

DIN EN 60870-6-802

ICS 33.200

Ersatz für
DIN EN 60870-6-802:2003-02
und
DIN EN 60870-6-802/A1:2006-05
Siehe Anwendungsbeginn

**Fernwirkeinrichtungen und -systeme –
Teil 6-802: Fernwirkprotokolle, die mit ISO-Normen und
ITU-T-Empfehlungen kompatibel sind –
TASE.2-Objektmodelle
(IEC 60870-6-802:2014);
Englische Fassung EN 60870-6-802:2014**

Telecontrol equipment and systems –
Part 6-802: Telecontrol protocols compatible with ISO standards and ITU-T
recommendations –
TASE.2 Object models
(IEC 60870-6-802:2014);
English version EN 60870-6-802:2014

Matériels et systèmes de téléconduite –
Partie 6-802: Protocoles de téléconduite compatibles avec les normes ISO et les
recommandations de l'UIT-T –
Modèles d'objets TASE.2
(CEI 60870-6-802:2014);
Version anglaise EN 60870-6-802:2014

Gesamtumfang 98 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2014-08-19 angenommene Europäische Norm als DIN-Norm ist 2015-04-01.

Für DIN EN 60870-6-802:2003-02 und DIN EN 60870-6-802/A1:2006-05 besteht eine Übergangsfrist bis 2017-08-19.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 60870-6-802:2013-04.

Für dieses Dokument ist das nationale Arbeitsgremium K 952 „Netzeleittechnik“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom TC 57 „Power systems management and associated information exchange“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieser Publikation bis zu dem Datum (stability date) unverändert bleiben soll, das auf der IEC-Website unter „<http://webstore.iec.ch>“ zu dieser Publikation angegeben ist. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees die Publikation

- bestätigt,
- zurückgezogen,
- durch eine Folgeausgabe ersetzt oder
- geändert.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf eine Norm ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils neueste gültige Ausgabe der in Bezug genommenen Norm.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe der Norm.

Der Zusammenhang der zitierten Normen mit den entsprechenden Deutschen Normen ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für die vorliegende Norm erfüllt.

Da sich die Benutzer der vorliegenden Norm der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der EN 60870-6-802 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA der vorliegenden Norm hinzugefügt. Für die meisten der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Diese Norm steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Änderungen

Gegenüber DIN EN 60870-6-802:2003-02 und DIN EN 60870-6-802/A1:2006-05 wurden folgende Änderungen vorgenommen:

- a) Pflege der Abwärtskompatibilität;
- b) Berücksichtigung der spezifischen Kommentare von IEC/TC 57 WG 19;
- c) Einarbeitung der Empfehlung, die Semantik von IEC 60870-6 TASE.2 und IEC 61970 an die TC 57 Referenzarchitektur anzugleichen;
- d) Aktualisierung der Sicherheitsbezüge;

- e) Einarbeitung der vorherigen Verbesserungsvorschläge;
- f) Einarbeitung der nationalen Kommentierungen zu 57/980/DC;
- g) Entfernung der Objekte und Dienste, die typischerweise nicht innerhalb eines informativen Annex verwendet werden;
- h) für die verbleibende Normung relevante Sektionen beinhalten Objekte und Dienste, die zur Unterstützung der Blogs 1,2,3,4 und 5 benötigt werden;
- i) Einarbeitung der Änderungen der bisherigen IEC 60870-6-802 am1 direkt in das Dokument;
- j) Bereitstellung einer Integrations-Spezifikation für IEC 60870-6-802 normative Objekte und IEC 61970;
- k) Erhaltung der Konsistenz mit den Korrekturen, die im Rahmen der Revisionen mit IEC 60870-6-503 durchgeführt wurden.

Frühere Ausgaben

DIN EN 60870-6-802: 1998-08, 2003-02

DIN EN 60870-6-802/A1: 2006-05

Nationaler Anhang NA (informativ)

1 Anwendungsbereich

Dieser Teil der IEC 60870 spezifiziert eine Methode zum Austausch zeitkritischer Leitstellen-Daten über Weitverkehrsnetzwerke (en: Wide Area Network, WAN) und lokale Netzwerke (en: Local Networks, LAN), die einen vollständig mit ISO übereinstimmenden Protokoll-„Stack“ verwenden. Dieser Teil enthält Bestimmungen, die zentralisierte und verteilte Architekturen gleichermaßen unterstützen. Die Norm beinhaltet den Austausch von Echtzeit-Daten-Anzeigen, Überwachung, Zeitreihendaten, Planung und Kosten-informationssysteme, Fernüberwachung der Programme und Ereignisbenachrichtigungen.

– Leerseite –

English Version

**Telecontrol equipment and systems - Part 6-802: Telecontrol
protocols compatible with ISO standards and ITU-T
recommendations - TASE.2 Object models
(IEC 60870-6-802:2014)**

Matériels et systèmes de téléconduite - Partie 6-802:
Protocoles de téléconduite compatibles avec les normes
ISO et les recommandations de l'UIT-T - Modèles d'objets
TASE.2
(CEI 60870-6-802:2014)

Fernwirkrichtungen und -systeme - Teil 6-802:
Fernwirkprotokolle, die mit ISO-Normen und ITU-T-
Empfehlungen kompatibel sind - TASE.2-Objektmodelle
(IEC 60870-6-802:2014)

This European Standard was approved by CENELEC on 2014-08-19. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.



European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Foreword

The text of document 57/1455/FDIS, future edition 3 of IEC 60870-6-802, prepared by IEC/TC 57 "Power systems management and associated information exchange" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 60870-6-802:2014.

The following dates are fixed:

- | | | |
|--|-------|------------|
| – latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement | (dop) | 2015-05-19 |
| – latest date by which the national standards conflicting with the document have to be withdrawn | (dow) | 2017-08-19 |

This document supersedes EN 60870-6-802:2002.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CENELEC [and/or CEN] shall not be held responsible for identifying any or all such patent rights.

Endorsement notice

The text of the International Standard IEC 60870-6-802:2014 was approved by CENELEC as a European Standard without any modification.

