INTERNATIONAL STANDARD

ISO 20684-1

First edition
2021-04

Intelligent transport systems — Roadside modules SNMP data interface —

Part 1: Overview

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

Partie 1: Vue d'ensemble

Reference number
ISO 20684-1:2021(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 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 the ISO 20684 series may be used with any relevant 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) How conformance is defined in subsequent parts of the ISO 20684 series (Clause 5);

b) Terminology and symbols used throughout the various parts of the ISO 20684 series (Clause 3 and Clause 4);

c) Conventions used throughout the various parts of the ISO 20684 series (Clause 6);

d) The ITS architectural services defined in ISO 14813-1 that are addressed by the ISO 20684 series (Clause 7);

e) The rules used by other parts of the ISO 20684 series in defining the user needs that drive the definition of requirements (Clause 8);

f) The rules used by other parts of the ISO 20684 series in defining requirements and constraints (Clause 9);

g) A set of generic dialogues that are referenced by other parts of the ISO 20684 series (Clause 10);

h) A discussion of security that applies to all devices conforming to the ISO 20684 series (Clause 11);

i) The management information base (MIB) for the features defined by this document (Annex A);

j) A description of the requirements traceability matrix that is provided in each subsequent part of the ISO 20684 series that traces defined requirements to the required design elements (Annex B).

In addition, the MIBs are available electronically at https://standards.iso.org/iso/20684/-1/ed-1/en.

Intelligent transport systems — Roadside modules SNMP data interface —

Part 1: Overview

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.

The ISO 20684 series defines data that can be used when field devices need to exchange information with other external entities (called "managers" in this document, even if they are other field devices). Field devices can be quite complex, necessitating the standardization of many data concepts for exchange. As such, the ISO 20684 series is divided into several individual parts. This document (Part 1) introduces the ISO 20684 series and provides normative content that applies to all subsequent parts.

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.

IETF RFC 2578, *Structure of Management Information Version 2 (SMIv2), April 1999*.

3 Terms and definitions

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

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

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

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

3.1
agent
entity (3.2) that can respond to `get` and `set` requests

3.2
entity
device or "thing" that becomes part of an intelligent transport system

3.3
event
information captured when a *trigger* (3.13) *fires* (3.6) within an *agent* (3.1)

Note 1 to entry: Events are often transmitted in notifications or stored in logs.

3.4
exception
condition that creates an *event* (3.3) that a user can want to store in a *log* (3.7) or transmit in a *notification* (3.9)

3.5
field device
fixed or portable roadside module that includes an *agent* (3.1)

3.6
fire
to start a process when a *trigger* (3.13) value transitions from false to true

3.7
log
registry of *events* (3.3) within an *agent* (3.1) that can be retrieved by a *manager* (3.8)

3.8
manager
entity (3.2) that can generate `get` and `set` requests and/or can receive `report`, `trap`, and/or other `inform` messages

[SOURCE: ISO 15784-2:2015, 4.9 — modified.]

3.9
notification
listing of one or more *events* (3.3) that the *agent* (3.1) can send to one or more *managers* (3.8)

3.10
response time
time from the receipt of the last byte of a Confirmed Class `pduType` to the start of the transmission of the first byte of the response message (when access is allowed by lower layers)

3.11
target
entity (3.1) to which the *field device* (3.5) can need to send `request`s or `notification`s

Note 1 to entry: Field devices often need to send notifications to managers when triggers fire.

Note 2 to entry: Field devices can be configured to request data from other field devices to use in their expression or trigger logic.

Note 3 to entry: Field devices can be configured to control other field devices in response to a trigger firing.

3.12
trap
notification (3.9) sent from an SNMP *agent* (3.1) to a SNMP *manager* (3.8) without an immediately preceding request from the manager or any expectation of an acknowledgement

3.13
trigger
condition that evaluates to a Boolean value

4 Symbols and abbreviated terms

ASN.1 abstract syntax notation one

BER basic encoding rules

IAB Internet Architecture Board

IP	internet protocol
ITS	intelligent transport systems
MIB	management information base
OER	octet encoding rules
OID	object identifier
PDU	protocol data unit
RFC	request for comments
	NOTE Specifically, RFCs published by the Internet Engineering Task Force.
RTM	requirements traceability matrix
SMIv2	structure of management information version 2
SNMP	simple network management protocol
STD	IAB standard
TLS	transport layer security
UDP	user datagram protocol
USM	user-based security model

5 Conformance

Conformance to each part of the ISO 20684 series is defined as per the conformance section of each part, which is written using the structure defined in this clause.

