International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.1080.2

(10/2021)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

Information and network security – Telebiometrics

Biology-to-machine (B2M) protocol

Recommendation ITU-T X.1080.2

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	X.1–X.199
OPEN SYSTEMS INTERCONNECTION	X.200–X.299
INTERWORKING BETWEEN NETWORKS	X.300–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	X.600–X.699
OSI MANAGEMENT	X.700–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	X.850–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	
General security aspects	X.1000–X.1029
Network security	X.1030–X.1049
Security management	X.1050–X.1069
Telebiometrics	**X.1080–X.1099**
SECURE APPLICATIONS AND SERVICES (1)	
Multicast security	X.1100–X.1109
Home network security	X.1110–X.1119
Mobile security	X.1120–X.1139
Web security (1)	X.1140–X.1149
Application Security (1)	X.1150–X.1159
Peer-to-peer security	X.1160–X.1169
Networked ID security	X.1170–X.1179
IPTV security	X.1180–X.1199
CYBERSPACE SECURITY	
Cybersecurity	X.1200–X.1229
Countering spam	X.1230–X.1249
Identity management	X.1250–X.1279
SECURE APPLICATIONS AND SERVICES (2)	
Emergency communications	X.1300–X.1309
Ubiquitous sensor network security	X.1310–X.1319
Smart grid security	X.1330–X.1339
Certified mail	X.1340–X.1349
Internet of things (IoT) security	X.1350–X.1369
Intelligent transportation system (ITS) security	X.1370–X.1399
Distributed ledger technology (DLT) security	X.1400–X.1429
Application Security (2)	X.1450–X.1459
Web security (2)	X.1470–X.1489
CYBERSECURITY INFORMATION EXCHANGE	
Overview of cybersecurity	X.1500–X.1519
Vulnerability/state exchange	X.1520–X.1539
Event/incident/heuristics exchange	X.1540–X.1549
Exchange of policies	X.1550–X.1559
Heuristics and information request	X.1560–X.1569
Identification and discovery	X.1570–X.1579
Assured exchange	X.1580–X.1589
Cyber Defence	X.1590–X.1599
CLOUD COMPUTING SECURITY	
Overview of cloud computing security	X.1600–X.1601
Cloud computing security design	X.1602–X.1639
Cloud computing security best practices and guidelines	X.1640–X.1659
Cloud computing security implementation	X.1660–X.1679
Other cloud computing security	X.1680–X.1699
QUANTUM COMMUNICATION	
Terminologies	X.1700–X.1701
Quantum random number generator	X.1702–X.1709
Framework of QKDN security	X.1710–X.1711
Security design for QKDN	X.1712–X.1719
Security techniques for QKDN	X.1720–X.1729
DATA SECURITY	
Big Data Security	X.1750–X.1759
Data protection	X.1770–X.1789
IMT-2020 SECURITY	X.1800–X.1819

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.1080.2

Biology-to-machine (B2M) protocol

Summary

Recommendation ITU-T X.1080.2 defines a general protocol for the exchange of biometric information from a patient facility to a medical expert facility so that a medical centre could remotely monitor a patient and retrieve information from that patient. It also allows the medical expert facility to control the sensors and other devices at the patient facility and to establish the environment for a monitor session at the patient facility. It defines a versatile and open-ended information model that allows any type of medical and non-medical information to be transferred.

This Recommendation is a biosignal communication protocol between computing devices and biological systems. This protocol is based upon an aggregation of the interactions between a computing device and the biological system. Biometric interactions are described using the telebiometric multimodal model as defined by Recommendation ITU-T X.1081, which is a three-layer model combining the science, sensor and metric layers. It is a horizontal market protocol designed to be used for all IoT biometric metric applications, for example, in the aerospace, medical, automotive, industrial and consumer markets. In the case of clinical medical trials, this protocol enriches the application with a versatile and open-ended information model filled with interaction descriptions thereby ensuring an accurate comparison of measurement processes.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T X.1080.2	2021-10-29	17	11.1002/1000/14795

Keywords

Biometrics, biosignals, e-health, Internet of Things (IoT), telebiometrics.

---

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Introduction

This Recommendation is organized as follows:

- SECTION 1 includes general information such as an overview of the biology-to-machine (B2M) protocol and the associated information models.
- SECTION 2 is a specification of the information models as the basis for the B2M protocol.
- SECTION 3 is the formal specification OF the B2M protocol.
- SECTION 4 is A list of information types to be carried in the B2M protocol.
- SECTION 5 includes application specific monitor type specifications. Currently, only moving detection monitor type is included.
- Annex A, which is an integral part of this Recommendation, provides a general introduction to sensing.
- Annex B, which is an integral part of this Recommendation, provides the formal ASN.1 specification for the B2M protocol and associated information model.
- Annex C, which is an integral part of this Recommendation, provides the formal ASN.1 specification for a list of defined information objects.
- Annex D, which is an integral part of this Recommendation, provides the formal ASN.1 specification for information objects representing monitor types.
- Annex E, which is an integral part of this Recommendation, provides the formal ASN.1 specification for the moving detection monitor types.

Recommendation ITU-T X.1080.2

Biology-to-machine protocol

SECTION 1 – GENERAL

1 Scope

The scope of this Recommendation is to include a versatile information model and and associated protocol entitled the biology-to-machine (B2M) protocol, a framework for communicating biosignals through biometric interactions using the telebiometrics multimodal model as defined by [ITU-T X.1081], which is a three-layer model combining the science, sensor and metric layers.

The scope is limited to the communication between a medical centre (medical expert facility) and a patient location.

The scope also includes definitions of some information objects to be carried by the protocol. Future editions of this Recommendation and other ITU-T Recommendations and/or international standards may add additional information objects.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

2.1 Identical Recommendations | International Standards

[ITU-T X.510] Recommendation ITU-T X.510 (2020) | ISO/IEC 9594-11:2020, *Information technology – Open Systems Interconnection – The Directory: Protocol specifications for secure operations*.

[ITU-T X.660] Recommendation ITU-T X.660 (2011) | ISO/IEC 9834-1:2012, *Information technology – Procedures for the operation of object identifier registration authorities: General procedures and top arcs of the international object identifier tree*.

[ITU-T X.680] Recommendation ITU-T X.680 (2021) | ISO/IEC 8824-1:2021, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*.

[ITU-T X.681] Recommendation ITU-T X.681 (2021) | ISO/IEC 8824-2:2021, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification*.

[ITU-T X.682] Recommendation ITU-T X.682 (2021) | ISO/IEC 8824-3:2021, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification*.

[ITU-T X.683] Recommendation ITU-T X.683 (2021) | ISO/IEC 8824-4:2021, *Information technology – Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications*.

[ITU-T X.691] Recommendation ITU-T X.691 (2021) | ISO/IEC 8825-2:2021, *Information technology – ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)*.

2.2 Additional references

[ITU-T X.1080.0] Recommendation ITU-T X.1080.0 (2017), A*ccess control for telebiometrics data protection*.

[ITU-T X.1080.1] Recommendation ITU-T X.1080.1 (2018), *E-health and worldwide telemedicines – Generic telecommunications protocol*.

[RFC 5890] IETF RFC 5890 (2010), *Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 association [ITU-T X.510]: A cooperative relationship between two application entities, which enables the communication of information and the coordination of their joint operation for an instance of communication.

3.1.2 biometrics [b-ITU-T X.1081]: Automated recognition of living persons based on observation of behavioural and biological (anatomical and physiological) characteristics.

3.1.3 client [ITU-T X.510]: The entity that initiates an association.

3.1.4 information object [ITU-T X.681]: An instance of some information object class, being composed of a set of fields which conform to the field specifications of the class.

3.1.5 information object class [ITU-T X.681]: A set of fields, forming a template for the definition of a potentially unbounded collection of information objects, the instances of the class.

3.1.6 object identifier [ITU-T X.660]: An ordered list of primary integer values from the root of the international object identifier tree to a node, which unambiguously identifies that node.

3.1.7 protected protocol data unit (PrPDU) [ITU-T X.510]: Application protocol data unit (APDU) defined by an application protocol to be protected by the wrapper protocol.

3.1.8 protocol data unit [ITU-T X.510]: Data that is transmitted as a single unit at the application layer between two application entities.

3.1.9 server [ITU-T X.510]: The entity that accepts or rejects an association.

3.1.10 telebiometrics [b-ITU-T X.1081]: The application of biometrics to telecommunications and of telecommunications to remote biometric sensing.

3.1.11 wrapper protocol data unit (WrPDU) [ITU-T X.510]: An application protocol data unit (APDU) carrying security protocol control information and, when relevant, carrying a protected protocol data unit.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 microwave motion detector: A type of sensor that emits electromagnetic pulses and measures the changes in frequency (Doppler) due to the reflection from a moving object.