CONTENTS

INTRODUCTION.....	5
1 Scope.....	6
2 Normative references	6
3 Terms and definitions	6
4 Abbreviations	6
5 Object models	6
5.1 General.....	6
5.2 Supervisory Control and Data Acquisition	7
5.2.1 General	7
5.2.2 IndicationPoint object.....	7
5.2.3 ControlPoint Object.....	10
5.2.4 Protection Equipment Event Object Model	12
5.3 Device Outage Object	15
5.4 InformationBuffer Object.....	18
6 MMS Types for Object Exchange	18
6.1 General.....	18
6.2 Supervisory Control and Data Acquisition Types	19
6.2.1 IndicationPoint Type Descriptions.....	19
6.2.2 ControlPoint Type Descriptions	22
6.2.3 Protection Equipment Type Descriptions.....	22
6.3 Device Outage Type Descriptions.....	23
6.4 InformationBuffer Type Descriptions	25
7 Mapping of Object Models to MMS Types.....	25
7.1 Supervisory Control and Data Mapping	25
7.1.1 Indication Object Mapping	25
7.1.2 ControlPoint Object Mapping	28
7.1.3 Protection Event Mapping	29
7.2 Device Outage Mapping	32
7.3 Information Buffer Mapping	34
8 Use of Supervisory Control Objects.....	35
8.1 General.....	35
8.2 Use of IndicationPoint Model	35
8.3 Use of ControlPoint Model.....	36
9 Conformance.....	36
Annex A (informative) TASE.2 (2002) Additional Object Models	38
A.1 General.....	38
A.2 Transfer Accounts	38
A.3 Power Plant Objects.....	45
A.3.1 General	45
A.3.2 Availability Report Object	45
A.3.3 Real Time Status Object.....	49
A.3.4 Forecast Schedule Object	52
A.4 General Data Report Object	55
A.4.1 General	55

A.4.2	General Data Request Object.....	55
A.4.3	General Data Response Object.....	58
Annex B (informative)	TASE.2 (2002) Additional MMS Object Types	60
B.1	General	60
B.2	Transfer Account Types	60
B.3	Power Plant Type Descriptions.....	62
B.4	Power System Dynamics.....	66
B.4.1	General	66
B.4.2	Matrix Data Types.....	66
B.5	GeneralDataReport Type Descriptions.....	67
B.6	GeneralDataResponse Type Descriptions.....	67
Annex C (informative)	TASE.2 (2002) Mapping of Objects to MMS Types	69
C.1	General	69
C.2	Transfer Accounts Mapping.....	69
C.2.1	TransferAccount Mapping	69
C.2.2	TransmissionSegment Mapping.....	73
C.2.3	ProfileValue Mapping	76
C.2.4	AccountRequest Mapping.....	76
C.3	Power Plant Mapping	77
C.3.1	Availability Report Mapping.....	77
C.3.2	Real Time Status Mapping	80
C.3.3	Forecast Mapping	82
C.3.4	Curve Mapping	83
C.4	General Data Report Mapping	84
C.4.1	General Data Request Mapping.....	84
C.4.2	General Data Response Mapping.....	88
Annex D (informative)	Transfer account examples.....	90
Annex ZA (normative)	Normative references to international publications with their corresponding European publications.....	94

INTRODUCTION

The primary purpose of Telecontrol Application Service Element (TASE.2) is to transfer data between control systems and to initiate control actions. Data is represented by object instances. This part of IEC 60870 proposes object models from which to define object instances. The object models represent objects for transfer. The local system may not maintain a copy of every attribute of an object instance.

The object models presented herein are specific to "control centre" or "utility" operations and applications; objects required to implement the TASE.2 protocol and services are found in IEC 60870-6-503. Since needs will vary, the object models presented here provide only a base; extensions or additional models may be necessary for two systems to exchange data not defined within this standard.

It is by definition that the attribute values (i.e. data) are managed by the owner (i.e. source) of an object instance. The method of acquiring the values is implementation dependent; therefore accuracy is a local matter.

The notation of the object modelling used for the objects specified in Clause 5 is defined in IEC 60870-6-503. This part of IEC 60870 is based on the TASE.2 services and protocol. To understand the modelling and semantics of this standard, some basic knowledge of IEC 60870-6-503 would be advisable.

The notation of the object modelling used for the objects specified in Clause B.2 is defined in IEC 60870-6-503. This part of IEC 60870-6 is based on the TASE.2 services and protocol. To understand the modelling and semantics of this part of IEC 60870-6, some basic knowledge of IEC 60870-6-503 would be advisable.

Clause 5 describes the control centre-specific object models and their application. They are intended to provide information to explain the function of the data.

Clause 6 defines a set of MMS type descriptions for use in exchanging the values of instances of the defined object models. It is important to note that not all attributes of the object models are mapped to types. Some attributes are described simply to define the processing required by the owner of the data and are never exchanged between control centres. Other attributes are used to determine the specific types of MMS variables used for the mapping, and therefore do not appear as exchanged values themselves. A single object model may also be mapped onto several distinct MMS variables, based on the type of access and the TASE.2 services required.

Clause 7 describes the mapping of instances of each object type MMS variables and named variable lists for implementing the exchange.

Clause 8 describes device-specific codes and semantics to be used with the general objects.

Clause 9 is the standards conformance table.

An informative Annex A is included which describes some typical interchange scheduling scenarios, along with the use of TASE.2 objects to implement the schedule exchange.

TELECONTROL EQUIPMENT AND SYSTEMS –

Part 6-802: Telecontrol protocols compatible with ISO standards and ITU-T recommendations – TASE.2 Object models

1 Scope

This part of IEC 60870 specifies a method of exchanging time-critical control centre data through wide-area and local-area networks using a full ISO compliant protocol stack. It contains provisions for supporting both centralized and distributed architectures. The standard includes the exchange of real-time data indications, control operations, time series data, scheduling and accounting information, remote program control and event notification.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 60870-5-101:2003, *Telecontrol equipment and systems – Part 5-101: Transmission protocols – Companion standard for basic telecontrol tasks*

IEC 60870-6-503:2014, *Telecontrol equipment and systems – Part 6-503: Telecontrol protocols compatible with ISO standards and ITU-T recommendations – TASE.2 Services and protocol*

ISO 9506-1:2003, *Industrial automation systems – Manufacturing Message Specification – Part 1: Service definition*

ISO 9506-2:2003, *Industrial automation systems – Manufacturing Message Specification – Part 2: Protocol specification*

3 Terms and definitions

For the purposes of this part of IEC 60870, the terms and definitions in the above referenced standards apply.

