined in ISO 14906,
```

```
-- although we use OER rather than PER to ease decoding and to allow
-- the range to go WELL beyond 2117. The date 1 March 2020 would be
-- represented as '07E40301'H.

fdClockResolution OBJECT-TYPE
  SYNTAX     INTEGER (1..1000)
    UNITS      "1 millisecond"
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The resolution of the internal clock as reported by fdClockUtcTime. A
    clock with a resolution of 1 second would report a value of 1000; a clock
    with a resolution of 1 millisecond would report a value of 1."
  ::= {fdClock 3}
-- Proposed object; the resolution indicates the step size of each
-- increment of the clock.

fdClockSupportedSources OBJECT-TYPE
  SYNTAX     BITS {other (0), snmp (1), network (2), radio (3), satellite (4)}
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The clock synchronization sources supported by the device.
    other     - one or more sources not defined by this standard
    snmp      - allows setting fdClockUtcTime via SNMP or other mechanism that
                may be subject to network delays
    network   - network time protocol (NTP) or similar mechanism that includes
                logic to compensate for network delays to provide
                accuracies better than 100 ms
    radio     - terrestrial radio-based time broadcast, such as WWV
    satellite - satellite-based signal such as provided by global navigation
                system satellites"
  REFERENCE  "NTCIP 1202 5.4.22.2.2"
  ::= {fdClock 4}
-- Proposed object; NTCIP 1202 combines the concept of time source
-- with time-keeping mechanism. This draft proposed to separate these
-- two concepts.

fdClockRequestedSource OBJECT-TYPE
  SYNTAX     INTEGER { other (1), snmp (2), network (3),
               radio (4), satellite (5), local (6) }
  MAX-ACCESS read-write
  STATUS     current
  DESCRIPTION
    "The requested clock synchronization source. The value of this object may
    default to 'local' to indicate that the device does not automatically
    attempt to synchronize with any external source. The value of this object
    shall automatically transition to 'snmp' whenever fdClockUtcTime or
    fdClockUtcDate is successfully set via an 'snmp' interface. If a request is
    received to set the value to 'other', 'snmp', or 'local', the field device
    shall return a 'wrongValue' error. If the value of this object is
    'network', 'radio', or 'satellite', the device shall attempt to synchronize
    to the requested source at least once every fdClockSyncCycle. If the field
    device fails to synchronize with the requested source, or fails to be set
    via 'snmp' when in the 'snmp' mode, for a period exceeding the
    fdClockSyncCycle duration, the device may attempt to synchronize with an
    alternate source. However, the field device shall continue to attempt to
    synchronize with the requested source at least once every fdClockSyncCycle.
    The device shall not synchronize with another source if synchronization
    with the requested source is successful."
  REFERENCE  "NTCIP 1202 5.4.22.3"
  ::= {fdClock 5}
-- Proposed object; Separates the time source from time-keeping.

fdClockSource OBJECT-TYPE
  SYNTAX     INTEGER { unknown (0), other (1), snmp (2),
               network (3), radio (4), satellite (5), local (6) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The source used for the most recent clock synchronization. When operating
```

```
    as expected, this value should reflect the value of fdClockRequestedSource
    once the clock synchronizes. If the clock has not been able to successfully
    synchronize with the requested source within the last fdClockSyncCycle,
    this object shall reflect the current source; if the field device has not
    successfully synchronized with any external source, this object shall
    report a value of 'local'."
  REFERENCE  "NTCIP 1202 5.4.22.4"
  ::= {fdClock 6}
-- Proposed object; the clock source is an indication of the accuracy
-- of the clock when it was last synchronized.

fdClockRequestedSourceStatus OBJECT-TYPE
  SYNTAX     INTEGER { other (1), normal (2), dataError (3),
               timeout (4), pending (5), discontinuity (6) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The status of the requested time source for the clock
    (fdClockRequestedSource).
    other: The status of the requested source is not represented by any of the
           other states.
    normal: The source is providing information as expected.
    dataError: The last update from the source contained an error in the
           supplied data.
    timeout: The source has failed to successfully provide an update within the
           fdClockSyncCycle plus the implementation defined grace period.
    pending: The field device has initiated a process to obtain an updated time
           or is otherwise expecting an update from the time source; while
           the field device has not yet successfully received an update the
           time since the last update is still within the fdClockSyncCycle
           plus any implementation-defined grace period.
    discontinuity: The source provided an update that resulted in a
           discontinuity in the time greater than
           fdClockDiscontinuityMaxAdjustment; when this occurs, the status
           will return to 'normal' after 10 seconds or when the status is
           read via SNMP, which ever comes first."
  ::= {fdClock 7}
-- Proposed object; in addition to the status of the current source,
-- it is useful to indicate why the requested source is not active.

fdClockSourceStatus OBJECT-TYPE
  SYNTAX     INTEGER { other (1), normal (2), dataError (3),
               timeout (4), pending (5), discontinuity (6) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The status of the current time source for the clock (fdClockSource).
    other: The status of the requested source is not represented by any of the
           other states.
    normal: The source is providing information as expected.
    dataError: The last update from the source contained an error in the
           supplied data.
    timeout: The source has failed to successfully provide an update within the
           fdClockSyncCycle plus the implementation defined grace period.
    pending: The source has failed to successfully provide an update within the
           fdClockSyncCycle, but is still within the implementation defined
           grace period.
    discontinuity: The source provided an update that resulted in a
           discontinuity greater than fdClockDiscontinuityMaxAdjustment; when
           this occurs, the status will return to 'normal' after 10 seconds or
           until the status is read via SNMP, which ever comes first."
  REFERENCE  "NTCIP 1202 5.4.22.5"
  ::= {fdClock 8}

fdClockSyncCycle OBJECT-TYPE
  SYNTAX     INTEGER {
               millisecond (1),
               centisecond (2),
               decisecond (3),
               second (4),
               ten-seconds (5),
```

```
                minute (6),
                ten-minutes (7),
                hour (8),
                six-hours (9),
                day (10),
                week (11),
                month (12) }
  MAX-ACCESS read-write
  STATUS     current
  DESCRIPTION
    "The amount of time between each synchronization with the time source.
    The device clock shall synchronize at least once every indicated cycle."
  DEFVAL     {day}
  ::= {fdClock 9}
-- Proposed object; it seems appropriate to allow this to be
-- configurable although the specific values supported may vary by
-- device type/implementation.