3.2.2 passive infrared (PIR) sensor: An electronic sensor that measures infrared light radiating from objects in its field of view. They are most often used in PIR-based motion detectors.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

APDU Application Protocol Data Unit

ASN.1 Abstract Syntax Notation One

B2M Biology-to-Machine

ECG Electrocardiogram

IDN Internationalized Domain Name

LDH Letters, Digits, Hyphen

M2M Machine-to-Machine

OTA Over-The-Air

PER Packed Encoding Rules

PIR Passive Infrared

PrPDU Protected protocol Data Unit

SDO Standards Development Organization

URL Uniform Resource Locator

URN Uniform Resource Name

UTM Universal Transverse Mercator

WrPDU Wrapper protocol Data Unit

5 Conventions

When ASN.1 types and values are referenced in normal text, they are differentiated from normal text by presenting them in the **`10 pt bold Courier New`** typeface.

This Directory Specification makes extensive use of the Abstract Syntax Notation One (ASN.1) for the formal specification of data types and values, as specified in Recommendations, ITU-T X.680 | ISO/IEC 8824-1, ITU-T X.681 (2015) | ISO/IEC 8824-2, ITU-T X.682 (2015) | ISO/IEC 8824-3, ITU-T X.683 (2015) | ISO/IEC 8824-4 and ITU-T X.691 | ISO/IEC 8825-2.

6 Basic biometric concepts

Traditionally, biometrics has been primarily limited to identity applications. However, biometrics literally means "life-measurements". Life-measurement applications far exceed identity applications and serve a greater need, which is not only to detect and quantify life, but also to protect life. Hence, biometrics is an essential component of biosafety and biosecurity applications. Attaching the prefix 'tele-' to biometrics further describes the telecommunication component required for networked and over-the-air (OTA) biometric applications.

As the market for autonomous systems grows so does their interaction with people, for example, autonomous vehicles and human interaction robots. Autonomous systems may leverage standardized guidelines to identify and protect life. The biology-to-machine (B2M) protocol is an interaction based language that supports telecommunication networks' ability to communicate between biological systems and Internet of Things (IoT) networks.

6.1 Biology-to-machine (B2M) Protocol

B2M expands the IoT to include not only electrically but also biologically based computation systems. B2M enables autonomous systems to recognize life and enable life-protecting measures, e.g., autonomous systems and medical devices. B2M also provides a common data format for mobile health devices independent of their manufacturer, e.g., wearables including but not limited to electrocardiogram (ECG or EKG) and continuous glucose monitors. B2M is designed to connect biological systems, which include but are not limited to people, livestock and plants, to computers and the network. B2M's purpose is to extend the IoT to include biological endpoints with a universal language and make B2M a native protocol to IoT devices globally.

Biological entities comprise complex computational actuator systems constructed of proteins, fats and carbohydrates. Since electronic systems attempt to emulate biological systems, we can conclude that both systems are computational entities, one organic and the other synthetic and much more primitive.

6.2 The telebiometric multimodal model

At the heart of the B2M protocol is a "telebiometric multimodal model" [b-ITU-T X.1081] which organizes all data collected or derived by the primary sciences: physics, chemistry, biology, culturology and psychology. A simple analogy equates the body as a territory and the primary scientific fields are the borders of the body. The aggregation of the following layers is what comprises the B2M protocol.

Primary science	Example
Physics	Temperature, ECG,
Chemistry	Blood test, urine Test, alcohol breath test, drug tests and blood oxygen saturation
Biology	DNA/protein, heart rate, sex and age
Culturology	Ethnic origin, nationality and religion
Psychology	Beliefs, psychiatric or personality disorders and syndromes

6.2.1 Interaction modality layer

Interaction modality per [b-ITU-T X.1081] is a layer that describes how an individual perceives an interaction, such as through touch, taste, sound and visually. Examples of interaction modalities across the biosphere are chemo-in (smell), audio-in, audio-out, etc. Music and speech are subcategories of audio. Gesture and facial expression are subcategories of video-out.

Biometric types	Interaction modality
Face image, iris, retina, hand geometry, vein pattern, finger image	Depends upon the technology used, video-in (input of artificial light is usually needed), video-ut, or tango-ut
Lip movement, thermal face image, thermal hand image, ear shape, finger geometry	Video-out
Voice	Audio-out
Signature dynamics, keystroke dynamics, footprint	Tango-out
Body Odour, DNA, blood and/or urine analysis	Chemo-out

6.2.2 Measurement system layer

By adding a measurement system layer such as the International System of Units (SI) including values and units, how the biological system was influenced by its environment can be quantified.

6.3 B2M packet construction

From this model (Figure 1), a data packet is structured containing essential information about the biological entity, hardware and the type of interaction. The data is extracted, managed and protected enhancing the value of the application by creating sets of raw data, derived data, and graphical data supported by deep learning algorithms. The datasets collected are considered biosignals. Biosignals flow to and from the biological entity. The B2M protocol enables an IoT network to bidirectionally communicate with a biological entity.

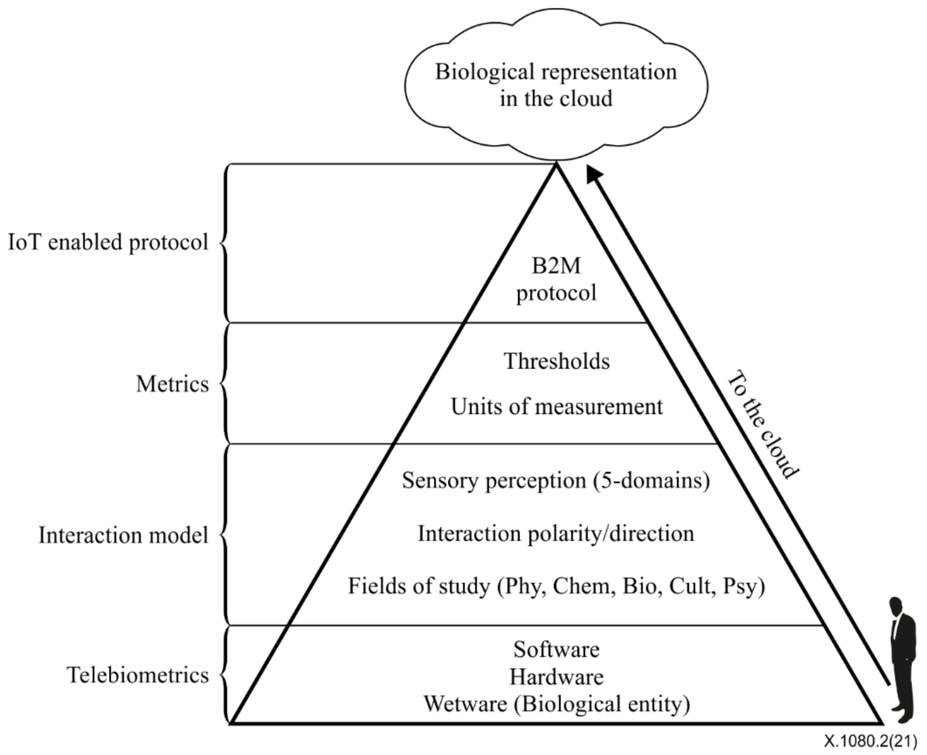


Figure 1 – The hierarchical components of the B2M model

B2M descriptors are included directly in machine-to-machine (M2M) protocols, thereby extending the IoT to include biological entities. The value of the B2M protocol is that it is application and market independent and supports unlimited devices, thereby increasing the biological representation in the cloud in a universally accessible and useful manner.

The B2M information model and exchange can be viewed from two complementary angles. Both are important:

a) A biosignal point of view, where the interaction between a patient and his or her environment is considered.

b) A pure protocol point of view, where the information generated according to item a) is transferred securely and unmodified from one point to another.

This clause is mostly concerned with the signal point of view, while protocol aspects are treated in clause 7.

The basic biosignal concept is to accurately describe the interaction(s) between a sensor or actuator and a biological entity, i.e., a person, animal or plant. A biosignal is either an IN or OUT interaction determined by the direction of the interaction (Figure 2).