4 Abbreviations

For the purposes of this part of IEC 60870, all the abbreviations defined in the above referenced standards apply.

5 Object models

5.1 General

Object models are required for various functions within a system. Clause 5 delineates abstract object models based on functionality. Object models within one functional area may be used in another functional area.

5.2 Supervisory Control and Data Acquisition

5.2.1 General

The object models in this clause are derived from the historical perspective of Supervisory Control and Data Acquisition (SCADA) systems. This subclause presents the context within which the object models are defined.

Fundamental to SCADA systems are two key functions: control and indication. The control function is associated with the output of data whereas the indication function is associated with the input of data. A more recent concept that is finding usage is the control and indication function where data output may also be input (i.e. bi-directional).

The previous identified functions within SCADA systems are mapped to point equipment (point). The primary attribute of a point is the data value. SCADA systems define three types of data for points: analog, digital and state.

The association of one or more points together is used to represent devices. For example, a breaker device may be represented by a control point and an indication point. The control point represents the new state that one desires for the breaker device. The indication point represents the current state of the breaker device. For SCADA to SCADA data exchange (e.g. control centre to control centre, control centre to SCADA master, etc.), additional data is often associated with point data. Quality of point data is often exchanged to defined whether the data is valid or not. In addition, for data that may be updated from alternate sources, quality often identifies the alternate source. Select-Before-Operate control is associated with Control Points for momentary inhibiting access except from one source. Two other informative data values are: time stamp and change of value counter. The time stamp, when available, details when a data value last changed. The change of value counter, when available, details the number of changes to the value.

From the context presented, the primary object models required are: Indication Point, and Control Point. The attributes Point Value, Quality, Select-Before-Operate, Time Stamp, and Change of Value Counter are required to meet the desired functionality for data exchange. The Indication Point and Control Point models may be logically combined to a single model to represent a device which implements a control function with a status indication as to its success/failure. The combined logical model will result in the same logical attributes, and map onto the same MMS types as the independent models.

5.2.2 IndicationPoint object

An IndicationPoint object represents an actual input point.

Object: **IndicationPoint** (Read Only)

Key Attribute: PointName

Attribute: PointType (REAL, STATE, DISCRETE, STATESUPPLEMENTAL)

Constraint PointType=REAL

Attribute: PointRealValue

Constraint PointType=STATE

Attribute: PointStateValue

Constraint PointType=DISCRETE

Attribute: PointDiscreteValue

Constraint PointType= STATESUPPLEMENTAL

Attribute: PointStateSupplementalValue

Attribute: QualityClass: (QUALITY, NOQUALITY)

Constraint: QualityClass = QUALITY

Attribute: Validity (VALID, HELD, SUSPECT, NOTVALID)

Attribute: CurrentSource (TELEMETERED, CALCULATED, ENTERED, ESTIMATED)

Attribute: NormalSource (TELEMETERED, CALCULATED, ENTERED, ESTIMATED)

Attribute: NormalValue (NORMAL, ABNORMAL)
Attribute: TimeStampClass: (TIMESTAMP, TIMESTAMPEXTENDED, NOTIMESTAMP)
Constraint: TimeStampClass = TIMESTAMP
Attribute: TimeStamp
Attribute: TimeStampQuality: (VALID, INVALID)
Constraint: TimeStampClass = TIMESTAMPEXTENDED
Attribute: TimeStampExtended
Attribute: TimeStampQuality: (VALID, INVALID)
Attribute: COVClass: (COV, NOCOV)
Constraint: COVClass = COV
Attribute: COVCounter

PointName

The PointName attribute uniquely identifies the object.

PointType

The PointType attribute identifies the type of input point, and must be one of the following: REAL, STATE, DISCRETE, STATESUPPLEMENTAL.

PointRealValue

The current value of the IndicationPoint, if the PointType attribute is REAL.

PointStateValue

The current value of the IndicationPoint, if the PointType attribute is STATE.

PointDiscreteValue

The current value of the IndicationPoint, if the PointType attribute is DISCRETE.

PointStateSupplementalValue

The current value of the IndicationPoint, if the PointType attribute is STATESUPPLEMENTAL.

A PointStateSupplementalValue shall have the ability to indicate the current value (State), tagging information (Tag), and the expected value/state (ExpectedState). If the ExpectedState value does not match the State value, this indicates that the provider of the ExpectedState value is indicating a potential issue.

QualityClass

The QualityClass has the value QUALITY if the object instance has any of the quality attributes (Validity, CurrentSource, or NormalValue), and takes the value NOQUALITY if none of the attributes are present.

Validity

The Validity attribute specifies the validity or quality of the PointValue data it is associated with. These are based on the source system's interpretation as follows:

Validity	Description
VALID	Data value is valid
HELD	Previous data value has been held over. Interpretation is local
SUSPECT	Data value is questionable. Interpretation is local
NOTVALID	Data value is not valid

CurrentSource

The CurrentSource attribute specifies the current source of the PointValue data it is associated with as follows:

CurrentSource	Description
TELEMETERED	The data value was received from a telemetered site
CALCULATED	The data value was calculated based on other data values
ENTERED	The data value was entered manually
ESTIMATED	The data value was estimated (State Estimator, etc.)

NormalSource

The NormalSource attribute specifies the normal source of the PointValue data it is associated with as follows:

NormalSource	Description
TELEMETERED	The data value is normally received from a telemetered site
CALCULATED	The data value is normally calculated based on other data values
ENTERED	The data value is normally entered manually
ESTIMATED	The data value is normally estimated (State Estimator, etc.)