Conformance is driven by defined user needs. Each part of the ISO 20684 series may define user needs and the design to fulfil the user need. User needs are written from the perspective of a manager for the field device.

NOTE Some parts of the ISO 20684 series do not define any user needs, but only provide reusable design elements that can be referenced by other parts of the ISO 20684 series.

Table 1 of each part of the ISO 20684 series (other than this document) identifies the user needs associated with the part and indicates whether they are mandatory or optional for conformance to that part. Each user need is also traced to a set of features. A feature is a high-level, architectural concept that represents a coherent capability of the device that can support multiple user needs.

Table 2 of each part of the ISO 20684 series (other than this document) traces each feature to the requirements for that feature.

Each user need, feature and requirement is identified by name and a reference. The reference can be within the same part, a different part, or a different standard. When references are made to other documents, all details and remaining traceability shall be defined in the referenced document.

Both tables indicate conformance for an item using one of the following conformance codes:

a) M – indicates the item is mandatory when an implementation claims conformance to its parent item.

b) O – indicates the item is optional when an implementation claims conformance to its parent item and if no other parent item makes the item mandatory.

Parent items are defined as follows:

a) The part is the parent item of each user need defined within it.

b) Each user need is a parent item for one or more features as shown by the indentation in Table 1. A user need may trace to multiple features and a feature may trace from multiple user needs.

c) Each feature is a parent item for one or more requirements as shown by the indentation in Table 2. A feature may trace to multiple requirements; a requirement may trace from multiple user needs but typically only traces from one feature.

Note that a feature defined in one part of the ISO 20684 series can have a parent defined in another part. Features should not be defined until at least one user need exists for the feature.

A qualifier may precede a conformance code. In such cases, the qualifier shall be a term followed by a colon. The term shall be defined in Table 3 of each part as a reference to a specific clause in a specific standard. The meaning of this notation is that the conformance code only applies when the referenced clause is supported by an implementation.

EXAMPLE The code "condition:M" means that the indicated row is "mandatory" if the clause referenced by the term "condition" is supported by the implementation.

An option group expression may follow the "O" conformance code. The option group expression is of the form ".<group> (<multiplicity>)", where group shall be a sequential number that groups a number of options together and <multiplicity> shall be a range of integers that indicate the number of options that may be supported by an implementation from the option group.

EXAMPLE The code "O.2 (1..*)" means that the indicated row is optional, but one or more options from option group 2 are to be supported.

The requirements referenced by Table 2 are written as "shall" statements. However, the "shall" only applies if the conformance table indicates that the feature is required.

NOTE This document defines the rules to be followed by subsequent parts of the ISO 20684 series; it does not define any user needs or features itself and therefore does not contain Tables 1-3.

Each requirement specifying a need for a data exchange shall trace to one dialogue and one or more data elements that an implementation claiming conformance to the requirement shall support. The traceability from requirements to dialogues and data elements shall be defined in a requirements traceability matrix (RTM) contained in Annex B of each part of the ISO 20684 series. The RTM may include references to dialogues and data elements defined in other documents; any locally defined dialogues shall be defined in the body of the standard while all locally defined data elements shall be defined in Annex A of the document using a management information base (MIB) conforming to the format defined in IETF RFC 2578. If the implementation supports SNMP, all supported data element instances (i.e. SNMP objects) shall be accessible via any dialogue that meets the requirements of SNMP and the data element definition.

NOTE The dialogues defined in the ISO 20684 series are specified to promote a common interface for testing purposes and are not intended to restrict otherwise allowable requests or notifications.

Annex A of this document defines a set of object identities (i.e. nodes on the OID tree) and textual conventions (i.e. useful data types) that should be imported as needed by other MIB modules contained in other parts of the ISO 20684 series.

As this document does not contain any user needs, the Annex B of this document does not contain an RTM; however, it does provide additional requirements that shall be applicable related to the RTMs contained in other parts of the ISO 20684 series.

6 Conventions

6.1 ASN.1

This document contains MIBs, which are written in the form of ASN.1. This document also contains references to and explanations of ASN.1 data concepts within its text. In all cases, the ASN.1 terms are presented in a fixed width font (e.g. `such as this`) to distinguish these terms from normal English.

6.2 SNMP terminology

Terminology between the different versions of SNMP is slightly different. For the purposes of the ISO 20684 series, the terminology of SNMPv3 is adopted.