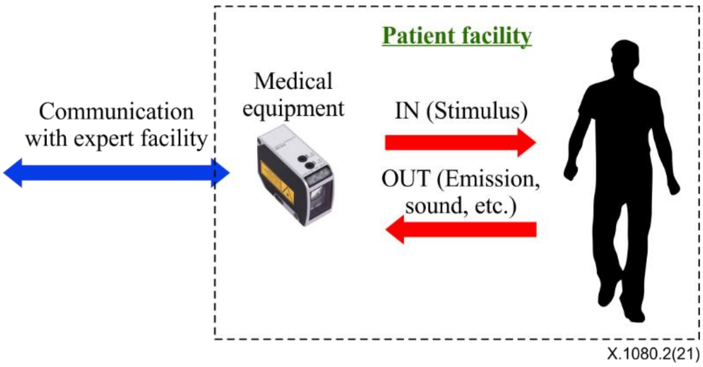
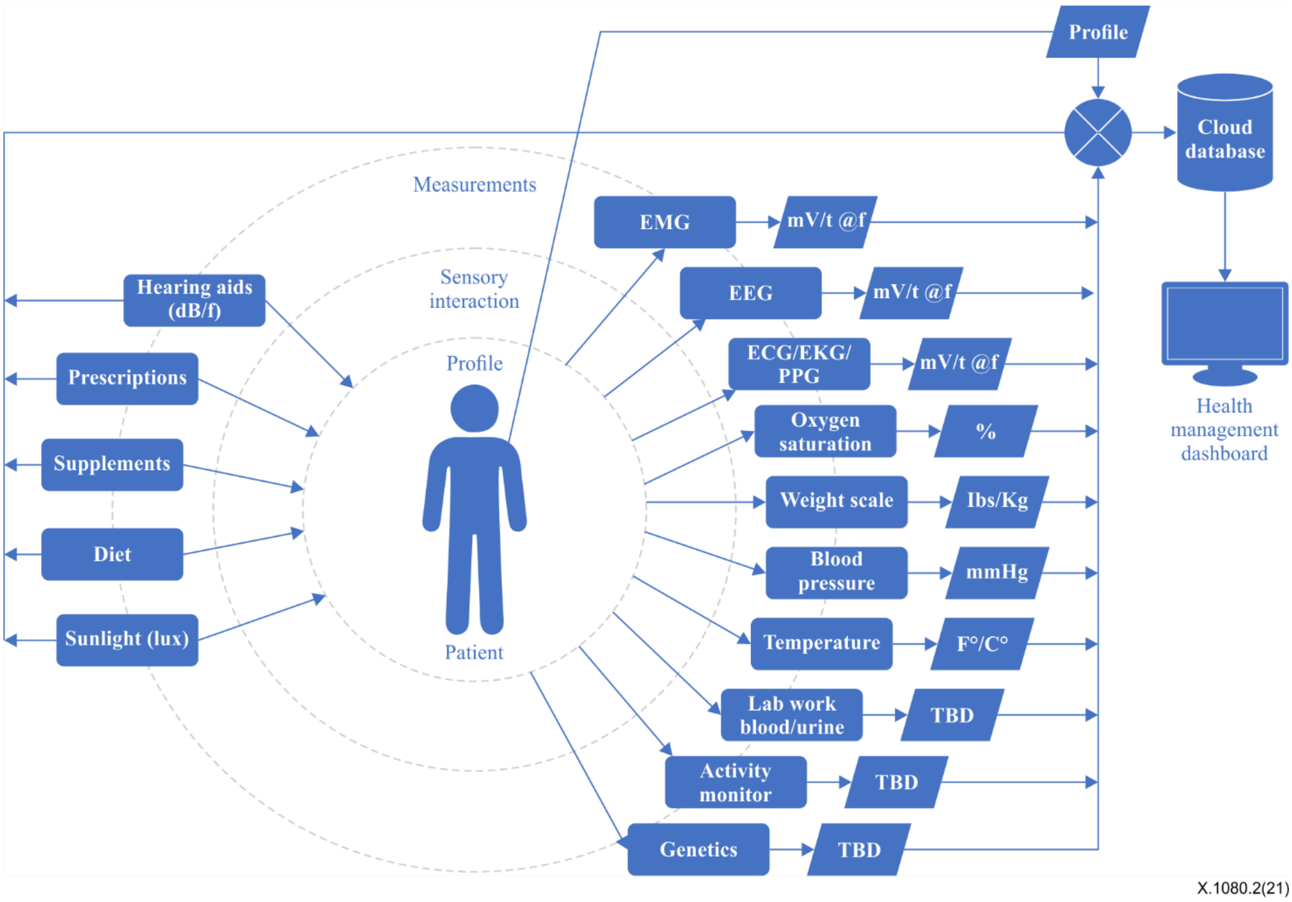


Figure 2 – IN and OUT interaction

The biological entity and sensor or actuator require an object identifier thereby enabling the postprocessing of biosignals searchable by their biological hierarchy and sensor or actuator type. The B2M protocol is structured at the biological endpoint/device and can postprocess to create new B2M protocol strands with the ability to transfer the B2M strand back to the network. For an ECG interaction example, a B2M packet may transfer mV/t (measurement system) as measured in the physics domain (scientific layer), whereas a postprocessed calculation of the ECG signal to determine heart rate is classified in the biology domain (scientific layer).



Key: EMG – electromyography; electroencephalography; PPG – photoplethysmogram;
TBD – to be defined

Figure 3 – Monitoring types and resulting diagnosis

In Figure 3, an array of B2M interaction devices demonstrates the types of measurements, interactions and direction of information to and from the patient. Each B2M string is searchable, postprocessable, and linked to the patient's identity. Parts of the information are searchable while maintaining the privacy of the patient. For example, information related to a global outbreak can be shared by government health agencies in real time without revealing the patient's identity.

7 Overview of the B2M protocol

7.1 Scope of B2M protocol

The B2M protocol is used to establish communication between a medical expert facility and a remote patient facility.

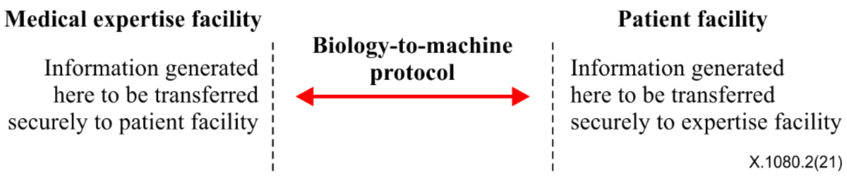


Figure 4 – The scope of the B2M protocol

Figure 4 illustrates the scope of the biology-to-machine protocol. Information is generated at one location and is transmitted to the other location. The details about how information is generated on the sender side and details about its processing by the recipient at the recipient side is outside the scope of this Recommendation. The main scope of the B2M protocol is:

a) To provide a versatile data structure that allows any relevant type of information to be mapped to that data structure.

b) To transfer the data so structured to the destination ensuring:

  – authentication, i.e., that the sender is properly authenticated;

  – integrity, i.e., that if data is modified during transmission, it is detected; and

  – optionally, confidentiality by use of encryption.

Details on how information is treated by the recipient is likewise outside the scope of this Recommendation. As an example, information received at the expert facility may be processed in many different ways, such as compared with locally available information, processed by non-standardized analysis systems, combined to establish diagnostics, etc.

7.2 Monitoring configuration

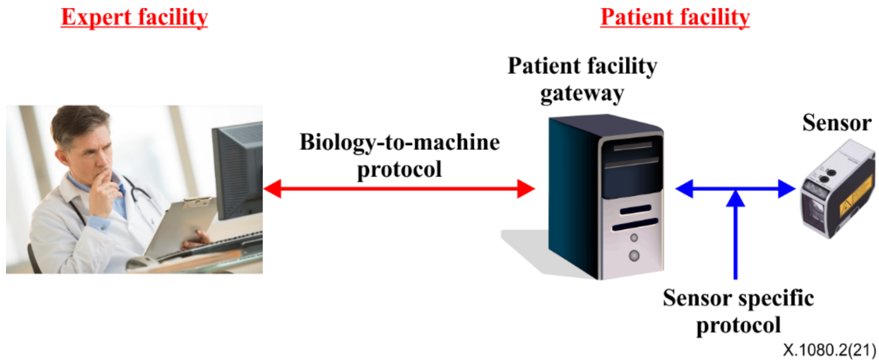


Figure 5 – Physical monitoring configuration

The patient facility is a system at a patient location that collects information about the patient, e.g., by using sensors of different types. The patient facility may be either in the patient's home or in a local medical centre.

The B2M protocol interacts with a gateway at the patient facility, which then provides the local mapping between the B2M protocol and the local interaction.

The medical expert facility may be either a centre with a medical staff or surveillance equipment. The medical expert facility uses the B2M protocol to instruct the patient facility gateway to operate a sensor in a particular way and to request specific information.

The protocol between the patient facility gateway and a sensor is determined by the type of sensor and may be a proprietary protocol. There may not be a one-to-one relationship between the B2M protocol elements and the protocol elements of a sensor specific protocol. The mapping between the sensor specific protocol and the B2M protocol is outside the scope of this Recommendation.

7.3 Use of security services

This Recommendation makes use of the security services of the wrapper protocol, as defined by [ITU-T X.510]. The wrapper protocol provides for authentication, integrity and optionally confidentiality by use of encryption. It also provides a migration path for cryptographic algorithms, e.g., when stronger cryptographic algorithms are required. This allows migration to quantum-safe algorithms in the future without affecting the B2M protocol.

[ITU-T X.510] performs a handshake exchange to establish an association between two entities before the actual data transfer phase may commence. The B2M protocol requires initialization information to be carried in this handshake.

[ITU-T X.510] also defines some terms used by this Recommendation. An application protocol data unit (APDU) specified by this Recommendation is called a protected protocol data unit (PrPDU), as it is protected by the wrapper protocol. When wrapped within the wrapper protocol, the resulting application protocol data unit is called a wrapper protocol data unit (WrPDU). A WrPDU may not in all cases include a PrPDU.

7.4 Initialization of the B2M protocol

[ITU-T X.510] requires the establishment of an association between two communicating entities by a handshake exchange. An association is a cooperative relationship between two entities, which enables the communication of information and the coordination of their joint operation for an instance of communication. To start a monitoring session, an association shall be established between the medical expert facility and the patient facility.

The initiator of an association is called the client, while the target entity is called the server.