NormalValue

The NormalValue attribute reports whether value of the PointValue attribute is normal. Only one bit is set, it is defined as follows:

NormalValue	Description
NORMAL	The point value is that which has been configured as normal for the point
ABNORMAL	The point value is not that which has been configured as normal for the point

TimeStampClass

The TimeStampClass attribute has the value **TIMESTAMP** or **TIMESTAMPEXTENDED** if the IndicationPoint is time stamped, and has the value **NOTTIMESTAMP** if the IndicationPoint contains no TimeStamp attribute.

TimeStamp

The TimeStamp attribute provides a time stamp (with a minimum resolution of one second) of when the value (attribute PointRealValue, PointStateValue, PointDiscreteValue, or PointStateSupplementalValue) of the IndicationPoint was last changed. It is set at the earliest possible time after collection of the IndicationPoint value from the end device.

TimeStampExtended

The TimeStampExtended attribute provides a time stamp (with a resolution of one millisecond) of when the value (attribute PointRealValue, PointStateValue, PointDiscreteValue, or PointStateSupplementalValue) of the IndicationPoint was last changed. It is set at the earliest possible time after collection of the IndicationPoint value from the end device.

TimeStampQuality

The TimeStampQuality attribute has the value VALID if the current value of the TimeStamp attribute contains the time stamp of when the value was last changed, and has the value INVALID at all other times.

COVClass

The COVClass (**C**hange **O**f **V**alue **C**ounter) attribute has the value COV if the IndicationPoint contains a COVCounter attribute, otherwise it has the value NOCOV.

COVCounter

The COVCounter attribute specifies the number of times the value (attribute PointRealValue, PointStateValue, PointDiscreteValue, or PointStateSupplementalValue) of the IndicationPoint has changed. It is incremented each time the owner sets a new value for the IndicationPoint.

5.2.3 ControlPoint Object

A ControlPoint Object is an integral part of the services provided by TASE.2. It is used to represent values of various types of data typical of SCADA and energy management systems. Typically, a ControlPoint object will be associated with some real world object.

Object: **ControlPoint** (Write Only, except for attributes CheckBackName, Tag, State and Reason)

Key Attribute: ControlPointName

Attribute: ControlPointType: (COMMAND, SETPOINT)

Constraint: ControlPointType = COMMAND

Attribute: CommandValue

Constraint: ControlPointType = SETPOINT

Attribute: SetPointType: (REAL, DISCRETE)

Constraint SetpointType=REAL

Attribute: SetpointRealValue

Constraint SetpointType=DISCRETE

Attribute: SetpointDiscreteValue

Attribute: DeviceClass: (SBO, NONSBO)

Constraint: DeviceClass = SBO

Attribute: CheckBackName

Attribute: State: (SELECTED, NOTSELECTED)

Attribute: Timeout

Attribute: TagClass: (TAGGABLE, NONTAGGABLE)

Constraint: TagClass = TAGGABLE

Attribute: Tag: (NO-TAG, OPEN-AND-CLOSE-INHIBIT, CLOSE-ONLY-INHIBIT)

Attribute: State: (IDLE, ARMED)

Attribute: Reason

ControlPointName

The ControlPointName attribute uniquely identifies the object.

ControlPointType

The value of the ControlPointType attribute for an instance of a ControlPoint will be COMMAND or SETPOINT, indicating the type of controlled device.

CommandValue

The CommandValue attribute indicates the command for a device.

SetPointType

The value of the SetPointType attribute for an instance of a ControlPoint of ControlPointType SETPOINT will be REAL or DISCRETE, indicating the type of setpoint.

SetPointRealValue

The SetPointRealValue attribute may be set with the floating point value requested for the setpoint control.

SetPointDiscreteValue

The SetPointDiscreteValue attribute may be set with the integer value requested for the setpoint control.

DeviceClass

The DeviceClass attribute of an instance of a ControlPoint has the value SBO if the device requires a Select operation before being operated, and the value NONSBO otherwise.

CheckBackName

The CheckBackName attribute contains a symbolic description of the physical object being controlled. This data is returned by the system operating the physical object to the system requesting the operation so that the person or system requesting the operation can be assured the proper object has been selected.

State

The State attribute indicates whether the ControlPoint is SELECTED or NOTSELECTED.

Timeout

The Timeout attribute of an instance of a ControlPoint has the value of the maximum allowed time for which the ControlPoint of DeviceClass SBO may remain SELECTED before operation.

TagClass

The TagClass attribute of an instance of a ControlPoint has the value TAGGABLE if the instance contains a Tag attribute, and otherwise has the value NONTAGGABLE.

Tag

The Tag attribute indicates whether or not the ControlPoint is tagged, and if it is, what the level of tagging is. The Tag attribute can take on the values NO-TAG, OPEN-AND-CLOSE-INHIBIT, CLOSE-ONLY-INHIBIT.

Reason

The Reason attribute contains a message that indicates the reason for tagging.

5.2.4 Protection Equipment Event Object Model

The following object model represents the events generated in the operation of protection equipment. Start events are generated by the protection equipment when it detects faults. Trip events report commands to output circuits which are generated by the protection equipment when it decides to trip the circuit-breaker. Both events are transient information. The protection event models are based on IEC 60870-5-101.

Object: **ProtectionEvent**

KeyAttribute: Name

Attribute: ElapsedTimeValidity (VALID, INVALID)

Attribute: Blocked (NOTBLOCKED, BLOCKED)

Attribute: Substituted (NOTSUBSTITUTED, SUBSTITUTED)

Attribute: Topical (TOPICAL, NOTTOPICAL)

Attribute: EventValidity (VALID, INVALID)

Attribute: ProtectionClass (SINGLE, PACKED)

Constraint: ProtectionClass = SINGLE

Attribute: EventState (INDETERMINATE, OFF, ON)

Attribute: EventDuration

Attribute: EventTime

Constraint: ProtectionClass = PACKED

Attribute: EventClass (START, TRIP)

Constraint: EventClass = START

Attribute: StartGeneral (NOSTART, START)

Attribute: StartPhase1 (NOSTART, START)

Attribute: StartPhase2 (NOSTART, START)

Attribute: StartPhase3 (NOSTART, START)

Attribute: StartEarth (NOSTART, START)

Attribute: StartReverse (NOSTART, START)

Attribute: DurationTime

Attribute: StartTime

Constraint: EventClass = TRIP

Attribute: TripGeneral (NOTRIP, TRIP)

Attribute: TripPhase1 (NOTRIP, TRIP)

Attribute: TripPhase2 (NOTRIP, TRIP)

Attribute: TripPhase3 (NOTRIP, TRIP)

Attribute: OperatingTime

Attribute: TripTime

Name

The Name attribute uniquely identifies the protection event.