fdClockLastSyncTime OBJECT-TYPE
  SYNTAX     ITSDailyTimeStamp
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The UTC time at which the last synchronization to an external reference
    clock was made."
  ::= {fdClock 10}
-- Proposed object

fdClockLastSyncDate OBJECT-TYPE
  SYNTAX     ITSDateStamp
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The UTC date on which the last synchronization to an external reference
    clock was made."
  ::= {fdClock 11}
-- Proposed object

fdClockSupportedTimeKeeping OBJECT-TYPE
  SYNTAX     BITS { other (0), lineFrequency (1), rtcSqareWave (2),
               crystal (3), external (4) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The mechanisms by which the fdClockUtcTime object is able to track the
    passage of time.
    other - a source not defined by this standard
    lineFrequency - tracking the AC power line frequency
    rtcSqareWave - tracking a real-time clock square wave
    crystal - using an internal time crystal
    external - the synchronizes with the fdClockSource at least once a minute"
  ::= {fdClock 12}
-- Proposed object

fdClockRequestedTimeKeeping OBJECT-TYPE
  SYNTAX     INTEGER { other (1), lineFrequency (2), rtcSqareWave (3),
               crystal (4), external (5) }
  MAX-ACCESS read-write
  STATUS     current
  DESCRIPTION
    "The requested mechanism to keep time within the fdClockUtcTime object
    between synchronization events.
    other - use a source not defined by this standard
    lineFrequency - track the AC power line frequency
    rtcSqareWave - track a real-time clock square wave
    crystal - use the internal time crystal
    external - synchronize with the fdClockSource at least once a minute"
  ::= {fdClock 13}
-- Proposed object

fdClockTimeKeeping OBJECT-TYPE
```

```
  SYNTAX     INTEGER { unknown (0), other (1), lineFrequency (2),
               rtcSqareWave (3), crystal (4), external (5) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The mechanism currently being used to keep time within the fdClockUtcTime
    object between synchronization events.
    unknown - the source is unknown or cannot be determined; this is considered
        an error state
    other - a source not defined by this standard
    lineFrequency - tracking the AC power line frequency
    rtcSqareWave - tracking a real-time clock square wave
    crystal - using an internal time crystal
    external - synchronizes with the fdClockSource at least once a minute"
  ::= {fdClock 14}
-- Proposed object; the time-keeping value indicates the precision of
-- the clock, i.e., how well it keeps time once it has been set.

fdClockDiscontinuitySource OBJECT-TYPE
  SYNTAX     INTEGER {
               unknown (0),            other (1),
               snmp (2),               network (3),
               radio (4),              satellite (5),
               local (6),              changedUnknown (128),
               changedOther (129),     changedSnmp (130),
               changedNetwork (131),   changedRadio (132),
               changedSatellite (133), changedLocal (134) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The source of the time that caused the discontinuity.
    unknown - The source of the discontinuity is unknown
    other - The discontinuity and previous source were both 'other'
    snmp - The discontinuity and previous source were both 'snmp'
    network - The discontinuity and previous source were both 'network'
    radio - The discontinuity and previous source were both 'radio'
    satellite - The discontinuity and previous source were both 'satellite'
    local - The discontinuity and previous source were both 'local'
    changedOther - The discontinuity source was 'other', but the previous
        source was not 'other'
    changedSnmp - The discontinuity source was 'snmp', but the previous source
        was not 'snmp'
    changedNetwork - The discontinuity source was 'network', but the previous
        source was not 'network'
    changedRadio - The discontinuity source was 'radio', but the previous
        source was not 'radio'
    changedSatellite - The discontinuity source was 'satellite', but the
        previous source was not 'satellite'
    changedLocal - The discontinuity source was 'local', but the previous
        source was not 'local'"
  REFERENCE  "NTCIP 1202 5.4.22.6"

  ::= {fdClock 15}
-- NTCIP uses a completely different enumeration with the following mapping:
--    - unknown: unknown
--    - dstChange: not needed as this design only looks at UTC time
--    - managementStation: snmp
--    - localUser: local
--    - timeSourceChange: 'changed' values
--    - timeSourceDiscontinuity: non-changed values.

fdClockDiscontinuityDelta OBJECT-TYPE
  SYNTAX     Integer32
    UNITS      "milliseconds"
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The number of milliseconds that the discontinuity advanced the clock. A
    negative value indicates that the new time moved the clock backwards. A
    value of 0x80 00 00 00 indicates an unknown state (e.g., the clock has not
    been set since the last reboot). The range accounts for any delta within
```

```
    24+ days; deltas outside of the range of this object shall indicate 0x80 00
    00 01 for excessive roll-backs and 0x7F FF FF FF for excessive advances of
    the clock."
  REFERENCE  "NTCIP 1202 5.4.22.8"
  ::= {fdClock 16}

fdClockDiscontinuityUpTime OBJECT-TYPE
  SYNTAX     TimeTicks
    UNITS      "centiseconds"
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The value of sysUpTime when the discontinuity value was implemented."
  REFERENCE  "NTCIP 1202 5.4.22.7"
  ::= {fdClock 17}
-- NTCIP bases the value on the updated time value; this proposal uses
-- sysUpTime as a more neutral reference.

fdClockDiscontinuityMaxAdjustment OBJECT-TYPE
  SYNTAX     ITSUnsigned16
    UNITS      "milliseconds"
  MAX-ACCESS read-write
  STATUS     current
  DESCRIPTION
    "The minimum number of milliseconds that the clock must be adjusted by
    (forwards or backwards) in order for the field device to record a
    discontinuity. The field device shall not support values less than its
    fdClockResolution value."
  ::= {fdClock 18}
-- Proposed object