This Recommendation does not specify whether the expert facility or the patient facility initiates the association. Any side may initiate an association. However, restriction may be imposed by means outside this Recommendation.

During the association establishment, [ITU-T X.510] allows the client to add B2M protocol initialization parameters for the server to accept or reject (see clause 9.2).

When an association is established, the two communicating entities enter the data transfer phase.

7.5 The data transfer phase

The type of information to be transferred between a medical expert facility and a patient facility depends on several factors, such as:

a) the type of patient to be monitored;

b) local policies; and

c) the facilities available.

The number of information types to be transferred is almost unbounded and new information types will constantly be defined. It is therefore important that the B2M protocol is flexible enough to carry any information type that fits into one of the information models specified in SECTION 2. Extended and new information models may be developed in future editions of this Recommendation.

7.6 Data types

7.6.1 General

This Recommendation uses a flexible concept of data types that assigns object identifiers to data types allowing the B2M protocol to transfer different kinds of information without having the information types hard coded in the basic protocol.

7.6.2 The concept of monitor types

A monitoring session may involve several monitor types. Monitor types need to be defined to allow the expert facility and the patient facility to negotiate what monitor types to activate during a monitoring session. Movement detection (see clause 12) is an example of a monitor type.

A monitor type is identified by an object identifier. Any organization that is authorized to allocate object identifiers may define new monitor types following the principles set out by this Recommendation. It is recommended that monitor types be defined by standard development organizations (SDOs) and made publicly available.

7.6.3 General information items

A general information item specifies non-medical information associated with a patient, such as name or personal identification number.

This Recommendation does not define types for all general information items that may be required for a specific environment. Future editions of this Recommendation may add new general information item specifications. Other international standards and recommendations may define their own general information items to be transferred by the B2M protocol. An object identifier is allocated to each type of general information item.

7.6.4 Device information

Beyond data items, there is a need for another simple type of information. Devices and sensors at the patient location need to be identified for the purpose of selecting devices for monitoring and for associating additional information to monitoring information.

8 The information model

8.1 Introduction to the information model

The information model is specified using the ASN.1 information object classes concept defined in [ITU-T X.681]. [ITU-T X.510] gives a short introduction to the ASN.1 information object class concept. The use of information object classes is a flexible way to specify information types without hard coding the specific instances of information types.

An information object class is the basis for defining an information object of a specific type. Such an information object is identified by an object identifier. An example of an information object could be a **`surname`** information object, instances of which may be carried in the protocol.

The use of the information object class concept has the advantage that other organizations, e.g., standards organization may define new information objects based on the information object class specifications within this Recommendation.

8.2 Useful data types

```
Direction ::= ENUMERATED {in(0), out(1)}
```

The **`Direction`** data type may be inserted at specific places to indicate whether the interaction with a living body is towards the body (in) or away from the body (out). See Figure 2.

```
DeviceInfo ::= SEQUENCE {
  vendor [0] UTF8String OPTIONAL,
  type   [1] UTF8String OPTIONAL,
  id     [2] UTF8String OPTIONAL,
  ... }
```

8.3 The MONITOR-TYPE information object class

A monitor type is specified using the **`MONITOR-TYPE`** information object class defined in [ITU-T X.681]. It binds an abstract syntax and an identifying object identifier.

```
MONITOR-TYPE ::= TYPE-IDENTIFIER
```

8.4 The OPTIONS information object class

The **`OPTION`** information object class is used for specifying options for voice and/or video, where such options are dependent on the type of equipment. It is assumed that such a specific option can be specified by a specific data type.

```
OPTIONS ::= CLASS {
  &Voice-options   OPTIONAL,
  &Video-options   OPTIONAL }
WITH SYNTAX {
  [VOICE OPTIONS   &Voice-options]
  [VIDEO OPTIONS   &Video-options] }
```

The **`OPTIONS`** information object class has the following fields:

a) The **`&Voice-options`** optional field is used for specifying a data type for the options for the voice equipment.

b) The **`&Video-options`** optional field is used for specifying a data type for the options for the video equipment.

8.6 The UNIT information object class

The **`UNIT`** information object class is used to specify units as defined in [b-BIPM].

```
UNIT ::= CLASS {
  &name       PrintableString,
  &symbol     PrintableString,
  &quantity   PrintableString OPTIONAL,
  &Value,
  &id         OBJECT IDENTIFIER UNIQUE }
WITH SYNTAX {
  NAME        &name
  SYMBOL      &symbol
  [QUANTITY   &quantity]
  VALUE       &Value
  ID          &id }
```

The **`UNIT`** information object class has the following fields:

a) The **`&name`** field is used for specifying the name for the unit.

b) The **`&symbol`** field is used for specifying the symbol for the unit.

c) The **`&quantity`** field is used for specifying the type of quantity for which this unit type is used.

d) The **`&Value`** field is used to specify the ASN.1 data type for the information objects of that information object class.

e) The **`&id`** field is used to specify the object identifier identifying the type of unit.

An instance of a **`UNIT`** information object has the following general syntax:

```
Unit{UNIT:SupportedUnits} ::= SEQUENCE {
  id             UNIT.&id({SupportedUnits}),
  name           UNIT.&name({SupportedUnits}{@id}),
  symbol         UNIT.&symbol({SupportedUnits}{@id}),
  quantity   [0] UNIT.&quantity({SupportedUnits}{@id}) OPTIONAL,
  value      [1] UNIT.&Value }

SupportedUnits UNIT ::= {...}
```

Clause 10 specifies some unit information objects.

8.7 The GEN-INFO information object class

The **`GEN-INFO`** information object class is used to specify general, non-medical information, such as patient name or address.

```
GEN-INFO ::= CLASS {
  &Type,
  &id      OBJECT IDENTIFIER UNIQUE }
WITH SYNTAX {
  SYNTAX  &Type
  ID      &id }
```

The **`&Type`** field is used for specifying the syntax of the defined data item. This shall be an ASN.1 type.

The **`&id`** field is used for specifying an object identifier identifying the type of data item.

Clause 11 specifies some data item information objects.

The syntax of a general information data item shall be as indicated by the following:

```
Gen-Info {GEN-INFO:SupportedGenInfo} ::= SEQUENCE {
  type      GEN-INFO.&id({SupportedGenInfo}),
  value     GEN-INFO.&Type({SupportedGenInfo}{@type}),
  ... }

SupportedGenInfo GEN-INFO ::= { ... }
```

9 Formal specification of the biometric to machine protocol

9.1 Top-level B2M protocol

```
B2M-protocol ::= CHOICE {
  b2mInitReq    [0] B2MInitReq,
  b2mInitAcc    [1] B2MInitAcc,
  b2mInitRej    [2] B2MInitRej,
  b2mInitAbt    [3] B2MInitAbt,
  b2mTranf      [4] B2MDataTransfer,
  ... }
```

9.2 Initialization of a monitoring session

9.2.1 Initialization request

The **B2MInitReq** PrPDU shall be included in a **HandshakeReq** WrPDU of the wrapper protocol.

```
B2MInitReq ::= SEQUENCE {
  version            Version,
  requirements       ENUMERATED {
    none                                (0),
    with-2-way-voice                    (1),
    with-reverse-video-and-2-way-voice (2),
    with-2-way-video-and-voice          (3),
    ... } DEFAULT none,
  voice-options [0] OPTIONS.&Voice-options OPTIONAL,
  video-options [1] OPTIONS.&Video-options OPTIONAL,
  monitorTypes       SEQUENCE SIZE (1..MAX) OF MONITOR-TYPE.&id
({SupportedMonitorTypes}),
  ... }

Version ::= BIT STRING {
  v1 (0)  -- version 1
  }
```

The **B2MInitReq** has the following components:

a) The **version** component shall specify the version(s) of the B2M protocol supported by the client. The **version** component is a bit string that allows the client to set multiple bits, if it supports multiple versions.

NOTE – Currently, only version 1 is defined.

b) The **requirements** component, when present, shall specify voice and video requirements. If this component is absent, it defaults to **none**.

c) The **voice-options** component, when present, shall specify the data type used for specifying the voice options. This component may be present if the **&requirements** field does not specify **none**. Otherwise, it shall be absent.

d) The **video-options** component, when present, shall specify the data type used for specifying the video options. This component may be present if the **requirements** component specifies **with-reverse-video-and-2-way-voice** or **with-2-way-video-and-voice**. Otherwise, it shall be absent.

e) The **monitorTypes** component shall specify what monitor types the client is proposing.

9.2.2 Initialization accept

The **B2MInitAcc** PrPDU shall be used by the server to accept an association establishment. It shall be included in a **HandshakeAcc** WrPDU of the wrapper protocol.

```
B2MInitAcc ::= SEQUENCE {
  version       Version,
  monitorTypes  SEQUENCE SIZE (1..MAX) OF MONITOR-TYPE.&id OPTIONAL,
  ... }
```

The **B2MInitAcc** PrPDU has the following components:

a) The **version** component shall specify exactly one version that is supported by the server. It shall be selected from those suggested in the corresponding **B2MInitReq** PrPDU. The highest supported version of those suggested should be selected.

b) The **monitorTypes** component shall hold a list of those monitor types suggested in the **B2MInitReq** PPDU that the server supports.