ElapsedTimeValidity

The elapsed time (attribute EventDuration, DurationTime, or OperatingTime depending on the event type) is valid if it was correctly acquired. If the acquisition function detects invalid conditions, the ElapsedTimeValidity attribute is INVALID, otherwise it is VALID.

Blocked

The Blocked attribute is BLOCKED if the value of protection event is blocked for transmission, and is NOTBLOCKED otherwise. The value remains in the state that was acquired before it was blocked. Blocking and deblocking may be initiated by a local lock or by a local automatic cause.

Substituted

The Substituted attribute takes the value SUBSTITUTED if the event was provided by input of an operator (dispatcher) or by an automated source.

Topical

The Topical attribute is TOPICAL if the most recent update was successful, and is NOTTOPICAL if it was not updated successfully during a specified time interval or is unavailable.

EventValidity

The EventValidity attribute takes the value INVALID if the acquisition function recognizes abnormal conditions of the information source, otherwise it is VALID.

ProtectionClass

The ProtectionClass attribute identifies the type of protection event, and must be one of the following: SINGLE or PACKED.

EventState

The EventState attribute of a SINGLE protection event takes the value of the protection event: OFF, ON or INDETERMINATE.

EventDuration

The EventDuration attribute takes the value of the event duration (total time the fault was detected) or operation time (time between start of operation and trip command execution).

EventTime

The EventTime attribute signifies the time of the start of the operation.

EventClass

The type of protection event being reported. The value START signifies a start event, and TRIP signifies a trip event.

StartGeneral

The value NOSTART signifies no general start of operation, and START signifies that the event includes a general start of operation.

StartPhase1

The value NOSTART for StartPhase1 implies that Phase L1 was not involved in the event, START implies that it was involved.

StartPhase2

The value NOSTART for StartPhase2 implies that Phase L2 was not involved in the event, START implies that it was involved.

StartPhase3

The value NOSTART for StartPhase3 implies that Phase L3 was not involved in the event, START implies that it was involved.

StartEarth

The value NOSTART for StartEarth implies that earth current was not involved in the event, START implies that it was involved.

StartReverse

The value NOSTART for StartReverse implies that reverse direction was not involved in the event, START implies that it was involved.

DurationTime

Time in milliseconds from the start of operation until the end of operation.

StartTime

Time of the start of operation of the protection equipment.

TripGeneral

The TripGeneral attribute takes on the value of TRIP if a general command to the output circuit was issued during the operation, NOTRIP otherwise.

TripPhase1

The TripPhase1 attribute takes on the value of TRIP if a command to output circuit Phase L1 command was issued during the operation, NOTRIP otherwise.

TripPhase2

The TripPhase2 attribute takes on the value of TRIP if a command to output circuit Phase L2 command was issued during the operation, NOTRIP otherwise.

TripPhase3

The TripPhase3 attribute takes on the value of TRIP if a command to output circuit Phase L3 command was issued during the operation, NOTRIP otherwise.

OperatingTime

The time in milliseconds from the start of operation until the first command to an output circuit was issued.

TripTime

Time of the start of the operation.

5.3 Device Outage Object

A DeviceOutage object is used to communicate schedule information regarding device outages. It is composed of a number of objects which define the device which will be (was) affected and describe the time period for which the outage will occur.

Object: DeviceOutage

- Key Attribute: OutageReferenceId
- Attribute: OwningUtilityID
- Attribute: Timestamp
- Attribute: StationName
- Attribute: DeviceType (GENERATOR, TRANSFORMER, CAPACITOR, TRANSMISSION_CIRCUIT, BREAKER_SWITCH, INDUCTOR, OTHER)
- Attribute: DeviceName
- Attribute: DeviceNumber
- Attribute: DeviceRating
- Attribute: ActivityDateAndTime
- Attribute: Activity (NEWPLAN, REVISE, CANCEL, ACTUAL)
- Constraint: Activity = NEWPLAN, REVISE
 - Attribute: PlanType (SCHEDULED, ESTIMATED)
 - Attribute: PlannedOpenOrOutOfServiceDateAndTime
 - Attribute: PlannedCloseOrInServiceDateAndTime
 - Attribute: OutagePeriod (CONTINUOUS, DAILY, WEEKDAYS, OTHER)
 - Attribute: OutageType (FORCED, MAINTENANCE, PARTIAL, ECONOMY, UNPLANNED, OTHER)
 - Attribute: OutageAmountType (PARTIAL, FULL)
 - Constraint: OutageAmountType = PARTIAL
 - Attribute: Amount
 - Attribute: UpperOperatingLimit
 - Attribute: LowerOperatingLimit
 - Attribute: Class (INSERVICE, OUTSERVICE)
- Constraint: Activity = ACTUAL
 - Attribute: Action (TRIPPED, OFFLINE, ONLINE, OPEN, CLOSE)
 - Constraint: Action = TRIPPED, OFFLINE, OPEN
 - Attribute: Affected Amount
- Attribute: Comments
- Attribute: OutageEffect

OutageReferenceId

The **OutageReferenceId** attribute is a unique reference value assigned by the originator for identifying this particular outage.

OwningUtilityID

ID of the utility or control area owning the equipment.

TimeStamp

The **TimeStamp** attribute provides the time at which a **DeviceOutage** object was generated.

StationName

Name of station at which the affected equipment, circuit, etc. resides.

DeviceType

Type of the device which is affected.

DeviceName

Name of the device which is affected.

DeviceNumber

The **DeviceNumber** attribute provides further qualification of **DeviceName** in cases where **DeviceName** may not provide sufficient specificity. As an example, multiple transmission lines may connect the same two substations and be assigned a single **DeviceName**. To uniquely identify one of these multiple lines, each line is assigned a **DeviceNumber**.