7 Architecture

7.1 ITS services

The ISO 20684 series defines mechanisms by which ITS field devices can be monitored, configured and controlled. ITS field devices may be used to support almost any ITS service, defined in ISO 14813-1, with a roadside component.

7.2 Physical view

Figure 1 depicts the physical view of this interface using the graphical conventions defined by the architecture reference for cooperative and intelligent transportation[19] and also documented in ISO 14813-5:2020, Annex B.

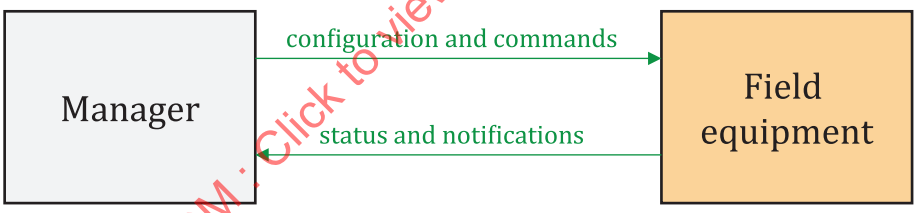

Figure 1 — Physical view of interface

The manager of the field device is shown in grey indicating that it can be any type of physical object, such as a central system, another field device, a maintenance laptop or any other device that supports the defined interface.

The field device is shown in orange, indicating that it is located in the field (e.g. along the roadside). It shall have a connection to the manager and may have any number of connections to other ITS-S or external systems.

The figure indicates two information transfers between these physical objects. The first is the "configuration and commands" information flow from the manager to the field device. The second is the "status and notifications" information flow from the field device to the manager. Both flows are shown in green indicating that authentication is required and both are shown with a single arrowhead indicating a unicast transfer.

Subsequent parts of the ISO 20684 series define needs, requirements and design details for various field device capabilities.

NOTE This document is based on the use of SNMP, which implements a GET/SET paradigm where there is a manager and an agent. However, a single field device entity can act as both a manager (e.g. sending requests to other field devices) and as an agent (e.g. responding to requests from a centre or other field device) simultaneously.

7.3 Communications view

Figure 2 depicts how the ISO 20684 series is intended to relate to other standards using the ITS Station architecture, as defined in ISO 21217.