9.2.3 Initialization reject

The **B2MInitRej** PrPDU shall be used by the server to reject an association establishment. It shall be included in a **HandshakeProRej** WrPDU of the wrapper protocol.

```
B2MInitRej ::= SEQUENCE {
  version Version,
  initErr InitError,
  ... }

InitError ::=  ENUMERATED {
  initiation-not-allowed       (0),
  versions-not-supported       (1),
  voice-option-not-available   (2),
  video-option-not-available   (3),
  monitor-types-not-supported (4),
  ... }
```

The **B2MInitRej** has the following components:

a) The **version** component shall specify exactly one version that is supported by the server. It should be selected from those suggested in the corresponding **B2MInitReq** PrPDU. The highest supported version of those suggested should be selected. However, if the server does not support any of the suggested versions, it shall return one alternative value it does support.

b) The **initErr** component shall indicate the reason for rejection by selecting one of the following diagnostic codes:

   – The **initiation-not-allowed** diagnostic code shall be selected if the server does not accept an incoming association request.

   – The **versions-not-supported** diagnostic code shall be selected if the server does not support any of the versions suggested in the **B2MInitReq** PrPDU.

   – The **voice-option-not-available** diagnostic code shall be selected if the client requested voice option, but that is not supported by the server.

   – The **video-option-not-available** diagnostic code shall be selected if the client requested video option, but that is not supported by the server.

   – The **monitor-types-not-supported** diagnostic code shall be selected if the server does not support any of the monitor types suggested in the **B2MInitReq** PrPDU.

9.2.4 Initialization abort

The **B2MInitAbt** PrPDU shall be used by the client in reaction to a received **B2MInitAcc** PrPDU deemed invalid. It shall be carried by a **HandshakeProAbort** WrPDU of the wrapper protocol.

```
B2MInitAbt ::= SEQUENCE {
  version Version,
  accErr  AccError,
  ... }

AccError ::= ENUMERATED {
  only-one-version               (0),
  unexpected-version             (1),
  unexpected-monitorTypes        (2),
  critical-monitorTypes-missing (3),
  ... }
```

The **B2MInitAbt** has the following components:

a) The **version** component shall have the same bits set as in the original **B2MInitAcc** PrPDU.

b) The **accErr** component shall indicate the reason for the abort by the selection of one of the following diagnostic codes:

- The **only-one-version** diagnostic code shall be selected if the **version** component of the **B2MInitAcc** PrPDU has bits set for more than one version.
- The **unexpected-version** diagnostic code shall be selected if the **version** component of the **B2MInitAcc** PrPDU specifies a version not suggested by the client.
- The **unexpected-monitorTypes** diagnostic code shall be selected if the **monitorTypes** component of the **B2MInitAcc** PrPDU specifies one or more monitor types not suggested by the client.
- The **critical-monitorTypes-missing** diagnostic code shall be selected if the **B2MInitAcc** PrPDU excludes one or more monitor types considered critical by the client.

9.3 Data transfer

The **B2MDataTransfer** PrPDU shall be carried by a **DataTransferReq** or **DataTransferAcc** WrPDU of the wrapper protocol.

```
B2MDataTransfer ::= SEQUENCE {
  time              GeneralizedTime,
  genInfos      [0] SEQUENCE SIZE (1..MAX) OF CHOICE {
    genInfoType     GEN-INFO.&id({SupportedGenInfo}),
    genInfo         Gen-Info{{SupportedGenInfo}},
    ... } OPTIONAL,
  monTypes    [1] SEQUENCE SIZE (1..MAX) OF SEQUENCE SIZE (1..MAX) OF SEQUENCE {
    deviceInfo  [1] DeviceInfo OPTIONAL,
    mon         [2] MONITOR-TYPE.&Type ({SupportedMonitorTypes}) OPTIONAL,
    ... },
  ... }

InvokeID ::= INTEGER (0..MAX)
```

The **B2MDataTransfer** PrPDU has the following components:

a) The **time** component shall hold the time of the generation of the PrPDU.

b) The **genInfos** components shall hold one or more types of general information.

9.4 Allocation of object identifier arcs

The following object identifier arcs defined by Annex A of [ITU-T X.1080.0] are relevant for this Recommendation:

a) The following arc is allocated to telebiometrics:

```
id-telebio OBJECT IDENFIER ::= { joint-iso-itu-t(2) telebiometrics(42) }
```

b) Below this arc, [ITU-T 1081] allocates the following arc for telehealth:

```
id-thprot OBJECT IDENTIFIER ::= { id-telebio thprot(10) }
```

c) The following arc is allocated to this Recommendation:

```
id-x1080-2 OBJECT IDENTIFIER ::= { id-thprot part2(2) }
```

The following arcs are allocated below the **id-x1080-2** arc:

a) The following arc is allocated to B2M ASN.1 modules:

```
id-x1080-2-modules OBJECT IDENTIFIER ::= { id-x1080-2 modules(0) }
```

b) The following arc is allocated to monitor types:

```
id-mt OBJECT IDENTIFIER ::= { id-x1080-2 monitor-type(1) }
```

c) The following arc is allocated to unit type specifications:

```
id-un OBJECT IDENTIFIER ::= { id-x1080-2 unit(2) }
```

d) The following arc is allocated to general information type specifications:

```
id-gi OBJECT IDENTIFIER ::= { id-x1080-2 gen-info(3) }
```

10 Unit information objects

10.1 Introduction

The SI units are defined as an information object of the **UNIT** ASN.1 information object class. This allows a value to be carried in the protocol together with the identifying object identifier.

10.2 The seven defining constants of the SI

10.2.1 General

All SI units are defined in terms of constants that describe the natural world. This will assure the future stability of the SI and open the opportunity for the use of new technologies, including quantum technologies, to implement the definitions.

The seven defining constants of the SI are defined in separate subclauses below.

10.2.2 The hyperfine transition frequency of Caesium

Exactly accurate clocks can be constructed by locking an electronic oscillator to the frequency of an atomic transition. The frequencies associated with such transitions are so reproducible that the definition of the second is now tied to the frequency associated with a transition in caesium-133.

The Caesium-133 atom in isolation at absolute zero temperature emits radiant energy, or light, in the form of microwaves at a very precise frequency. A period (circle) is the time constant used for defining a second (see clause 10.3.3).

10.2.3 Speed of light in vacuum

The speed of light in vacuum, *c*, is a universal physical constant used to define a metre. As a metre is defined as the distance light can travel in 1/299 792 458 of a second (see clause 10.3.1), by definition, the speed of light is then 299 792 458 m/s. In theory, nothing travels faster than light.

10.2.4 Planck constant

The Planck constant, *h*, links a particle's frequency with its total energy by specifying that the energy is equal to the Planck constant times frequency. As frequency is the inverse of time, the Planck constant is energy multiplied by time. Its value is $h = 6.626\ 070\ 15 \times 10^{-34}$ joules per second (J s) or as kg m^2 s^{-1} (see clause 10.4.5).

10.2.5 Elementary charge

The elementary charge, *e*, is the electrical charge carried by a single electron. This is equivalent but opposite in polarity to the electrical charge carried by a proton. By convention, electrons have negative (minus) charge, and protons have positive (plus) charge. The elementary charge is equal to $1.602\ 176\ 634 \times 10^{-19}$ C (see clause 10.4.7).

10.2.6 Boltzmann constant

The Boltzmann constant, k_B, is the relationship between absolute temperature and the kinetic energy contained in each molecule of an ideal gas.

Its value is $k = 1.380\ 649 \times 10^{-23}$ J·K^{-1}. The average kinetic energy per molecule is $k \times$ T (ktemperature given in degrees kelvin). Kinetic energy is the energy an object has because of its motion.

10.2.7 Avogadro constant

The Avogadro constant, N_A, specifies the number of particles in a mole (see clause 10.3.6) and its value is $6.022\ 140\ 76 \times 10^{23}\ \text{mol}^{-1}$.

10.2.8 Luminous efficacy

Luminous efficacy describes how well a light source emits visible light and is the ratio of luminous flux to power. The luminous efficacy of monochromatic radiation of a frequency of 540 THz is equal to 683 lm/W.

10.3 The seven base units

10.3.1 The metre as an SI base unit

The metre is the SI unit of length.

```
metre UNIT ::= {
  NAME     "metre"
  SYMBOL   "m"
  QUANTITY "length"
  VALUE    REAL
  ID       id-un-metre }
```

One metre is the length light in vacuum travels in 1/299 792 458 of a second, which corresponds to light travelling at a speed of 299 792 458 metres per second.

10.3.2 The kilogram as an SI base unit

The kilogram is the SI base unit of mass.

```
kilogram UNIT ::= {
  NAME     "kilogram"
  SYMBOL   "kg"
  QUANTITY "mass"
  VALUE    REAL
  ID       id-un-kilogram }
```

The kilogram is defined by three fundamental constants: the speed of light, c, a specific atomic transition frequency, $\Delta\nu_{Cs}$, and the Planck constant, h.