-- ********************************************************************
-- ClockLocal
-- ********************************************************************

fdClockLocal OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node for managing the local clock parameters."
  ::= {fdClock 19}

fdClockLocalStandardTimeZone OBJECT-TYPE
  SYNTAX     INTEGER (-46800..46800)
    UNITS      "1 second"
  MAX-ACCESS read-write
  STATUS     current
  DESCRIPTION
    "The number of seconds between UTC time and the local standard time zone
    (not considering any daylight savings time adjustments). Positive values
    represent time zones in the eastern hemisphere; negative values represent
    time zones in the western hemisphere."
  REFERENCE  "NTCIP 1201 Clause 2.4.6"
  DEFVAL     {0}
  ::= {fdClockLocal 1}
-- NTCIP allows +/- 12 hours; this allows +/- 13 hours to support NZ

fdClockLocalTime OBJECT-TYPE
  SYNTAX     ITSDailyTimeStamp
    UNITS      "milliseconds"
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The local time of day measured in milliseconds from local midnight; the
    value shall reflect the sum of fdClockUtcTime +
    (fdClockLocalStandardTimeZone * 1000) + (fdClockLocalDstAdjustment *
    1000)."
  REFERENCE  "NTCIP 1201 Clause 2.4.7"
  ::= {fdClockLocal 2}
-- Split into two in a similar fashion as fdClockUtcTime

fdClockLocalDate OBJECT-TYPE
```

```
  SYNTAX     ITSDateStamp
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The local Gregorian Date per the values of fdClockUtcTime,
    fdClockLocalStandardTimeZone, and fdClockLocalDstAdjustment."
  REFERENCE  "NTCIP 1201 Clause 2.4.7"
  ::= {fdClockLocal 3}

fdClockLocalDstAdjustment OBJECT-TYPE
  SYNTAX     ITSInteger16
    UNITS      "1 second"
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The number of seconds of the current offset due to daylight savings time.
    The value of this object shall be sum of the fdClockLocalDstOffset values
    from all active rows of the fdClockLocalDstTable where
    fdClockLocalDstApplied is equal to 'true'; if no such rows exist, the value
    of this object shall be zero (0)."
  ::= {fdClockLocal 4}
-- This allows for a total offset of 9 hours in either direction;
-- should also assist in debugging


-- ********************************************************************
-- ClockDst
-- ********************************************************************

fdClockDst OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node for managing daylight saving parameters."
  ::= {fdClock 20}

fdClockDstMaxEntries OBJECT-TYPE
  SYNTAX     Unsigned32
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The maximum number of entries allowed within the fdClockDstTable."
  ::= {fdClockDst 1}

fdClockDstTable OBJECT-TYPE
  SYNTAX     SEQUENCE OF FdClockDstEntry
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "A dynamic table where each entry defines the characteristics of a daylight
    savings time rule (e.g., when it begins, when it ends, and how much of an
    offset is created)."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2"
  ::= {fdClockDst 2}

fdClockDstEntry OBJECT-TYPE
  SYNTAX     FdClockDstEntry
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "The definition of a daylight saving time event pair."
  INDEX      {fdClockDstIndex}
  ::= {fdClockDstTable 1}

FdClockDstEntry ::= SEQUENCE {
               fdClockDstIndex            ITSUnsigned8,
               fdClockDstBeginMonth       ITSMonth,
               fdClockDstBeginOccurrences INTEGER,
               fdClockDstBeginDayOfWeek   ITSDayOfWeek,
               fdClockDstBeginDayOfMonth  ITSDayOfMonth,
               fdClockDstBeginTime        ITSDailyTimeStamp,
               fdClockDstEndMonth         ITSMonth,
```

```
                fdClockDstEndOccurrences    INTEGER,
                fdClockDstEndDayOfWeek      ITSDayOfWeek,
                fdClockDstEndDayOfMonth     ITSDayOfMonth,
                fdClockDstEndTime           ITSDailyTimeStamp,
                fdClockDstOffset            ITSInteger16,
                fdClockDstApplied           TruthValue,
                fdClockDstStorageType       StorageType,
                fdClockDstRowStatus         RowStatus   }

fdClockDstIndex OBJECT-TYPE
  SYNTAX     ITSUnsigned8
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "An index used to identify the row within the fdClockDstTable."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.1"
  ::= {fdClockDstEntry 1}

fdClockDstBeginMonth OBJECT-TYPE
  SYNTAX     ITSMonth
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "Indicates the month that is combined with fdClockLocalDstBeginDayOfMonth
    to determine the base date used to calculate the start of daylight savings
    time using other object values in this row. This is often the month in
    which daylight savings time begins, but may not be, for example, if the
    value of this object is December, beginDayOfMonth is 31, and
    beginOccurences is 'second'; daylight savings time will begin on the second
    week after 31 December, which would be in January of the following year."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.2"
  ::= {fdClockDstEntry 2}
-- We drop the disabled value as we have added a RowStatus object that
-- achieves this purpose in a more standardized way; we dropped the
-- absolute value as it is not needed and it complicates the design
-- with a two part time.

fdClockDstBeginOccurrences OBJECT-TYPE
  SYNTAX     INTEGER {
               first (1),       second (2),
               third (3),       fourth (4),
               last (5),        secondToLast (6),
               thirdToLast (7), fourthToLsat (8),
               specificDayOfMonth (9) }
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "If the value is 1-4, daylight savings time shall begin on the <named value
    of this object> <beginDayOfWeek> on or after <beginDayOfMonth><beginMonth>.
    If the value is 5-8, daylight savings time shall begin on the <named value
    of this object> <beginDayOfWeek> on or before <beginDayOfMonth>
    <beginMonth>. If the value is 9, daylight savings time shall begin on the
    <beginDayOfMonth> <beginMonth>. For example, if the value of this object is
    2, the value of fdClockDstBeginDayOfWeek is 7, fdClockDstBeginDayOfMonth is
    8 and fdClockDstBeginMonth is 3; daylight savings time would begin on the
    <second> <Sunday> on or after <8> <March>."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.3"
  DEFVAL     {first}
  ::= {fdClockDstEntry 3}

fdClockDstBeginDayOfWeek OBJECT-TYPE
  SYNTAX     ITSDayOfWeek
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "Identifies the day of week on which daylight savings time begins."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.4"
  DEFVAL     {sunday}
  ::= {fdClockDstEntry 4}
-- NTCIP defines 1 to be Sunday; we have converted to be consistent
-- with ISO 8601 and ISO 17575 convention of 1 = Monday.
```