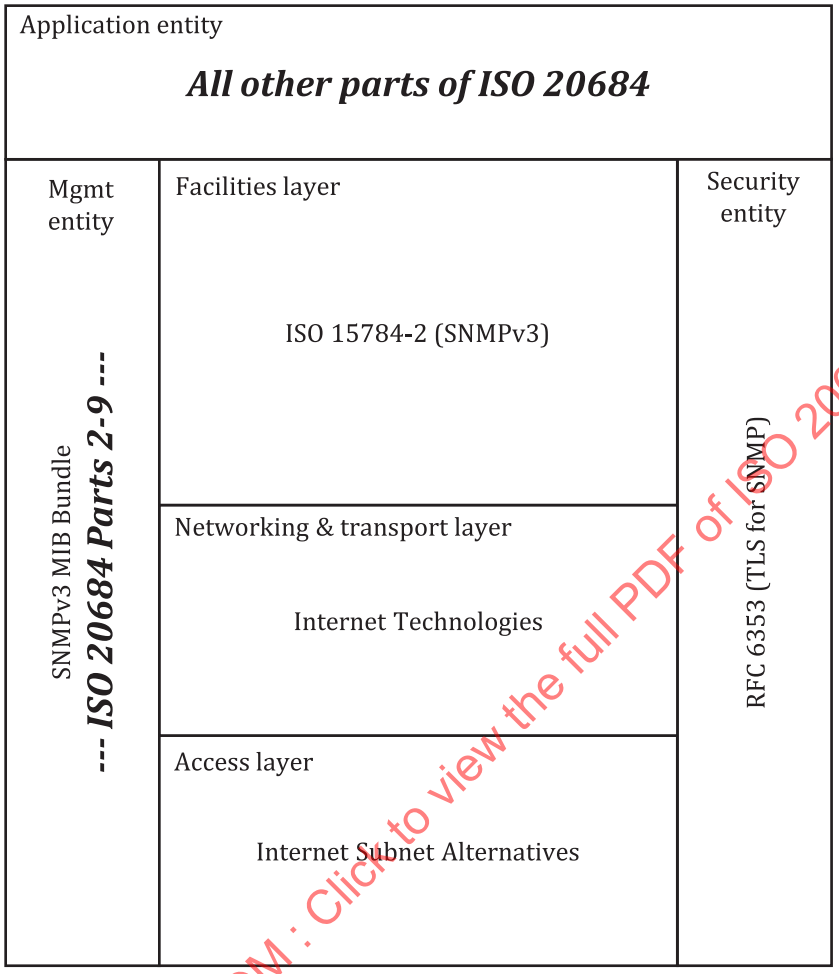

Figure 2 — Typical communications stack

Parts 2-9 of the ISO 20684 series are reserved for features that are a part of the Management Plane of an ITS Station. These standards do not define end-device functionality; rather they define management services that may support end-device functionality. The management layer typically also supports other data to manage the communications stack as defined by a variety of SNMPv3 MIBs.

Parts 10 and above of the ISO 20684 series will address the end-device functionality and are considered part of the definition of ITS-S applications.

The information defined for the Management Plane and ITS-S applications can theoretically be exchanged over any communications stack, but is designed to be exchanged using SNMPv3, as described in ISO 15784-2, with certificate-based security using TLS for SNMP. SNMPv3 is typically exchanged using well-known internet protocols, such as UDP/IP over any number of access layers to media.

8 User needs

Specific user needs are defined in subsequent parts of the ISO 20684 series. Each user need should be written from the perspective of a manager and provide a definition with a justification for the need. The definition of the user need may be followed with an explanation of how the user need is fulfilled using defined features.

9 Requirements

9.1 Features

Within each part of the ISO 20684 series, requirements should be grouped by feature. This will reduce the amount of cross-referencing of requirements. Within a feature, requirements may be further grouped by type of requirement (e.g. data exchange, functional, performance, etc.).

9.2 Identifiable requirements

Each logical requirement shall be written as a "shall" statement and be contained within its own subclause. A requirement may include a list of items, where a letter identifies each item. Multiple "shall" statements should be avoided in a subclause, unless the secondary "shall" statement(s) further refines the initial "shall". The applicability of a subclause to the definition of conformance is separately defined in the conformance section (Clause 5). Placing each requirement in a separate subclause allows easy referencing of requirements within traceability tables. The subclause may contain additional explanatory text to assist the reader in understanding the requirement and/or to justify the requirement.

9.3 Structure of data exchange requirements

Due to the nature of SNMP, data exchange requirements within the ISO 20684 series are generally requirements for the agent to allow a manager to perform operations. Each such data exchange requirement shall follow the following structure, to the extent possible:

<Constraint1> the field device shall allow a manager to <Action> <Target> <Constraint2>

The action is a required component and should typically be something similar to "determine", "retrieve", "configure", "toggle", "reset", "clear", "control", or "confirm".

The target is a required component and shall identify the information that is the recipient of the action.

The constraints are optional. If there are two constraints that apply to the entire requirement, they should generally be placed at opposite ends of the sentence to avoid ambiguities. If there are more than two constraints, great care should be taken and consideration given to restructuring the requirement.

NOTE Ambiguities sometimes arise when two conditional clauses are placed at the end of a sentence; specifically depending on the exact wording, it can be ambiguous whether the second clause is a modifier of the entire sentence or if it only modifies the first conditional clause.

9.4 Agent performance requirements

In the absence of any other specification, the maximum allowed response time for any standardized request shall be 100 ms.

The maximum response time for any non-standard request shall be calculated as follows:

a) Identify the minimum number of standardized request messages that contain all of the objects included in the request for which the calculation is being made.

b) The maximum response time for the non-standard request shall be the sum of the maximum response times for all of the standardized requests identified in Step a).

10 Dialogues

10.1 General dialogue rules

10.1.1 Manager initiated

Unless otherwise stated, all dialogues are initiated by the manager and the logic used by the manager to initiate a dialogue is outside the scope of the ISO 20684 series.

10.1.2 Generic and custom dialogues

This document defines several generic dialogues that may be referenced by subsequent parts of the ISO 20684 series. In addition, subsequent parts of the ISO 20684 series may define additional dialogues.

10.2 Generic dialogues

10.2.1 Get elemental data

This dialogue shall be initiated by the manager using logic that is implementation specific. The dialogue shall be as follows:

a) The manager shall send a `GetRequest-PDU` for the data elements shown for the dialogue in the RTM.

b) The device shall respond as per the rules of SNMP.

10.2.2 Set elemental data

This dialogue shall be initiated by the manager using logic that is implementation specific. The dialogue shall be as follows:

a) The manager shall send a `SetRequest-PDU` for the data elements shown for the dialogue in the RTM using values determined by the manager.

b) The device shall respond as per the rules of SNMP.