10.3.3 The second as an SI base unit

The second is the SI base unit of time.

```
second UNIT ::= {
  NAME     "second"
  SYMBOL   "s"
  QUANTITY "time"
  VALUE    REAL
  ID       id-un-second }
```

A second is equal to the duration of 9 192 631 770 periods of the frequency of the caesium-133 atom.

10.3.4 The ampere as an SI base unit

Ampere is the SI unit of electric current.

```
ampere UNIT ::= {
  NAME     "ampere"
  SYMBOL   "A"
  QUANTITY "electric-current"
  VALUE    REAL
  ID       id-un-ampere }
```

An ampere is equal to the flow of 1/(1.602 176 634 × 10^{-19}) elementary charges per second.

10.3.5 The kelvin as an SI base unit

The kelvin is the SI base unit of thermodynamic temperature.

```
kelvin UNIT ::= {
  NAME     "kelvin"
  SYMBOL   "K"
  QUANTITY "thermodynamic-temperature"
  VALUE    REAL
  ID       id-un-kelvin }
```

10.3.6 The mole as an SI base unit

The mole is the unit of measurement for amount of substance.

```
mole UNIT ::= {
  NAME     "mole"
  SYMBOL   "mol"
  QUANTITY "amount-of-substance"
  VALUE    REAL
  ID       id-un-mole }
```

A mole is defined as exactly 6.022 140 76 × 10^{23} particles, which may be atoms, molecules, ions or electrons.

10.3.7 The candela as an SI base unit

The candela is the SI unit of luminous intensity in a given direction.

```
candela UNIT ::= {
  NAME     "candela"
  SYMBOL   "cd"
  QUANTITY "luminous-intensity"
  VALUE    REAL
  ID       id-un-candela }
```

10.4 SI derived units

10.4.1 Introduction

All other units, described as derived units, are constructed as products of one or more of the base units and in many cases are scaled by exponentiation.

10.4.2 The hertz as an SI derived unit

The hertz is the derived unit of frequency and is defined as one cycle per second.

```
hertz UNIT ::= {
  NAME    "hertz"
  SYMBOL  "Hz"
  VALUE   REAL
  ID      id-un-hertz }
```

The hertz is derived from second and is equal to s^{-1}.

10.4.3 The newton as an SI derived unit

The newton is the derived unit of force and is the force needed to accelerate one kilogram one metre per second squared. It may also be defined as the change in velocity per second.

```
newton UNIT ::= {
  NAME    "newton"
  SYMBOL  "N"
```

```
  VALUE   REAL
  ID      id-un-newton }
```

The newton expressed in SI base units is kg m s^{-2}.

10.4.4 The pascal as an SI derived unit

Pascal is the derived unit of pressure and is defined as one newton per square metre.

```
pascal UNIT ::= {
  NAME    "pascal"
  SYMBOL  "Pa"
  VALUE   REAL
  ID      id-un-pascal }
```

The pascal expressed in SI base units is kg m^{-1} s^{-2}.

10.4.5 The joule as an SI derived unit

The joule is a derived unit of energy.

```
joule UNIT ::= {
  NAME    "joule"
  SYMBOL  "J"
  VALUE   REAL
  ID      id-un-joule }
```

The joule expressed in SI base units is kg·m^2·s^{-2}.A joule is equal to the work required to produce one watt for one second, which means that one kWH is 3 600 000 joules or 3.6 megajoules. A joule is also equal to the force of one newton acting on an object in the direction of the force’s motion for one metre.

10.4.6 The watt as an SI derived unit

The watt is a derived unit of power or radiant flux and is equivalent to joules (energy) per second.

```
watt UNIT ::= {
  NAME    "watt"
  SYMBOL  "W"
  VALUE   REAL
  ID      id-un-watt }
```

The joule expressed in SI base units is kg·m^2·s^{-3}.

10.4.7 The coulomb as an SI derived unit

The coulomb is the derived unit of electric charge.

```
coulomb UNIT ::= {
  NAME    "coulomb"
  SYMBOL  "C"
  VALUE   REAL
  ID      id-un-coulomb }
```

The flow of one ampere for one second is an electric charge of one coulomb. As for the official SI definition, a coulomb is the amount of electricity of exactly $1/(1.602\ 176\ 634 \times 10^{-19})$ elementary charges or approximately $6.241\ 509\ 074\ 4 \times 10^{18}$ elementary charges (or 1.036×10^{-5} mol).

In SI base units, the coulomb is expressed as A s.

10.4.8 The volt as an SI derived unit

The volt is the derived unit of electric potential difference.

```
volt UNIT ::= {
```

```
  NAME    "volt"
  SYMBOL  "V"
  VALUE   REAL
  ID      id-un-volt }
```

One volt is defined as the difference in electric potential across a wire when an electric current of one ampere dissipates one watt of power.

A volt can also be defined as the potential difference between two points in an electric circuit that will impart one joule (J) of energy per coulomb (C) of charge that passes through it.

In SI base units, the volt is expressed as kg m^2 s^{-3} A^{-1}.

10.4.9 The farad as an SI derived unit

The farad is the derived unit of electrical capacitance.

```
farad UNIT ::= {
  NAME    "farad"
  SYMBOL  "F"
  VALUE   REAL
  ID      id-un-farad }
```

In SI base units, the farad is expressed as kg^{-1} m s^4 A^2.

10.4.10 The ohm as an SI derived unit

The ohm is the derived unit of electrical resistance.

```
ohm UNIT ::= {
  NAME    "ohm"
  SYMBOL  "Ω"
  VALUE   REAL
  ID      id-un-ohm }
```

In SI base units, ohm is expressed as kg m^2 s^{-3} A^{-2}.

10.4.11 The siemens as an SI derived unit

The siemens is the derived unit of electric conductance and is the reciprocal of electric resistance.

```
siemens UNIT ::= {
  NAME    "siemens"
  SYMBOL  "S"
  VALUE   REAL
  ID      id-un-siemens }
```

In SI base units, the siemens is expressed as kg^{-1} m^{-2} s^3 A^2.

10.4.12 The weber as an SI derived unit

The weber is the derived unit of magnetic flux.

```
weber UNIT ::= {
  NAME    "weber"
  SYMBOL  "Wb"
  VALUE   REAL
  ID      id-un-weber }
```

In SI base units, the weber is expressed as kg m^2 s^{-2} A^{-1}.

10.4.13 The tesla as an SI derived unit

The tesla is the derived unit of magnetic flux density.

```
tesla UNIT ::= {
  NAME    "tesla"
```

```
  SYMBOL  "T"
  VALUE   REAL
  ID      id-un-tesla }
```

In SI base units, the tesla is expressed as kg s^{-2} A^{-1}.

10.4.14 The henry as an SI derived unit

The henry is the derived unit of electric inductance.

```
henry UNIT ::= {
  NAME    "henry"
  SYMBOL  "H"
  VALUE   REAL
  ID      id-un-henry }
```

In SI base units, the henry is expressed as kg m^2 s^{-2} A^{-2}.

10.4.15 The degree Celsius as an SI derived unit

The degree Celsius is a derived unit of temperature.

```
degreeCelsius UNIT ::= {
  NAME    "degreeCelsius"
  SYMBOL  "°C"
  VALUE   REAL
  ID      id-un-degreeCelsius }
```

The temperature in °C is expressed as $t = T - T_0$, where T is the kelvin (absolute) temperature and $T_0 = 273.15$.

10.4.16 The lumen as an SI derived unit

The lumen is the derived unit of luminous flux, i.e., the quantity of visible light emitted by a source per unit of time.

```
lumen UNIT ::= {
  NAME    "lumen"
  SYMBOL  "lm"
  VALUE   REAL
  ID      id-un-lumen }
```

In SI base units, the lumen is expressed as cd sr.

10.4.17 The lux as an SI derived unit

The lux is the derived unit of illuminance. It is used as a measure of intensity.

```
lux UNIT ::= {
  NAME    "lux"
  SYMBOL  "lx"
  VALUE   REAL
  ID      id-un-lux }
```

In SI base units, the lux is expressed as cd sr m^{-2}.

10.4.18 The becquerel as an SI derived unit

The becquerel is the derived unit of radioactivity.

```
becquerel UNIT ::= {
  NAME    "becquerel"
  SYMBOL  "Bq"
  VALUE   REAL
  ID      id-un-becquerel }
```

In SI base units, the becquerel is expressed as s^{-1}.

11 Personal and identifying general information items

11.1 General

This clause defines general information items to be used for identifying a patient and for adding non-medical information, such as telephone number or postal address.

If the value of a general information item together with the associated object identifier is included in a data transfer, it provides relevant information for the recipient. If only the object identifier is included, it is a signal to the recipient that it is supposed to return the associated information.

11.2 Surname general information item

A value of the type **surname** specifies the linguistic construct of one or more words which, in cultures where it is used to differentiate individuals, often indicates familial relationships.

```
surname GEN-INFO::= {
  SYNTAX   UTF8String
  ID       id-di-surname }
```

11.3 Given name general information item

A value of the type **givenName** specifies the linguistic construct which is normally given to an individual by the individual's parent, or is chosen by the individual, or by which the individual is commonly known.

```
givenName GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-givenName }
```

11.4 Initials general information item

A value of the type **givenName** specifies the initials of some or all of an individual's names, but not the surname(s).