```
fdClockDstBeginDayOfMonth OBJECT-TYPE
  SYNTAX     ITSDayOfMonth
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "Identifies the day of the fdClockDstBeginMonth from which to calculate the
    start date of daylight savings. See fdClockDstBeginOccurrences for more
    details."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.5"
  DEFVAL     {1}
  ::= {fdClockDstEntry 5}

fdClockDstBeginTime OBJECT-TYPE
  SYNTAX     ITSDailyTimeStamp
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "Indicates the nominal (local) time of day when the offset shall be applied
    in milliseconds past midnight. The nominal milliseconds past midnight may
    be different than actual milliseconds past midnight if the fdClockLocalTime
    was adjusted for any reason (e.g., another entry in this table)."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.6"
  DEFVAL     {7200000}
  ::= {fdClockDstEntry 6}
-- NTCIP defines an invalid 32-bit unsigned integer for this to support
-- an absolute mode; this will cause problems in 20 years. We have
-- paralleled the format used above for time

fdClockDstEndMonth OBJECT-TYPE
  SYNTAX     ITSMonth
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The month that is combined with fdClockDstEndDayOfMonth to determine the
    base date used to calculate the end of daylight savings time using other
    object values in this row. This is often the month in which daylight
    savings time ends, but may not be. For example, if the value of this object
    is December, endDayOfMonth is 31, and endOccurences is 'second'; daylight
    savings time will end in January of the following year."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.7"
  DEFVAL     {january}
  ::= {fdClockDstEntry 7}

fdClockDstEndOccurrences OBJECT-TYPE
  SYNTAX     INTEGER {
               first (1),          second (2),
               third (3),          fourth (4),
               last (5),           secondToLast (6),
               thirdToLast (7), fourthToLast (8),
               specificDayOfMonth (9) }
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "If the value is 1-4, daylight savings time shall end on the <named value of
    this object> <endDayOfWeek> on or after <endDayOfMonth> <endMonth>. If the
    value is 5-8, daylight savings time shall end on the <named value of this
    object> <endDayOfWeek> on or before <endDayOfMonth> <endMonth>. If the
    value is 9, daylight savings time shall end on the <endDayOfMonth> of
    <endMonth>."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.8"
  DEFVAL     {first}
  ::= {fdClockDstEntry 8}

fdClockDstEndDayOfWeek OBJECT-TYPE
  SYNTAX     ITSDayOfWeek
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "Identifies the day of week on which daylight savings time ends."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.9"
```

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

fdClockDstEndDayOfMonth OBJECT-TYPE
  SYNTAX     ITSDayOfMonth
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The day of the month from which to calculate the end of daylight savings
    time per the rules defined in endOccurrences."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.10"
  DEFVAL     {1}
  ::= {fdClockDstEntry 10}

fdClockDstEndTime OBJECT-TYPE
  SYNTAX     ITSDailyTimeStamp
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "Indicates the time of day when the offset specified in this row shall be
    removed (if it is currently applied). For example, if the value of this
    object is 1800000 and offset is 3600000, the daylight savings time offset
    will be removed at the first instance of 00:30 on the identified end date.
    Removing the offset in this case would cause the local time to fall back to
    23:30 on the previous day and then the clock would run normally."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.11"
  DEFVAL     {7200000}
  ::= {fdClockDstEntry 11}

fdClockDstOffset OBJECT-TYPE
  SYNTAX     ITSInteger16
    UNITS      "1 second"
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The number of seconds to advance the local clock when this row of daylight
    savings time is in effect. The total adjustment made to the local clock
    shall be the sum of all rows where the daylight savings time is in effect."
  REFERENCE  "NTCIP 1201 Clause 2.4.8.2.12"
  DEFVAL     {0}
  ::= {fdClockDstEntry 12}
-- NTCIP uses a range of 0..21600; this expands to the full two-byte
-- range (in practice, I think all systems use 3600 today)

fdClockDstApplied OBJECT-TYPE
  SYNTAX     TruthValue
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "Indicates whether the offset defined by the row is currently applied."
  ::= {fdClockDstEntry 13}
-- Proposed object; might want to consider using TruthValue instead?

fdClockDstStorageType OBJECT-TYPE
  SYNTAX     StorageType
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The type of memory in which this entry is stored."
  ::= {fdClockDstEntry 14}

fdClockDstRowStatus OBJECT-TYPE
  SYNTAX     RowStatus
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The status of this conceptual row. The value of this object shall be
    'notReady' whenever the offset for this row is set to zero (0). Setting
    this object to 'notInService' while the offset defined by this row is
    currently applied will immediately remove the offset (adjusting the local
    time). Setting this object to 'active' shall immediately cause the offset
```

```
    to be applied per the rules defined by this row of the table as if the row
    had been active the entire time. The other objects in this row may not be
    modified while the value of this object is 'active'. An attempt to set any
    other object in this row while this object is 'active' shall result in an
    inconsistentValue error."
  ::= {fdClockDstEntry 15}
-- Standard IETF Row Management


-- ****************************************************************************
-- A.1.4 Conformance Information
-- ****************************************************************************

fdClockMIBCompliance MODULE-COMPLIANCE
  STATUS     current
  DESCRIPTION
    "The conformance statement for the fdClockMIB."