10.2.3 Get tabular data

This dialogue shall be initiated by the manager using logic that is implementation specific. The dialogue shall be as follows:

a) The manager shall know which rows of the table to retrieve.

b) The manager shall send one `GetRequest-PDU` for the data elements shown for the dialogue in the RTM for each row for which information is desired.

c) The device shall respond to each request as per the rules of SNMP.

10.2.4 Set tabular data

This dialogue shall be initiated by the manager using logic that is implementation specific. The dialogue shall be as follows:

a) The manager shall know which rows of the table to set.

b) The manager shall send one `SetRequest-PDU` for the data elements shown for the dialogue in the RTM for each row for which information is desired to be set.

c) The device shall respond to each request as per the rules of SNMP.

10.2.5 Get data column

This dialogue shall be initiated by the manager using logic that is implementation specific. The dialogue shall be as follows:

a) The manager shall set the value of *x* to zero (0).

b) The manager shall send a GetNextRequest-PDU for the "*x*" instance of the data element shown for the dialogue in the RTM.

c) The device shall respond to the request as per the rules of SNMP.

d) If the response indicates 'noError' and the response contains an instance of the data element requested, the manager shall set the value of x to the index of the object instance that was received.

e) The manager shall repeat Steps b) through d) until it receives an error response, or the response does not contain an instance of the data element requested.

10.2.6 Get data from dynamic table entry

This dialogue shall be initiated by the manager using logic that is implementation specific. All messages sent by the manager in this dialogue shall be sent to the agent. The dialogue shall be as follows:

a) The manager shall know which row of the table to retrieve (referred to as the "subject row"). All operations performed in this dialogue shall be performed on object instances within the subject row.

b) The manager shall send one `GetRequest-PDU` containing the objects shown for the dialogue in the RTM.

c) The device shall respond to the request as per the rules of SNMP.

d) The manager shall consider the value of the "RowStatus" object before interpreting the data from the other columns.

NOTE If the "RowStatus" object indicates a value of "notInService", the data from the other columns do not necessarily represent current conditions.

10.2.7 Get row status of dynamic table entry

This dialogue shall be initiated by the manager using logic that is implementation specific. All messages sent by the manager in this dialogue shall be sent to the agent. The dialogue shall be as follows:

a) The manager shall know which row of the table to retrieve (referred to as the "subject row"). All operations performed in this dialogue shall be performed on object instances within the subject row.

b) The manager shall send one `GetRequest-PDU` containing the "RowStatus" object.

c) The device shall respond to the request as per the rules of SNMP.

d) The manager shall consider the status of the "RowStatus" object.

10.2.8 Configure entry of a dynamic table

This dialogue shall be initiated by the manager using logic that is implementation specific. All messages sent by the manager in this dialogue shall be sent to the agent. The dialogue shall be as follows:

a) The manager shall know which row of the table to set (referred to as the "subject row"). All operations performed in this dialogue shall be performed on object instances within the subject row.

b) The manager shall send one `GetRequest-PDU` containing the "RowStatus" object.

c) The device shall respond to the request per the rules of SNMP.

d) If the response indicates that the row is in the "active" state, the manager shall send one `SetRequest-PDU` containing the "RowStatus" object with its value set to "notInService"; the device shall respond to the request as per the rules of SNMP.

e) If the response from Step "c" indicates that the row does not exist, the manager shall send one `SetRequest-PDU` containing the "RowStatus" object with its value set to "createAndWait"; the device shall respond to the request as per the rules of SNMP.

f) The manager shall send one `SetRequest-PDU` containing the objects shown for the dialog in the RTM, with the exception of the "RowStatus" object, which shall be omitted from the request.

g) The device shall respond to the request as per the rules of SNMP.

h) The manager shall send one `SetRequest-PDU` containing the "RowStatus" object with its value set to "active".

i) The device shall respond to the request as per the rules of SNMP.