DeviceRating

The Device rating in KV, MW, MVAR.

ActivityDateAndTime

This is the time that the activity occurred or, for a planned activity or cancellation, this the time the activity is planned to occur.

Activity

This describes the type of activity reported. A **NEWPLAN** or **REVISE** requires dates for starting and ending the outage. A **CANCEL** only requires a cancellation date, which is recorded in the **ActivityDateAndTime**. An **ACTUAL** activity requires a date the event occurred, which is recorded in the **ActivityDateAndTime**, and a description of the actual event that occurred.

PlanType

A plan can be either a scheduled activity (with firm committed dates) or an activity with only estimated dates.

PlannedOpenOrOutOfServiceDateAndTime

Date and time the device is going to be taken out of service (or switch/breaker is to be opened).

PlannedCloseOrInServiceDateAndTime

Date and time the device is going to be returned to service (or switch/breaker is to be closed).

OutagePeriod

This describes the periodicity of the outage for plans which are to be executed periodically, such as daily or weekly.

OutageType

This describes the reason the equipment is being taken out of service. FORCED is a controlled but unscheduled outage MAINTENANCE is a scheduled outage for maintenance purposes PARTIAL is an outage where only a portion of the capacity is removed from service. ECONOMY is an outage planned for economic reasons. UNPLANNED is an unscheduled outage.

OutageAmountType

This describes whether the equipment outage is PARTIAL, in that some capacity is still available, or FULL, in which case no capacity is available.

Amount

For partial outages, this is the amount of normal capacity which will be unavailable during outage.

UpperOperatingLimit

Upper limit of operating range during outage.

LowerOperatingLimit

Lower limit of operating range during outage.

Class

Specifies whether device is in service or out of service when outage occurs.

Action

Describes the event which caused the outage or return to service.

AffectedAmount

This describes the load being carried at the time the outage starts.

Comments

This field contains 128 characters of ASCII text which may be used for operator comments.

OutageEffect

Effect of outage on transfer capacity. The field contains up to 128 characters of ASCII text which may be used to describe the effect of the outage on transfer capacity.

5.4 InformationBuffer Object

An InformationBuffer object is used to send multiple line ASCII text messages or binary data. It may be used to transfer messages limited in size to the maximum message size of the underlying communications structure (i.e. the maximum MMS PDU size). The application and coding of the content of this object is outside the scope of TASE.2, and is left as a local issue for agreement between the sending and receiving implementations. Note that this object is referred to as the Information Message object in IEC 60870-6-503.

Object: **InformationBuffer**

Key Attribute: InfoReference

Attribute: LocalReference

Attribute: Messageld

Attribute: Size

Attribute: InfoStream

InfoReference

The InfoReference attribute uniquely identifies the object. It is used to identify and/or trigger some special handling required by the receiving system.

LocalReference

The LocalReference attribute specifies a value agreed upon between the sender and receiver of the message that further identifies the message (file name, application identification, etc.).

Messageld

Identifies the particular instance of a message.

Size

The length of the valid data in the InfoStream.

InfoStream

The InfoStream attribute contains the byte stream of information being passed. It is limited only by the maximum size of a single message.

6 MMS Types for Object Exchange

6.1 General

This clause defines the MMS Types to be used within TASE.2 for exchanging standard objects. The mapping of the objects onto these types is defined in Clause 7. The MMS type definitions are defined in terms of ASN.1 value notation, following the MMS grammar for Data as defined in ISO 9506-1 and ISO 9506-2.

Throughout this clause, all field widths specified are maximum field widths. The process of ASN.1 encoding used within MMS may reduce the actual transmitted widths to the minimum required to represent the value being transmitted.

6.2 Supervisory Control and Data Acquisition Types

6.2.1 IndicationPoint Type Descriptions

The following foundation types are referenced in complex IndicationPoint Type Descriptions:

Data_Real floating-point: { format-width 32, exponent-width 8 }

Data_State bit-string:

```
{
    State_hi[0],
    State_lo[1],
    Validity_hi[2],
    Validity_lo[3],
    CurrentSource_hi[4],
    CurrentSource_lo[5],
    NormalValue[6],
    TimeStampQuality[7]
}
```

Data_Discrete integer {width 32 }

Data_StateSupplemental bit-string:

```
{
    State_hi[0],
    State_lo[1],
    Tag_hi[2],
    Tag_lo[3],
    ExpectedState_hi[4],
    ExpectedState_lo[5],
    Reserved[6],
    Reserved[7]
}
```

Data_Flags bit-string:

```
{
    unused[0],
    unused[1],
    Validity_hi[2],
    Validity_lo[3],
    CurrentSource_hi[4],
    CurrentSource_lo[5],
    NormalValue[6],
    TimeStampQuality[7]
}
```

Data_TimeStamp GMTBasedS

Data_TimeStampExtended TimeStampExtended

COV_Counter unsigned { width 16 }

The following complex types are used in transferring IndicationPoint object values:

Data_RealIQ STRUCTURE

```
{
```

COMPONENT Value	Data_Real,
COMPONENT Flags	Data_Flags

}

Data_StateQ STRUCTURE

{

COMPONENT Value	Data_State,
COMPONENT Flags	Data_Flags

}

Data_DiscreteQ STRUCTURE

{

COMPONENT Value	Data_Discrete,
COMPONENT Flags	Data_Flags

}

Data_StateSupplementalQ STRUCTURE

{

COMPONENT Value	Data_StateSupplemental,
COMPONENT Flags	Data_Flags

}

Data_RealQTimeTag STRUCTURE

{

COMPONENT Value	Data_Real,
COMPONENT TimeStamp	Data_TimeStamp,
COMPONENT Flags	Data_Flags

}

Data_StateQTimeTag STRUCTURE

{

COMPONENT TimeStamp	Data_TimeStamp,
COMPONENT Flags	Data_State

}

Data_DiscreteQTimeTag STRUCTURE

{

COMPONENT Value	Data_Discrete,
COMPONENT TimeStamp	Data_TimeStamp,
COMPONENT Flags	Data_Flags