  MODULE     -- this module
    MANDATORY-GROUPS {
                fdUtcClockCapabilitiesGroup,
                fdUtcClockConfigurationGroup,
                fdUtcClockStatusGroup,
                fdUtcClockTimeSourceStatusGroup,
                fdUtcClockValueGroup,
                fdUtcClockDiscontinuityGroup
              }

    GROUP      fdLocalClockConfigurationGroup
    DESCRIPTION
      "Management information for configuring a local clock."

    GROUP      fdLocalClockValueGroup
    DESCRIPTION
      "Management information for determining current local time."

    GROUP      fdDstClockCapabilitiesGroup
    DESCRIPTION
      "Management information that identifies capabilities of the daylight
      saving time feature."

    GROUP      fdDstClockManagementGroup
    DESCRIPTION
      "Management information for controlling access to the daylight saving
      time feature."

    GROUP      fdDstClockConfigurationGroup
    DESCRIPTION
      "Management information for configuring the daylight saving time feature."

    GROUP      fdDstClockActiveGroup
    DESCRIPTION
      "Management information for determining the activity status of a daylight
      saving time entry."

    GROUP      fdDstClockStatusGroup
    DESCRIPTION
      "Management information for determining the enabled status of a daylight
      saving time entry."
  ::= {fdClockCompliances 1}

fdUtcClockCapabilitiesGroup OBJECT-GROUP
  OBJECTS    {
              fdClockResolution,
              fdClockSupportedSources,
              fdClockSupportedTimeKeeping
              }
  STATUS     current
  DESCRIPTION
    "Management information that identifies the capabilities of the UTC Clock."
  REFERENCE "Clause 6.2.2.1"
  ::= {fdClockGroups 1}
```

```
fdUtcClockConfigurationGroup OBJECT-GROUP
  OBJECTS     {
              fdClockRequestedSource,
              fdClockSyncCycle,
              fdClockRequestedTimeKeeping,
              fdClockDiscontinuityMaxAdjustment
              }
  STATUS      current
  DESCRIPTION
    "Management information used to configure the UTC Clock."
  REFERENCE "Clause 6.2.2.2, Clause 6.2.2.3"
  ::= {fdClockGroups 2}

fdUtcClockStatusGroup OBJECT-GROUP
  OBJECTS     {
              fdClockSource,
              fdClockLastSyncTime,
              fdClockLastSyncDate,
              fdClockTimeKeeping
              }
  STATUS      current
  DESCRIPTION
    "Management information that identifies the status of the UTC Clock."
  REFERENCE "Clause 6.2.2.4"
  ::= {fdClockGroups 3}

fdUtcClockTimeSourceStatusGroup OBJECT-GROUP
  OBJECTS     {
              fdClockRequestedSourceStatus,
              fdClockSourceStatus
              }
  STATUS      current
  DESCRIPTION
    "Management information that identifies the status of the UTC Clock time
    source."
  REFERENCE "Clause 6.2.2.5"
  ::= {fdClockGroups 4}

fdUtcClockValueGroup OBJECT-GROUP
  OBJECTS     {
              fdClockUtcTime,
              fdClockUtcDate
              }
  STATUS      current
  DESCRIPTION
    "Management information the represents the current UTC Clock time."
  REFERENCE "Clause 6.2.2.6, Clause 6.2.2.7"
  ::= {fdClockGroups 5}

fdUtcClockDiscontinuityGroup OBJECT-GROUP
  OBJECTS     {
              fdClockDiscontinuitySource,
              fdClockDiscontinuityDelta,
              fdClockDiscontinuityUpTime
              }
  STATUS      current
  DESCRIPTION
    "Management information that provides details regarding time
    discontinuities for the UTC Clock."
  REFERENCE "Clause 6.2.2.8"
  ::= {fdClockGroups 6}

fdLocalClockConfigurationGroup OBJECT-GROUP
  OBJECTS     {
              fdClockLocalStandardTimeZone
              }
  STATUS      current
  DESCRIPTION
    "Management information used to configure the Local Clock."
  REFERENCE "Clause 6.1.2.1, Clause 6.1.2.2"
```

```
  ::= {fdClockGroups 7}

fdLocalClockValueGroup OBJECT-GROUP
  OBJECTS     {
              fdClockLocalTime,
              fdClockLocalDate
              }
  STATUS      current
  DESCRIPTION
    "Management information that provides the current local time."
  REFERENCE "Clause 6.1.2.3"
  ::= {fdClockGroups 8}

fdDstClockCapabilitiesGroup OBJECT-GROUP
  OBJECTS     {
              fdClockDstMaxEntries
              }
  STATUS      current
  DESCRIPTION
    "Management information that describes the capabilities of the daylight
    saving time (a.k.a. summer time) feature of the Local Clock."
  REFERENCE "Clause 6.3.2.1"
  ::= {fdClockGroups 9}

fdDstClockManagementGroup OBJECT-GROUP
  OBJECTS     {
              fdClockDstRowStatus
              }
  STATUS      current
  DESCRIPTION
    "Management information that provides for creation, deletion, and
    management of daylight saving time entries."
  REFERENCE "Clause 6.3.2.2, Clause 6.3.2.3"
  ::= {fdClockGroups 10}

fdDstClockConfigurationGroup OBJECT-GROUP
  OBJECTS     {
              fdClockDstBeginMonth,
              fdClockDstBeginOccurrences,
              fdClockDstBeginDayOfWeek,
              fdClockDstBeginDayOfMonth,
              fdClockDstBeginTime,
              fdClockDstEndMonth,
              fdClockDstEndOccurrences,
              fdClockDstEndDayOfWeek,
              fdClockDstEndDayOfMonth,
              fdClockDstEndTime,
              fdClockDstOffset,
              fdClockDstStorageType
              }
  STATUS      current
  DESCRIPTION
    "Management information used to configure the daylight saving time feature
    of the Local Clock."
  REFERENCE "Clause 6.3.2.2, Clause 6.3.2.3"
  ::= {fdClockGroups 11}

fdDstClockActiveGroup OBJECT-GROUP
  OBJECTS     {
              fdClockDstApplied
              }
  STATUS      current
  DESCRIPTION
    "Management information that identifies whether the daylight saving time
    entry is active."
  REFERENCE "Clause 6.3.2.4"
  ::= {fdClockGroups 12}

fdDstClockStatusGroup OBJECT-GROUP
  OBJECTS     {
              fdClockLocalDstAdjustment
```

```
            }
  STATUS      current
  DESCRIPTION
    "Management information that identifies the status of the daylight saving
    time feature of the Local Clock."
  REFERENCE "Clause 6.3.2.5"
  ::= {fdClockGroups 13}

END
-- ASN1END
```