10.2.9 Toggle active status of a dynamic table entry

This dialogue shall be initiated by the manager using logic that is implementation specific. All messages sent by the manager in this dialogue shall be sent to the agent. The dialogue shall be as follows:

a) The manager shall know which row of the table to toggle (referred to as the "subject row"). All operations performed in this dialogue shall be performed on object instances within the subject row.

b) The manager shall send one `GetRequest-PDU` containing the "RowStatus" object.

c) The device shall respond to the request as per the rules of SNMP.

d) If the response indicates that the row does not exist or if the response indicates that the row status is "notReady", this dialogue shall exit unsuccessfully.

e) If the response indicates that the row is "active", the manager shall send one `SetRequest-PDU` containing the "RowStatus" object with its value set to "notInService".

f) If the response indicates that the row is "notInService", the manager shall send one `SetRequest-PDU` containing the "RowStatus" object with its value set to "active".

g) The device shall respond to the request as per the rules of SNMP.

10.2.10 Delete entry from a dynamic table

This dialogue shall be initiated by the manager using logic that is implementation specific. All messages sent by the manager in this dialogue shall be sent to the agent. The dialogue shall be as follows:

a) The manager shall know which row of the table to delete (referred to as the "subject row"). All operations performed in this dialogue shall be performed on object instances within the subject row.

b) The manager shall send one `SetRequest-PDU` containing the "RowStatus" object with its value set to "destroy".

c) The device shall respond to the request as per the rules of SNMP.

11 Security

11.1 Vulnerabilities

There are data elements defined in the ISO 20684 series with a `MAX-ACCESS` clause of `read-write` and/or `read-create`. These and other data elements are sensitive and should 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. Depending on the type of field device, the effect can result in an annoyance, impaired traffic operations, or even safety-of-life issues. Specific vulnerabilities shall be highlighted within each part of the ISO 20684 series.

11.2 Authentication and access control

SNMP versions prior to SNMPv3 with (D)TLS as defined in RFC 6353 do not provide adequate authentication and access control. Even if the network itself is secure (for example, by using IPsec or TLS), there is inadequate control as to who on the secure network is allowed to access (read/change/create/delete) the data defined by the ISO 20684 series.

Previous security models (including SNMPv1, SNMPv2c, and the SNMPv3 user-based security model [USM]) derive the `securityName` and `securityLevel` from the SNMP message received, even when the message is received over a secure transport. Access control decisions are therefore made based on the contents of the SNMP message, without proper authorization of the request. The result is that any authorized network connection is able to access any data in the device by simply providing a valid (but unauthenticated) `securityName`. Once one `securityName` is known, the features of SNMP can then be used to potentially discover other `securityName`s and to potentially open a major vulnerability, especially on the oldest versions of SNMP.

11.3 Encryption

Some data exchanged by ITS field devices is sensitive. In particular, data elements that provide context names, target tags, and similar information can be used to attack a system. All sensitive data exchanged by an entity should be encrypted to ensure continued secure operation of the field device.

11.4 Security recommendation

Deployment of SNMP versions prior to SNMPv3 with (D)TLS as defined in RFC 6353 is not recommended.

Instead, it is recommended to deploy SNMPv3 with (D)TLS (or later security solutions) and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity is properly configured to only give access to data to those users who have legitimate needs.

Annex A
(normative)

Management information base (MIB)