}

Data_StateSupplementalQTimeTag STRUCTURE

{

COMPONENT Value	Data_StateSupplemental,
COMPONENT TimeStamp	Data_TimeStamp,
COMPONENT Flags	Data_Flags

}

Data_RealExtended STRUCTURE

{

COMPONENT Value	Data_Real,
COMPONENT TimeStamp	Data_TimeStamp,
COMPONENT Flags	Data_Flags,
COMPONENT COV	COVCounter

```

}
Data_StateExtended STRUCTURE
{
    COMPONENT TimeStamp      Data_TimeStamp,
    COMPONENT Flags          Data_State,
    COMPONENT COV            COVCounter
}
Data_DiscreteExtended STRUCTURE
{
    COMPONENT Value          Data_Discrete,
    COMPONENT TimeStamp      Data_TimeStamp,
    COMPONENT Flags          Data_Flags,
    COMPONENT COV            COVCounter
}
Data_StateSupplementalExtended STRUCTURE
{
    COMPONENT Value          Data_StateSupplemental,
    COMPONENT TimeStamp      Data_TimeStamp,
    COMPONENT Flags          Data_Flags,
    COMPONENT COV            COVCounter
}

Data_RealQTimeTagExtended STRUCTURE
{
    COMPONENT Value          Data_Real,
    COMPONENT TimeStamp      Data_TimeStampExtended,
    COMPONENT Flags          Data_Flags
}

Data_StateQTimeTagExtended STRUCTURE
{
    COMPONENT TimeStamp      Data_TimeStampExtended,
    COMPONENT Flags          Data_State
}
Data_DiscreteQTimeTagExtended STRUCTURE
{
    COMPONENT Value          Data_Discrete,
    COMPONENT TimeStamp      Data_TimeStampExtended,
    COMPONENT Flags          Data_Flags
}
Data_State_SupplementalQTimeTagExtended STRUCTURE
{
    COMPONENT Value          Data_StateSupplemental,
    COMPONENT TimeStamp      Data_TimeStampExtended,
    COMPONENT Flags          Data_Flags
}
IndicationPointConfig STRUCTURE
{
    COMPONENT PointType      integer { width 8, range 0 .. 2 },
    COMPONENT QualityClass   integer { width 8, range 0 .. 1 },
    COMPONENT NormalSource   integer { width 8, range 0 .. 3 },

```

```

        COMPONENT TimeStampClass    integer { width 8, range 0 .. 1 },
        COMPONENT COVClass          integer { width 8, range 0 .. 1 }
    }

```

6.2.2 ControlPoint Type Descriptions

The following foundation types are referenced in complex type descriptions:

```

Control_Command          integer { width 16 }
Control_Setpoint_Real    floating-point { format-width 32, exponent-width 8 }
Control_Setpoint_Discrete integer { width 16 }
SBO_CheckBackName        integer { width 16 }
SelectState              boolean
TagFlags                bit-string:
    {
        tag_hi [0],
        tag_lo [1],
        tag_state[2]
    }
TextString                VisibleString { width 255 }

```

The following complex type descriptions are used in accessing ControlPoint object values:

SBO STRUCTURE

```

{
    COMPONENT TimeOut          Data_TimeStamp,
    COMPONENT Select           SelectState
}

```

Tag_Value STRUCTURE

```

{
    COMPONENT Flags            TagFlags,
    COMPONENT Reason          TextString
}

```

ControlConfig STRUCTURE

```

{
    COMPONENT ControlPointType integer { width 8, range 0..2 },
    COMPONENT SetPointType     integer { width 8, range 0 .. 2 },
    COMPONENT DeviceClass      integer { width 8, range 0 .. 1 },
    COMPONENT TagClass         integer { width 8, range 0 .. 1 }
}

```

6.2.3 Protection Equipment Type Descriptions

The following foundation types are used to build complex types for report protection equipment events.

SingleFlags bit-string:

```

{
    ElapsedTimeValidity[0],
    Blocked[1],
    Substituted[2],
    Topical[3],
    EventValidity[4],
    unused[5],
    EventState_hi[6],
}

```

```

        EventState_lo[7]
    }
EventFlags bit-string:
    {
        General[0],
        Phase1[1],
        Phase2[2],
        Phase3[3],
        Earth[4],
        Reverse[5],
        unused[6],
        unused[7]
    }
PackedFlags bit-string:
    {
        ElapsedTimeValidity[0],
        Blocked[1],
        Substituted[2],
        Topical[3],
        EventValidity[4],
        unused[5],
        unused[6],
        unused[7]
    }

```

The following complex types are used to report protection equipment events.

SingleProtectionEvent STRUCTURE

```

{
    COMPONENT SingleEventFlags      SingleFlags,
    COMPONENT OperatingTime         TimeIntervalL16,
    COMPONENT EventTime              TimeStampExtended
}

```

PackedProtectionEvent STRUCTURE

```

{
    COMPONENT PackedEvent           EventFlags,
    COMPONENT PackedEventFlags      PackedFlags,
    COMPONENT OperatingTime         TimeIntervalL16,
    COMPONENT EventTime              TimeStampExtended
}

```

6.3 Device Outage Type Descriptions

The following foundation types are referenced in complex type descriptions for the DeviceOutage object:

ActionId	integer {width 32}
ClassId	integer {width 16}
CommentString	VisibleString 128
Data_Real	floating-point {format-width 32, exponent-width 8}
DeviceId	integer {width 32}
DeviceName	visiblestring {width 32 characters, varying}
DeviceNumber	integer {width 32}
Number	integer {width 16}

OutageAmountTypeId	integer {width 16}
OutagePeriodId	integer {width 16}
OutageTypeId	integer {width 16}
PlanTypeId	integer {width 16}
ReferenceNum	integer {width 32}
ScheduleTime	GMTBasedS
StationNameString	VisibleString 32
TimeStampS	GMTBasedS
UtilityId	integer {width 32}

DeviceOutage composite type definitions are:

DONewRevSched STRUCTURE

{	
COMPONENT OutageReferenceId	ReferenceNum,
COMPONENT OwningUtilityId	UtilityId,
COMPONENT TimeStamp	TimeStampS,
COMPONENT StationName	StationNameString,
COMPONENT DeviceName	DeviceName,
COMPONENT DeviceType	DeviceId,
COMPONENT DeviceNumber	Number,
COMPONENT DeviceRating	Data_Real,
COMPONENT ActivityDateAndTime	ScheduleTime,
COMPONENT PlanType	PlanTypeId,
COMPONENT PlannedOpenOrOutOfServiceDateAndTime	ScheduleTime,
COMPONENT PlannedCloseOrInServiceDateAndTime	ScheduleTime,
COMPONENT OutagePeriod	OutagePeriodId,
COMPONENT OutageType	OutageTypeId,
COMPONENT OutageTypeAmount	OutageTypeAmountId,
COMPONENT Amount	Data_Real,
COMPONENT UpperOperatingLimit	Data_Real,
COMPONENT LowerOperatingLimit	Data_Real,
COMPONENT Class	ClassId,
COMPONENT Comments	CommentString,
COMPONENT OutageEffect	CommentString
}	

DOCancel STRUCTURE

{	
COMPONENT OutageReferenceId	ReferenceNum,
COMPONENT OwningUtilityId	UtilityId,
COMPONENT TimeStamp	TimeStampS,
COMPONENT StationName	StationNameString,
COMPONENT DeviceName	DeviceName,
COMPONENT DeviceType	DeviceId,
COMPONENT DeviceNumber	Number,
COMPONENT DeviceRating	Data_Real,
COMPONENT ActivityDateAndTime	ScheduleTime,
COMPONENT Comments	CommentString,
COMPONENT OutageEffect	CommentString
}	

DOActual STRUCTURE

{

COMPONENT OutageReferenceId	ReferenceNum,
COMPONENT OwningUtilityId	UtilityId,
COMPONENT TimeStamp	TimeStampS,
COMPONENT StationName	StationNameString,
COMPONENT DeviceName	DeviceName,
COMPONENT DeviceType	DeviceId,
COMPONENT DeviceNumber	Number,
COMPONENT DeviceRating	Data_Real,
COMPONENT ActivityDateAndTime	ScheduleTime,
COMPONENT Action	ActionId,
COMPONENT AffectedAmount	Data_Real,
COMPONENT Comments	CommentString,
COMPONENT OutageEffect	CommentString
}	

6.4 InformationBuffer Type Descriptions

The following foundation types are referenced in complex type descriptions for the InformationBuffer object:

ReferenceNum	integer {width 32}
Number	integer {width 32}

The following types are used in exchanging the Information Message object types:

InfoMessHeader STRUCTURE

{	
COMPONENT InfoReference	ReferenceNum,
COMPONENT Localreference	ReferenceNum,
COMPONENT MessageId	ReferenceNum,
COMPONENT Size	Number
}	

InfoBufXX OCTET STRING {width XX octets}

where XX is the number of octets in the buffer. Any number of bytes are permitted, although the maximum buffer size should not exceed the maximum MMS PDU size.

Examples: 64 byte buffer = InfoBuf64, 256 byte buffer = InfoBuf256, 1024 byte buffer = InfoBuf1024. Note that leading zeros are not permitted.

7 Mapping of Object Models to MMS Types

7.1 Supervisory Control and Data Mapping

7.1.1 Indication Object Mapping

This clause defines the mapping of each object attributes onto MMS. In general, most objects are represented by one or more MMS Named Variables of the predefined TASE.2 types from Clause 6.

PointName

Maps to an MMS variable identifier (either VMD specific or Domain specific)

PointType

Used in selecting the named type of the variable. If COVClass is NOCOV, the type of the MMS variable is selected according to the following criteria:

PointType	QualityClass	TimeStampClass	Map to type:
REAL	NOQUALITY	NOTIMESTAMP	Data_Real
STATE	NOQUALITY	NOTIMESTAMP	Data_State
DISCRETE	NOQUALITY	NOTIMESTAMP	Data_Discrete
STATE SUPPLEMENTAL	NOQUALITY	NOTIMESTAMP	Data_StateSupplemental
REAL	QUALITY	NOTIMESTAMP	Data_RealQ
STATE	QUALITY	NOTIMESTAMP	Data_StateQ
DISCRETE	QUALITY	NOTIMESTAMP	Data_DiscreteQ
STATE SUPPLEMENTAL	QUALITY	NOTIMESTAMP	Data_StateSupplementalQ
REAL	QUALITY	TIMESTAMP	Data_RealQTimeTag
STATE	QUALITY	TIMESTAMP	Data_StateQTimeTag
DISCRETE	QUALITY	TIMESTAMP	Data_DiscreteQTimeTag
STATE SUPPLEMENTAL	QUALITY	TIMESTAMP	Data_StateSupplementalQTimeTag
REAL	QUALITY	TIMESTAMPEXTENDED	Data_RealQTimeTagExtended
STATE	QUALITY	TIMESTAMPEXTENDED	Data_StateQTimeTagExtended
DISCRETE	QUALITY	TIMESTAMPEXTENDED	Data_DiscreteQTimeTagExtended
STATE SUPPLEMENTAL	QUALITY	TIMESTAMPEXTENDED	Data_StateSupplementalQTimeTag Extended

If COVClass is COV, the following criteria are used:

PointType	Map to type:
REAL	Data_RealExtended
STATE	Data_StateExtended
DISCRETE	Data_DiscreteExtended
STATESUPPLEMENTAL	Data_StateSupplementalExtended

The PointType attribute may optionally be mapped to the **PointType** component of an MMS named variable of type **IndicationPointConfig** with the following interpretation: 0=STATE, 1=DISCRETE, 2=REAL, 3=StateSupplemental.

PointRealValue

If present, maps to either the value of an MMS variable of type **Data_Real** (if QualityClass and TimeStampClass are NOQUALITY, NOTIMESTAMP) or to the **Value** COMPONENT of the MMS variable.

PointStateValue

If present, maps to either the value of an MMS variable of type **Data_State** (if QualityClass and TimeStampClass are NOQUALITY, NOTIMESTAMP) or to bits **State_hi** and **State_lo** of the **Flags** COMPONENT of the MMS variable.