A.2 Object Group MIB

```
-- ********************************************************************************
-- A.2.1 Object Group Header
-- ********************************************************************************
OBJECT-GROUP-MIB DEFINITIONS ::= BEGIN
IMPORTS

MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY, Integer32, Unsigned32
                                                    FROM SNMPv2-SMI
                                                      -- RFC 2578

TruthValue, StorageType, RowStatus
                                                    FROM SNMPv2-TC
                                                      -- RFC 2579

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

SnmpAdminString
                                                    FROM SNMP-FRAMEWORK-MIB
                                                      -- RFC 3411

ITSDailyTimeStamp, ITSDateStamp, ITSPduErrorStatus, fieldDevice, iso20684p7
                                                    FROM FIELD-DEVICE-TC-MIB
                                                      -- ISO 20684-1 Annex A

;
fdObjectGroupMIB MODULE-IDENTITY
  LAST-UPDATED "202001090107Z"
  ORGANIZATION "ISO TC 204 WG 9"
  CONTACT-INFO
   "name: Kenneth Vaughn
     phone: +1-571-331-5670
     email: kvaughn@trevilon.com
     postal: 6606 FM 1488 RD STE 148-503
             Magnolia, TX 77354
             USA"
  DESCRIPTION
    "This MIB defines a mechanism by which a manager can create a more compact
    encoding of a group of object instances that are requested together. An
    Object Group is configured by defining the desired sequence of object
    instances to be reported as a group. The result is that the device provides a
    new object instance (fdObjectSetData) which represents the values of each
    object instance in the specified order, without the name (OID) field for each
    one and with the values encoded as an OER string rather than
    a BER string, which further compacts the encoding, especially if the
    listed objects are all one-octet integers."

  REVISION "202001090107Z"
  DESCRIPTION
    "Initial revision of the document as proposed for CD ballot."

  ::= {iso20684p7 2}

-- ********************************************************************************
-- A.2.2 Node Definitions
-- ********************************************************************************
```

```
fdObjectGroupConformance OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node containing conformance statements related to the fdObjectGroupMIB,
      as defined in ISO/TS 20684-7."
  ::= {fdObjectGroupMIB 2}

fdObjectGroupCompliances OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node for compliance statements for the fdObjectGroupMIB."
  ::= {fdObjectGroupConformance 1}

fdObjectGroupMIBGroups OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node for group definitions related to fdObjectGroupMIB."
  ::= {fdObjectGroupConformance 2}

fdObjectGroup OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node defining management information related to the field device's Object
     Group."
  ::= {fieldDevice 10}

-- ****************************************************************************
-- A.2.3 Object Group
-- ****************************************************************************

fdObjectGroupsSupportedEncodings OBJECT-TYPE
  SYNTAX     BITS { ber (0), oer (1) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The formats in which the field device is able to encode the object
     group values.      Defined bits are:
     Value | Name | Description
     0     | ber  | Supports Basic Encoding Rules (ISO 8825-1)
     1     | oer  | Supports Octet Encoding Rules (ISO 8825-7)"
  ::= {fdObjectGroup 1}

fdObjectGroupsMaxObjects OBJECT-TYPE
  SYNTAX     Unsigned32
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The maximum number of objects that are supported by each object group."
  ::= {fdObjectGroup 2}

fdObjectGroupsNewValueSupport OBJECT-TYPE
  SYNTAX     INTEGER { none (1), partial (2), full (3) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "Indicates the level of support for the fdObjectGroupNewValue column of
     the table. Values are defined as follows:
     Value   | Description
     none    | The fdObjectGroupNewValue column is not supported
      partial | The fdObjectGroupNew column is supported for some possible
              | configurations of an object group, but limitations apply full
             | fdObjectGroupNewValue is conceptually supported for any
              | allowed configuration of the object group"
  ::= {fdObjectGroup 3}

fdObjectGroupsProcessSupport OBJECT-TYPE
  SYNTAX     BITS { oneStep (1), twoStep (2) }
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
```

```
    "The processes supported by the object group. Defined values are:
       Value    | Description
       oneStep | The one step process (mandatory)
       twoStep | The two step process"
  ::= {fdObjectGroup 4}

fdObjectGroupTable OBJECT-TYPE
  SYNTAX     SEQUENCE OF FdObjectGroupEntry
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "A table that manages the object groups within the field device."
  ::= {fdObjectGroup 5}

fdObjectGroupEntry OBJECT-TYPE
  SYNTAX     FdObjectGroupEntry
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "An object group is a user-defined sequence of object instances that can be
     exchanged as a single entity without having to pass the OID of each object
     instance in the exchange. The fdObjectGroupEntry contains
     information that applies to the entire object group."
  INDEX      {fdObjectGroupOwner, fdObjectGroupName}
  ::= {fdObjectGroupTable 1}

FdObjectGroupEntry ::= SEQUENCE {
                fdObjectGroupOwner           SnmpAdminString,
                fdObjectGroupName            SnmpAdminString,
                fdObjectGroupDescription     SnmpAdminString,
                fdObjectGroupEncoding        INTEGER,
                fdObjectGroupProcess         INTEGER,
                fdObjectGroupRefresh         INTEGER,
                fdObjectGroupLastRefreshDate ITSDateStamp,
                fdObjectGroupLastRefreshTime ITSDailyTimeStamp,
                fdObjectGroupRefreshDuration Unsigned32,
                fdObjectGroupCurrentValue    OCTET STRING,
                fdObjectGroupNewValue        OCTET STRING,
                fdObjectGroupLastError       ITSPduErrorStatus,
                fdObjectGroupLastErrorIndex  Integer32,
                fdObjectGroupClear           TruthValue,
                fdObjectGroupStorageType     StorageType,
                fdObjectGroupRowStatus       RowStatus }

fdObjectGroupOwner OBJECT-TYPE
  SYNTAX     SnmpAdminString (SIZE (0..32))
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "The owner of this entry. The exact semantics of this string are subject to
       the security policy defined by the security administrator."
  REFERENCE  "RFC 2981 mteOwner"
  ::= {fdObjectGroupEntry 1}

fdObjectGroupName OBJECT-TYPE
  SYNTAX     SnmpAdminString (SIZE (1..32))
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "The name of this entry, unique within the scope of the fdObjectGroupOwner."
  REFERENCE  "RFC 2981 mteObjectsName"
  ::= {fdObjectGroupEntry 2}

fdObjectGroupDescription OBJECT-TYPE
  SYNTAX     SnmpAdminString
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "A description of the object group's function and use."
  ::= {fdObjectGroupEntry 3}
```

```
fdObjectGroupEncoding OBJECT-TYPE
  SYNTAX     INTEGER { other (1), ber (2), oer (3) }
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The format in which the field device shall encode this object group.
    Defined values are:
       Value | Description
       other | Encoding format is not defined by this standard. Implementations
             | using 'other' encodings should define a separate object allowing
             | the specification of the desired encoding format and the deter-
             | mination of the specific custom format when selected. A SET to