This annex provides definitions that it is useful to import into other MIB modules of the ISO 20684 series. The definitions are contained within a MIB, which conforms to the format defined in IETF RFC 2578. Near the start of the MIB, an imports clause is used to identify elements defined in other MIB modules that are normatively required for the complete definition of the current MIB module. Per the rules of IETF RFC 2578, the source for each group of imported elements is identified by the respective module name (e.g., "SNMPv2-SMI"). This document supplements the module name with the document identifier (e.g., "RFC 2578") that contains the formal definition of the imported MIB module. The formal references to each of these normatively referenced documents are provided in Clause 2 of this document.

```
-- ASN1START
-- A.1
FIELD-DEVICE-TC-MIB { iso(1) standard(0) 20684 part1(1) version1(1) annexA(1) }
DEFINITIONS ::= BEGIN

IMPORTS
  MODULE-IDENTITY, OBJECT-IDENTITY, Integer32
                                                FROM SNMPv2-SMI
                                                  -- RFC 2578
  TEXTUAL-CONVENTION
                                                FROM SNMPv2-TC;
                                                  -- RFC 2579


fdTCMib MODULE-IDENTITY
  LAST-UPDATED "201701030029Z"
  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
    "The MIB that defines textual conventions that apply to generic field
     devices. It is expected that the Textual Conventions defined in this module
     will be imported by a variety of other MIBs.

     Copyright (C) International Organization for Standardization (ISO) (2017).
     This version of this MIB module is part of ISO 20684-1; see ISO 20684-1
     itself for full legal notices."
  REVISION     "201909090324Z"
  DESCRIPTION
    "Added identifiers for fdVms and iso20684p10."
  REVISION     "201701030029Z"
  DESCRIPTION
    "Initial version of the MIB module as distributed for NP Ballot."
  ::= { iso20684p1 1}

-- ****************************************************************
-- Node Definitions
-- ****************************************************************
its OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for defining management information used by the intelligent
    transport system (ITS) industry."
```

```
  ::= { joint-iso-ccitt 28 }

fieldDevice OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for defining management information to manage field devices
    within the intelligent transport system (ITS) industry."
  ::= { its 3 }

fdVms OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for defining management information to manage variable message
    signs within the intelligent transport system (ITS) industry."
  ::= { its 4 }

iso20684 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by the ISO 20684
    series that does not need to be recorded under the joint-iso-ccitt its node."
  ::= { iso 20684 }

iso20684p1 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by Part 1 of the
    ISO 20684 series that does not need to be recorded under the joint-iso-ccitt
    ITS node. Part 1 is primarily focused on defining textual conventions that
    can be used by other parts of this series."
  ::= { iso20684 1 }

iso20684p2 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by Part 2 of the
    ISO 20684 series that does not need to be recorded under the joint-iso-ccitt
    ITS node. Part 2 defines basic field device data related to the controller,
    cabinet, and general-purpose input/outputs."
  ::= { iso20684 2 }

iso20684p3 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by Part 3 of the
    ISO 20684 series that does not need to be recorded under the joint-iso-ccitt
    ITS node. Part 3 defines triggers that can initiate actions."
  ::= { iso20684 3 }

iso20684p4 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by Part 4 of the
    ISO 20684 series that does not need to be recorded under the joint-iso-ccitt
    ITS node. Part 4 defines data related to handling exception conditions within
    a device by sending notices to a manager."
  ::= { iso20684 4 }

iso20684p5 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by Part 5 of the
    ISO 20684 series that does not need to be recorded under the joint-iso-ccitt
    ITS node. Part 5 defines data related to handling exception conditions within
    a device by recording the exception in a log."
  ::= { iso20684 5 }

iso20684p6 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by Part 6 of the
```

```
    ISO 20684 series that does not need to be recorded under the joint-iso-ccitt
    ITS node. Part 6 defines data related to handling exception conditions within
    a device by initiating an action based on the exception."
  ::= { iso20684 6 }

iso20684p7 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by Part 7 of the
    ISO 20684 series that does not need to be recorded under the joint-iso-ccitt
    ITS node. Part 7 defines data that is used by other features, such as clock
    and object group data."
  ::= { iso20684 7 }

iso20684p10 OBJECT-IDENTITY
  STATUS     current
  DESCRIPTION
    "A node used for identifying management information defined by Part 10 of the
    ISO 20684 series that does not need to be recorded under the joint-iso-ccitt
    ITS node. Part 10 defines data related to variable message signs."
  ::= { iso20684 10 }


-- ********************************************************************
-- A.2 Useful integers
-- ********************************************************************
-- A.2.1
ITSBitmap ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A bitmapped sequence of Boolean values where 0 means off or false and 1
     means on or true. Bits are numbered sequentially starting with Bit 0
     representing the highest order bit of the first byte, Bit 7 representing
     the lowest order bit of the first byte, Bit 8 representing the highest order
     bit of the second byte, etc. The length of this shall only be as long as
     necessary to transmit the required number of bits with meaning. Any
     trailing bits shall have the value of zero (0)."
  SYNTAX     OCTET STRING

-- A.2.2
ITSBitmap8 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A bitmapped sequence of boolean values where 0 means off or false and 1