A data value item for Initials is a string, e.g., "D" or "D." or "J.P.".

```
initials GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-initials }
```

11.5 Generation qualifier general information item

A value of the type **generationQualifier** specifies the generation qualifier used to provide generation information to qualify an individual's name.

A data value for Generation Qualifier is a string, e.g., "Jr." or "II".

```
generationQualifier GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-generationQualifier }
```

11.6 Serial number general information item

A value of the type **serialNumber** specifies an identifier or the serial number of an object.

```
serialNumber GEN-INFO ::= {
  SYNTAX   UTF8String
  id       id-di-serialNumber }
```

11.7 Pseudonym general information item

A value of the type **pseudonym** specifies a pseudonym for an object. It is used for naming an object when it is to be made clear that its name is a pseudonym.

```
pseudonym GEN-INFO ::= {
  SYNTAX   UTF8String
  id       id-di-pseudonym
```

11.8 Uniform resource identifier (URI) general information item

A value of the type **uri** specifies a uniform resource identifier (URI) as defined in [b-RFC 3986].

```
uri GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-uri }
```

11.9 Uniform resource name (URN) general information item

A value of the type **urn** specifies a uniform resource name (URN) as defined in [b-RFC 3406].

```
urn GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-urn }
```

11.10 Uniform resource locator (URL)general information item

A value of the type **url** specifies a uniform resource locator (URL).

```
url GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-url }
```

11.11 Domain name general information item

A value of the type **dnsName** specifies a DNS domain name, which may be an internationalized domain name (IDN).

```
dnsName GEN-INFO ::= {
  SYNTAX   DomainName
  ID       id-di-dnsName }

DomainName ::= UTF8String (CONSTRAINED BY
  { -- Conforms to the format of a (internationalized) domain name. -- })
```

A value of the **DomainName** data type shall be in the syntax specified by clause 2.3.1 of [RFC 5890], meaning that a domain name is a sequence of labels in the letters, digits, hyphen (LDH) format separated by dots.

A label may be in three formats:

a) All characters in the label are from the Basic Latin collection as defined by [b-ISO/IEC 10646] (i.e., with code points in the ranges 002D, 0030-0039, 0041-005A and 0061-007A) and not starting with "xn--". The maximum length is 63 octets.

b) It is an A-label as defined in [RFC 5890], i.e., it starts with the "xn--" and is a U-label converted to valid ASCII characters as in item a) using the Punycode algorithm defined by [b-RFC 3492]. The converted string shall be maximum 59 octets. To be valid, it shall be possible for an A-label to be converted to a valid U-label.

   NOTE – An A-label is normally not human readable.

c) It is a U-label as defined in [RFC 5890], i.e., it contains characters outside the Basic Latin collection. A valid U-label shall not include any characters that are not included in the restricted Unicode repertoire as defined by [b-RFC 5892] and it shall be convertible to a valid A-label as defined in item b). A valid U-label may be more than 63 octets.

11.12 E-mail general information item

A value of the type email specifies an email address associated with a patient.

```
email GEN-INFO ::= {
  SYNTAX   IA5String
  ID       id-di-email }
```

11.13 Country name general information item

A value of the type `countryName` specifies a country.

The country name is a string chosen from [b-ISO 3166-1] alpha-2.

```
countryName GEN-INFO ::= {
  SYNTAX   CountryName
  ID       id-di-countryName }

CountryName ::= PrintableString(SIZE (2))
  (CONSTRAINED BY { -- ISO 3166-1 alpha-2 codes only -- })
```

11.14 Country name with three characters general information item

A value of `countryCode3a` attribute type specifies a country.

A value for `countryCode3a` is a string chosen from [b-ISO 3166-1] alpha-3.

```
countryCode3c GEN-INFO ::= {
  SYNTAX   CountryCode3c
  ID       id-di-countryCode3c }

CountryCode3c ::= PrintableString(SIZE (3))
  (CONSTRAINED BY { -- ISO 3166-1 alpha-3 codes only -- })
```

11.15 Locality name general information item

This general information item identifies a locality. A locality name is a string, e.g., "Geneva".

```
localityName GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-localityName }
```

11.16 Street address general information item

The street address specifies a site for local distribution and physical delivery in a postal address, i.e., the street name, square or avenue.

A street address is a string, e.g., "Arnulfstraße".

```
streetAddress GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-streetAddress }
```

11.17 House identifier general information item

The house identifier is a linguistic construct used to identify a particular building, for example, a house number or house name relative to a street, avenue, town or city etc.

A value for house identifier is a string, e.g., "14".

```
houseIdentifier GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-houseIdentifier }
```

11.18 Universal Transverse Mercator (UTM)coordinates general information item

A **utmCoordinates** value gives coordinates in the Universal Transverse Mercator (UTM) coordinate system.

```
utmCoordinates GEN-INFO ::= {
  SYNTAX   UtmCoordinates
  ID       id-di-utmCoordinates }

UtmCoordinates ::= SEQUENCE {
  zone      PrintableString,
  easting   NumericString,
  northing  NumericString }
```

The **zone** component gives the value of the UTM zone. It consists of a single letter followed by up to two numeric characters.

The **easting** component gives the easting values in metres.

The **northing** component gives the northing value in metres.

11.19 Organization name general information item

An organization name value specifies an organization. It identifies an organization with which the named object is affiliated.

An organization name is a string chosen by the organization (e.g., O = "Scottish Telecommunications plc").

```
organizationName GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-organizationName }
```

11.20 Organizational unit name general information item

An organizational unit name specifies an organizational unit.

The designated organizational unit is understood to be part of an organization identified by an **organizationName** value.

An organizational unit name is a string chosen by the organization of which it is part (e.g., "Technology Division").

```
organizationalUnitName GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-organizationalUnitName }
```

11.21 Title general information item

A title specifies the designated position or function of the object within an organization.

```
title GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-title }
```

11.22 Organization identifier general information item

An organization identifier contains an identification of an organization that is different from the organization name.

```
organizationIdentifier GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-organizationIdentifier }
```

11.23 Description general information item

A description specifies text that describes the associated patient.

```
description GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-description }
```

11.24 Business category general information item

A business category specifies information concerning the occupation of the associated patient.

```
businessCategory GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-businessCategory }
```

11.25 Postal code general information item

The postal code specifies the address information required for the physical delivery of postal messages by the postal authority to the associated patient.

```
postalCode GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-postalCode }
```

11.26 Post office box general information item

The post office box specifies the post office box number of the associated patient.

```
postOfficeBox GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-di-postOfficeBox }
```

11.27 Telephone number general information item

The telephone number information item specifies a telephone number associated with the patient.

```
telephoneNumber GEN-INFO ::= {
  SYNTAX   TelephoneNumber
  ID       id-di-telephoneNumber }

TelephoneNumber ::= PrintableString(SIZE (1..ub-telephone-number))
-- String complying with Rec. ITU-T E.123 only

ub-telephone-number INTEGER ::= 32
```

11.28 Mobile number general information item

The mobile number information item specifies a telephone number associated with the patient.

```
mobileNumber GEN-INFO ::= {
  SYNTAX   TelephoneNumber
  ID       id-di-mobileNumber }
```

Many sensory activities are possible. This Recommendation probably cannot cover all sensory activities. New sensor devices may be invented requiring special considerations. Future additions may add new clauses to this section and/or other Recommendations and international standards may also provide supplementary specifications. Currently, only motion detection is included.

Temporary note: The motion detection is also used as a test case for the techniques developed for this Recommendation.

12 Motion detection

12.1 Motion sensing techniques

12.1.1 Passive infrared (PIR) sensing

The Stefan–Boltzmann law, also known as Stefan's law, states that the total energy radiated per unit surface area of a black body per unit time is directly proportional to the fourth power of the black body's thermodynamic temperature *T* (also called the absolute temperature).

A grey body emits less radiated energy than a black body. Instead, it radiates a fraction of that, characterized by its emissivity.

The irradiance is measured in joules per second per square metre, or equivalently, in watts per square metre.

The energy in watts or joules per second radiated from a black object per square metre is given by:

$$\phi = \sigma T^4$$

where

$$\sigma = 5.670 \times 10^{-8}$$

and where *T* is the absolute temperature (kelvin).

12.1.2 Microwave motion sensing

A microwave motion detector sends out electromagnetic pulses and measures the changes in frequency (Doppler) due to reflection off a moving object.

This type of motion sensing is most efficient if the distance between the sensor and the moving object changes.