             | fdObjectGroupEncoding (this object) to 'other' shall return a
             | 'wrongValue' error.
       ber   | The value shall be encoded using BER (ISO 8825-1).
       oer   | The value shall be encoded using OER (ISO 8825-7)."
  ::= {fdObjectGroupEntry 4}

fdObjectGroupProcess OBJECT-TYPE
  SYNTAX     INTEGER { other (1), oneStep (2), twoStep (3) }
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The process to be followed when making requests related to the object set
      Valid values are:
      Value    | Description
      one-step | The field device shall process requests for
               | fdObjectGroupCurrentValue and fdObjectGroupNewValue normally
               | and return the resultant value when done. When in this mode,
               | the value of fdObjectGroupRefresh shall be 'notReady', the
               | value of fdObjectGroupLastRefreshDate shall indicate 01 Jan
               | 2000 and value of fdObjectGroupLastRefreshTime shall indicate
               | 00:00:00.000.
               | An attempt to set the value of fdObjectGroupRefresh shall
               | result in an 'inconsistentValue' result. User's should note
               | that the time for such an operation may take considerably
               | longer than a normal SNMP request since the size of
               | the message may be much greater than normal.
      two-step | To retrieve the current value of the object set, a manager
               | shall first set the value of fdObjectGroupRefresh to
               | 'refresh'. The field device will calculate the value
               | for fdObjectGroupCurrentValue whereupon it will become
               | available for retrieval operations.  The values of
               | fdObjectGroupLastRefreshDate and fdObjectGroupLastRefreshTime
               | shall be updated once the updated value of
               | fdObjectGroupCurrentValue is ready. An attempt to retrieve
               | the value of fdObjectGroupCurrentValue while it is being
               | calculated shall result in a zero-length string."
  ::= {fdObjectGroupEntry 5}

fdObjectGroupRefresh OBJECT-TYPE
  SYNTAX     INTEGER { other (1),
                       ready (2),
                       refresh (3),
                      pending (4),
                       oneStep (5),
                      disabled (6),
                        notReady (7) }
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The state of the refresh operation for fdObjectGroupCurrentValue. Valid
    values are:
       Value    | Description
       other    | A state not defined by this standard
       ready    | The object set is ready for a command to calculate
                | fdObjectGroupCurrentValue
       refresh  | A command to begin calculating fdObjectGroupCurrentValue
       pending  | The value of fdObjectGroupCurrentValue is currently being
                | calculated
       oneStep  | The value of fdObjectGroupProcess is one-step and use of this
```

```
                | is not valid
        disabled | The value of fdObjectGroupEnabled is disabled
        notReady | The value of fdObjectGroupRowStatus is notInService or notReady

      A request to SET this object to any value other than 'refresh' shall
      result in an 'wrongValue' error. Setting this value to 'refresh' when it
      is in the 'ready' mode shall:
        1. Cause the object to immediately transition to the 'pending' state,
        2. Force the value of fdObjectGroupCurrentValue to a zero-length string,
        3. Begin the calculation of a new value for fdObjectGroupCurrentValue

      Once the calculation of fdObjectGroupCurrentValue is complete, the value
      will be stored in that object (replacing the zero-length string) and the
      value of this object shall transition to 'ready'.

      Setting this value to 'refresh' when its value is anything other than
      'ready' shall result in a 'inconsistentValue' error."
  ::= {fdObjectGroupEntry 6}

fdObjectGroupLastRefreshDate OBJECT-TYPE
  SYNTAX     ITSDateStamp
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The UTC date on which the object group was last refreshed when in twoStep
      mode. If the object set is in oneStep mode, the value of this object
     shall be 01 Jan 2000."
  ::= {fdObjectGroupEntry 7}

fdObjectGroupLastRefreshTime OBJECT-TYPE
  SYNTAX     ITSDailyTimeStamp
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The UTC time at which the object group was last refreshed when in twoStep
     mode. If the object set is in oneStep mode, the value of this object
     shall be 00:00:00.000."
  ::= {fdObjectGroupEntry 8}

fdObjectGroupRefreshDuration OBJECT-TYPE
  SYNTAX     Unsigned32
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The estimated duration to perform a request (get or set) on the object
     group. On initial use, this may be based on a simple calculation of the
     number of objects contained within the object group. After initial use,
     it should be based on the actual time required for one or more recent
     requests."
  ::= {fdObjectGroupEntry 9}

fdObjectGroupCurrentValue OBJECT-TYPE
  SYNTAX     OCTET STRING
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "The current value of the SEQUENCE of object instance values referenced by
     this object group. The value is encoded using the rules specified in
     fdObjectGroupEncoding.

      When in oneStep mode, the device shall calculate a new response and return
     resultant value. Given the potential complexity of object groups, an
     implementation may require more time than normal to process this type of
     request. When in twoStep mode, the device shall return the last response
     calculated by the fdObjectGroupRefresh process.
     When calculating the value of this object (in either oneStep or twoStep
      mode), the field device shall act as if it had received a single
     retrieval request containing each referenced object using the same
     security parameters associated with the request (i.e., the parameters in
     the request for this object when in oneStep mode and the parameters in the
     request for fdObjectGroupRefresh when in twoStep mode). If an error occurs
```

```
      during the mimicked retrieval request, fdObjectGroupLastError shall be
      updated to reflect the reported error and the value of this object
      (fdObjectGroupCurrentValue) shall be a zero-length string. Otherwise, if
      there is no error in the retrieval request, the value of this object shall
      be the response encoded according to the chosen encoding rules. If an error
      occurs during the encoding process (e.g., too large for the buffer), the
      error shall be reflected in fdObjectGroupLastError and the value of this
      object shall be a zero-length string. Otherwise, upon successful encoding,
      the encoding shall be returned (oneStep) or stored (twoStep) and the value
      of fdObjectGroupLastError shall be set to 'noError'."
  REFERENCE  "NTCIP 1103 watchBlockValue"
::= {fdObjectGroupEntry 10}

fdObjectGroupNewValue OBJECT-TYPE
  SYNTAX     OCTET STRING
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "This object allows a manager to set new values for the SEQUENCE of object
     instances referenced by this object group. The value is encoded using the
     rules specified in fdObjectGroupEncoding.

     An attempt to set an instance of this object type using a request that
     contains any other object instance shall result in an 'inconsistentValue'
     error. An attempt to set an instance of this object while the
     fdObjectGroupLastError object instance of the same entry has a value of
     'pending' shall result in an 'inconsistentValue' error. An attempt to set
     an instance of this object while the fdObjectGroupRowStatus object
     instance of the same entry has a value of anything other than 'active'
     shall result in an 'inconsistentValue' error.

     Upon receipt of a SET request for this object (and the value of