     means on or true. Bits are numbered sequentially starting with Bit 0
     representing the highest order bit of the byte, Bit 7 representing the
     lowest order bit of the byte. The length shall be one byte (8 bits). Any
     trailing bits shall have the value of zero (0)."
  SYNTAX     OCTET STRING (SIZE(1))

-- A.2.3
ITSBitmap16 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A bitmapped sequence of boolean values where 0 means off or false and 1
     means on or true. Bits are numbered sequentially starting with Bit 0
     representing the highest order bit of the first byte, Bit 7 representing
     the lowest order bit of the first byte, Bit 8 representing the highest order
     bit of the second byte, etc. The length shall be two bytes (16 bits). Any
     trailing bits shall have the value of zero (0)."
  SYNTAX     OCTET STRING (SIZE(2))

-- A.2.4
ITSBitmap32 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A bitmapped sequence of boolean values where 0 means off or false and 1
     means on or true. Bits are numbered sequentially starting with Bit 0
     representing the highest order bit of the first byte, Bit 7 representing
     the lowest order bit of the first byte, Bit 8 representing the highest order
     bit of the second byte, etc. The length shall be four bytes (32 bits). Any
```

```
     trailing bits shall have the value of zero (0)."
  SYNTAX     OCTET STRING (SIZE(4))

-- A.2.5
ITSDailyTimeStamp ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A specific time of day represented as nominal milliseconds from midnight,
     in the local time zone unless otherwise specified. (It is nominal in the
     sense that a daylight savings adjustment may increase or decrease this
     value from actual milliseconds from midnight.)"
  SYNTAX     Integer32 (0..86401001)

-- A.2.6
ITSInteger8 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A signed integer that can be presented as a one-byte value in OER."
  SYNTAX     Integer32 (-128..127)

-- A.2.7
ITSInteger16 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A signed integer that can be presented as a two-byte value in OER."
  SYNTAX     Integer32 (-32768..32767)

-- A.2.8
ITSPositive8 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A positive integer that can be presented as a one-byte value in OER."
  SYNTAX     Integer32 (1..255)

-- A.2.9
ITSPositive16 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A positive integer that can be presented as a two-byte value in OER."
  SYNTAX     Integer32 (1..65535)

-- A.2.10
ITSUnsigned8 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "An unsigned integer that can be presented as a one-byte value in OER."
  SYNTAX     Integer32 (0..255)

-- A.2.11
ITSUnsigned16 ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "An unsigned integer that can be presented as a two-byte value in OER."
  SYNTAX     Integer32 (0..65535)

-- A.2.12
ITSDayOfMonth ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "The day of the month."
  SYNTAX     Integer32 (1..31)

-- A.2.13
ITSOerString ::= TEXTUAL-CONVENTION
  STATUS     current
  DESCRIPTION
    "A string of octets that represents an ASN.1 value encoded using the
     octet encoding rules (OER)."
  SYNTAX     OCTET STRING

-- A.2.14
```

```
ITSPduErrorStatus ::= TEXTUAL-CONVENTION
  STATUS      current
  DESCRIPTION
    "Reasons for failures in an attempt to perform a management request.

    The first group of errors, numbered less than 0, are related to problems in
    sending the request. The existence of a particular error code here does not
    imply that all implementations or uses of this textual convention are
    capable of sensing that error and returning that code.

    The second group, numbered greater than 0, are copied directly from SNMP
    protocol operations and are intended to carry exactly the meanings
    defined for the protocol as returned in an SNMP response.

    localResourceLack       some local resource such as memory lacking or
                            mteResourceSampleInstanceMaximum exceeded
    badDestination          unrecognized domain name or otherwise invalid
                            destination address
    destinationUnreachable  can't get to destination address
    noResponse              no response to SNMP request
    badType                 the data syntax of a retrieved object as not as
                            expected
    sampleOverrun           another sample attempt occurred before the
                            previous one completed"
  REFERENCE
    "The initial version of this textual convention is identical to the
    FailureReason textual convention defined in RFC 2981 and is an extension
    to the SnmpPduErrorStatus textual convention defined in RFC 3231."
  SYNTAX      INTEGER { localResourceLack(-1),
                        badDestination(-2),
                        destinationUnreachable(-3),
                        noResponse(-4),
                        badType(-5),
                        sampleOverrun(-6),

                        noError(0),

                        tooBig(1),
                        noSuchName(2),
                        badValue(3),
                        readOnly(4),
                        genErr(5),
                        noAccess(6),
                        wrongType(7),
                        wrongLength(8),
                        wrongEncoding(9),
                        wrongValue(10),
                        noCreation(11),
                        inconsistentValue(12),
                        resourceUnavailable(13),
                        commitFailed(14),
                        undoFailed(15),
                        authorizationError(16),
                        notWritable(17),
                        inconsistentName(18) }

-- ************************************************************
-- A.3 Useful structures
-- ************************************************************
-- A.3.1
ITSDateStamp ::= TEXTUAL-CONVENTION
  STATUS      current
  DESCRIPTION
    "A Gregorian calendar date expressed using the following OER encoded
     sequence:
     SEQUENCE {
     year    ITSInteger16,
     month   ITSMonth,
     date    ITSDayOfMonth
     }
     An attempt to set a date to an invalid date (e.g. February 30) shall
```