     fdObjectGroupLastError.x is not already 'pending'), the field device shall
     transition the value of fdObjectGroupLastError.x to 'pending'. The field
     device shall then attempt to decode the value contained within the
     SetRequest-PDU according to the configuration of the object group. If the
     field device encounters an error in decoding the provided values, it shall
     stop processing, set fdObjectGroupLastError.x to 'newValueEncodingError',
     set fdObjectGroupLastErrorIndex.x to zero (0), and return a 'wrongValue'
     error to the manager.

      If the field device successfully decodes the value and it is in oneStep
     mode, the field device shall mimic as if it had received a single SET
     request containing each object referenced within the object group with the
     the values provided in the request using the same security parameters
     associated with the request used to SET this object. Upon completing the
     processing of the mimicked request, the field device shall set
     fdObjectGroupLastError to reflect the error status of the mimicked
     request, set fdObjectGroupLastErrorIndex to the error index of the
     mimicked request and then respond to the original request indicating the
     error status of the mimicked request and the index of one (i.e., by
     definition a set on an object group only contains a single object).

     If the field device successfully decodes the value and it is in twoStep
     mode, the field device shall respond to the request with a 'noError'
     response and then mimic as if it had received a single SET request
     containing each object referenced within the object group with the values
     provided in the request using the same security parameters associated with
     the request used to SET this object. Upon completing the processing of the
     mimicked request, the field device shall set fdObjectGroupLastError to
     reflect the error status of the mimicked request and set
     fdObjectGroupLastErrorIndex to the error index of the mimicked request.

      Retrieving this object shall return the last value SET (with a 'noError'
     response), which may be different than fdObjectGroupCurrentValue. Upon the
     activation of this row of the table, the value of this object is ignored.
     In other words, a set operation is only mimicked when this object is set
     while the row is 'active'."

  ::= {fdObjectGroupEntry 11}
```

```
fdObjectGroupLastError OBJECT-TYPE
  SYNTAX     ITSPduErrorStatus
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "An indication of any error associated with the last operation on
     fdObjectGroupCurrentValue or fdObjectGroupNewValue."
  ::= {fdObjectGroupEntry 12}

fdObjectGroupLastErrorIndex OBJECT-TYPE
  SYNTAX     Integer32
  MAX-ACCESS read-only
  STATUS     current
  DESCRIPTION
    "An indication of the field within the object group that caused the error."
  ::= {fdObjectGroupEntry 13}

fdObjectGroupClear OBJECT-TYPE
  SYNTAX     TruthValue
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "A command to delete all rows in the fdObjectGroupFieldTable that are
     associated with this row of the fdObjectGroupTable. Setting this object to
     true shall delete all associated fields; once the fields are deleted, the
     value of this object shall automatically transition back to false."
  ::= {fdObjectGroupEntry 14}

fdObjectGroupStorageType OBJECT-TYPE
  SYNTAX     StorageType
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The storageType for this conceptual row. Conceptual rows having the value
     'permanent' and supporting SET operations shall allow write access to the
     fdObjectGroupNewValue column; all other columns in the row do not need to
     allow write-access."
  ::= {fdObjectGroupEntry 15}

fdObjectGroupRowStatus OBJECT-TYPE
  SYNTAX     RowStatus
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The status of this conceptual row. This object may be used to create or
     remove rows from this table.
     With the exception of fdObjectGroupNewValue, all columns of this table and
     associated rows of the fdObjectGroupFieldTable shall be appropriately
     configured before the value of this object can transition to active. In
     addition, there shall be at least two rows in the fdObjectGroupFieldTable
     associated with this entry before the value of this object can transition
     to active.

     With the exception of fdObjectGroupNewValue, objects within this row of the
     table can not be modified while the value of this object is active.

     The value of fdObjectGroupNewValue, if it exists, may be modified, but it
     can not be created. An attempt to violate these rules shall result in an
     an inconsistentValue error.

     Creation and modification of any object within this conceptual row and
      corresponding rows of the fdObjectGroupFieldTable is allowed when the value
     of this object is notInService or notReady, within the capabilities of the
     implementation. For example, the implementation might not support newValues
     for all combinations of data."
  REFERENCE  "NTCIP 1103 watchBlockStatus; RFC 2981 mteObjectsEntryStatus"
  ::= {fdObjectGroupEntry 16}

fdObjectGroupFieldTable OBJECT-TYPE
  SYNTAX     SEQUENCE OF FdObjectGroupFieldEntry
  MAX-ACCESS not-accessible
```

```
  STATUS     current
  DESCRIPTION
    "This table defines the sequential list of object instances that are
       contained within the object group."
  ::= {fdObjectGroup 6}

fdObjectGroupFieldEntry OBJECT-TYPE
  SYNTAX     FdObjectGroupFieldEntry
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "An entry specifying one object instance to be contained within the object
     group."

  INDEX      {fdObjectGroupOwner, fdObjectGroupName, fdObjectGroupFieldIndex}
  ::= {fdObjectGroupFieldTable 1}

FdObjectGroupFieldEntry ::= SEQUENCE {
                fdObjectGroupFieldIndex   Unsigned32,
                fdObjectGroupFieldObject  OBJECT IDENTIFIER }

fdObjectGroupFieldIndex OBJECT-TYPE
  SYNTAX     Unsigned32
  MAX-ACCESS not-accessible
  STATUS     current
  DESCRIPTION
    "The logical order of the fdObjectGroupFieldObject within the object group
    SEQUENCE structure where the object with the lowest index occurs first
    within the SEQUENCE followed by each next larger index.

    fdObjectGroupFieldIndex values for an object group do not have to begin with
    one nor are they required to increment by one. For example, an object group
    could be defined with two entries within the fdObjectGroupFieldTable with
    values for fdObjectGroupFieldIndex of 3 and 5. Such an entry would result in
    a sequence as follows:
              SEQUENCE {
                  <object referred to by fdObjectGroupFieldObject.3>,
                  <object referred to by fdObjectGroupFieldObject.5> }"
  REFERENCE  "mteObjectsIndex"
  ::= {fdObjectGroupFieldEntry 1}

fdObjectGroupFieldObject OBJECT-TYPE
  SYNTAX     OBJECT IDENTIFIER
  MAX-ACCESS read-create
  STATUS     current
  DESCRIPTION
    "The OBJECT IDENTIFIER of the object instance to be contained within the
     object group in the position indicated by fdObjectGroupFieldIndex."
  REFERENCE  "NTCIP 1103 watchOID; RFC 2981 mteObjectsID"
  ::= {fdObjectGroupFieldEntry 2}

-- ****************************************************************************
-- A.2.4 Conformance Information
-- ****************************************************************************

fdObjectGroupMIBCompliance MODULE-COMPLIANCE
  STATUS     current
  DESCRIPTION
    "The conformance statement for a device that supports object groups."
  MODULE     -- this module
    MANDATORY-GROUPS {
                fdObjectGroupCapabilitiesGroup,
                fdObjectGroupManagementGroup,
                fdObjectGroupConfigurationGroup,
                fdObjectGroupRetrieveValueGroup,
                fdObjectGroupClearGroup,
                fdObjectGroupErrorGroup
              }

    GROUP      fdObjectGroupRefreshGroup
    DESCRIPTION
```