12.2 Monitor type specifications

12.2.1 Overview

```
MotionPDU ::= CHOICE {
  powerOnReq    [0]  MotionPowerOnReq,
  powerOnAcc    [1]  MotionPowerOnAcc,
  powerOnErr    [2]  MotionPowerOnErr,
  reportReq     [3]  MotionReportReq,
  motionStopped [4]  MotionStopped,
  reportAck     [5]  MotionReportAck,
  reportErr     [6]  MotionReportErr,
  powerOffReq   [7]  MotionPowerOffReq,
  powerOffAcc   [8]  MotionPowerOffAcc,
  powerOffErr   [9]  MotionPowerOffErr,
  walkTestOnReq [10] MotionWalkTestOnReq,
  walkTestOnAcc [11] MotionWalkTestOnAcc,
  walkTestOnErr [12] MotionWalkTestOnErr,
```

```
  testReportReq [13] MotionWalkTestReportReq,
  reportStopped [14] MotionWalkTestReportStopped,
  testReportAck [15] MotionWalkTestReportAck,
  testReportErr [16] MotionWalkTestReportErr,
  testOffReq    [17] MotionWalkTestOffReq,
  testOffAcc    [18] MotionWalkTestOffAcc,
  testOffErr    [19] MotionWalkTestOffErr,
  ... }
```

12.2.2 Power-on request

The `motionPowerOnReq` PrPDU is used by the expert facility to request the powering-up of a motion detection device

```
MotionPowerOnReq ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

InvokeID ::= INTEGER (0..MAX)
```

12.2.3 Power-on accept

```
MotionPowerOnAcc ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

12.2.4 Power-on error

```
MotionPowerOnErr ::= SEQUENCE {
  invokeID    InvokeID,
  err         MotionPowerOnError,
  ... }

MotionPowerOnError ::= ENUMERATED {
  outOfSequence                 (0),
  operationFailed               (1),
  sensorTemporarilyUnavailable  (2),
  sensorPermanentlyUnavailable    (3),
  walkTest                      (4),
  tampered                      (5),
  ... }
```

The `MotionPowerOnError` data type shall take one of following values:

a) `outOfSequence`: The operation was invoked out of the allowed sequence.

b) `operationFailed`: The patient-local facility was not able to power-on the motion detector. Either the remote medical expert facility may try again after a while or it may terminate the sensing session. It is a local decision when to retry and how many times to retry.

c) `sensorTemporarilyUnavailable`: The sensor needed for the type of sensing is temporarily out of service. It is possible to retry the operation after a period. It is a local decision when to retry and how many times to retry.

d) `sensorPermanentlyUnavailable`: The motion detector has a permanent error that needs attention.

e) `walkTest`: The motion detector was locally strapped for walk test and will therefore not send normal motion report. When the walk test has completed, the patient-local facility shall invoke am instance of the `motionReport` indicating that the walk test has stopped and normal reporting will commence.

f) `tampered`: The motion detector has been tampered with and is therefore out of operation.

12.2.5 Motion detector report request

A **motionReport** operation shall be invoked by the patient facility when a motion is detected or when the motion has stopped again. The patient facility may issue multiple requests without waiting for response.

```
MotionReportReq ::= SEQUENCE {
  invokeID    InvokeID,
  unit        Unit {{watt}},
  direction   Direction (out),
  ... }
```

a) The **motion** alternative shall be taken if a motion has been detected. It includes two components:

   – The **ok** component signals that a motion has been detected, which is a **BOOLEAN** having the value **logic1** (**TRUE**) indicating that a motion has been detected.

   – The **value** component, when present, shall report the strength of the emission in joules per second and per square metre (watts per square metre). It shall be present if the motion detector is a PIR sensor. Otherwise, it shall be absent.

12.2.6 Motion stopped result

```
MotionStopped ::= SEQUENCE {
  invokeID    InvokeID,
  stopped     StopReason,
  ... }

StopReason ::= SEQUENCE {
  exceptions   ENUMERATED {
    walkTestON                   (0),
    walkTestOff                  (1),
    tampered                     (2),
    poweredOff                   (3),
    sensorTemporarilyUnavailable (4),
    sensorPermanentlyUnavailable (5),
    ... },
  ... }
```

– **walkTestOn**: The patient-local facility reports that the motion detector has been locally strapped for walk test and that for a period it will not send motion reports.

– **walkTestOff**: The strapping for local walk test has been removed and the motion detector returns to normal reporting.

– **tampered**: This enumeration shall be taken if the motion senor has been tampered with. When this alternative is taken, the motion sensor is assumed to be permanently out of operation. The sensing session shall be terminated.

– **poweredOff**: This enumeration shall be taken if the motion sensor has been powered off. When this alternative is taken, the motion sensor is assumed to be out of operation. The sensing session shall be terminated.

– **sensorTemporarilyUnavailable**: The sensor is temporarily unavailable due to a short-lived condition. A new power-on after a time-out period is necessary for continuous operation. Alternatively, the sensing session may be terminated.

– **sensorPermanentlyUnavailable**: The sensor is temporarily unavailable due to a long-lived condition that may require manual intervention. The sensing session shall be terminated.

12.2.7 Motion detector report acknowledgement

```
MotionReportAck ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

12.2.8 Motion detector report error

```
MotionReportErr ::= SEQUENCE {
  invokeID    InvokeID,
  error       MotionReportError,
  ... }

MotionReportError ::= ENUMERATED {
  outOfSequence (0),
  ... }
```

The `MotionReportError` data type shall take the following value:

a) `outOfSequence`: A motion report was unexpectedly received.

12.2.9 Motion detector power-off request

```
MotionPowerOffReq ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

The `motionPowerOff` shall be invoked when the remote medical expert facility wants to terminate the motion sensing reporting from the patient-local facility.

12.2.10 Motion detector power-off accept

```
MotionPowerOffAcc ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

12.2.11 Motion detector power-off error

```
MotionPowerOffErr ::= SEQUENCE {
  invokeID    InvokeID,
  err         MotionPowerOffError,
  ... }

MotionPowerOffError ::= ENUMERATED {
  outOfSequence  (0),
  ... }
```

The `MotionPowerOffError` data type shall take the following value:

– `outOfSequence`: A motion power-off was unexpectedly received.

12.2.12 Motion walk test on request

A walk test on request is carried as an instance of the `motionWalkTestOnReq` object.

```
MotionWalkTestOnReq ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

12.2.13 Motion walk test on accept

```
MotionWalkTestOnAcc ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

12.2.14 Motion walk test on error

```
MotionWalkTestOnErr ::= SEQUENCE {
  invokeID    InvokeID,
  error       MotionWalkTestOnError,
  ... }

MotionWalkTestOnError ::= ENUMERATED {
  notSupported                 (0),
  voiceNotProvided             (1),
  operationFailed              (2),
  sensorTemporarilyUnavailable (3),
  sensorPermanentlyUnavailable (4),
  tampered                     (5),
  outOfSequencece              (6),
  ... }
```

The **`MotionWalkTestOnError`** data type defines the following error codes:

a) **`notSupported`**: The patient-local facility does not support remotely initiated walk test. Normal motion detection is assumed to proceed.

b) **`voiceNotProvided`**: The patient-local facility has no means for voice communication, which means that the remote walk test cannot commence. Normal motion detection is assumed to proceed.

c) **`operationFailed`**: The patient-local facility was not able to switch the motion detector to remote walk test. Either the remote medical expert facility may try again after a while or it may terminate the sensing session. It is a local decision when to retry and how many times to retry.

d) **`sensorTemporarilyUnavailable`**: The sensor is temporarily unavailable due to a short-lived condition. A new power-on after a time-out period is necessary for continuous operation. Alternatively, the sensing session may be terminated.

e) **`sensorPermanentlyUnavailable`**: The sensor is temporarily unavailable due to a long-lived condition that may require manual intervention. The sensing session shall be terminated.

f) **`tampered`**: The motion detector has been tampered with and is out of operation.

g) **`outOfSequence`**: Motion detector was not powered-on.

12.2.15 Motion walk report request

```
MotionWalkTestReportReq ::= SEQUENCE {
  invokeID    InvokeID,
  unit        Unit {{watt}},
  direction   Direction (out),
  ... }
```

12.2.16 Motion walk test stopped

```
MotionWalkTestReportStopped ::= SEQUENCE {
  invokeID    InvokeID,
  stopped     WalkTestReportStopped,
  ... }

WalkTestReportStopped ::= ENUMERATED {
  tampered,
  poweredOff,
  sensorTemporarilyUnavailable,
  sensorPermanentlyUnavailable,
  ... }
```

12.2.17 Motion walk report acknowledgement

```
MotionWalkTestReportAck ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

12.2.18 Motion walk report error

```
MotionWalkTestReportErr ::= SEQUENCE {
  error       WalkTestError,
  invokeID    InvokeID,
  ... }

WalkTestError ::= ENUMERATED {
  outOfSequence (0),
  ... }
```

12.2.19 Walk test off request

```
MotionWalkTestOffReq ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

12.2.20 Walk test off acknowledgement

```
MotionWalkTestOffAcc ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

12.2.21 Walk test off error

```
MotionWalkTestOffErr ::= SEQUENCE {
  error       WalkTestOffError,
  invokeID    InvokeID,
  ... }

WalkTestOffError ::= ENUMERATED {
  outOfSequence (0),
  ... }
```

Annex A

General introduction to sensing

(This annex forms an integral part of this Recommendation.)

A.1 Scope of annex

This clause gives some examples of sensing types and body positions.

A.2 Types of physical interaction

Physical interactions with a patient may be classified as:

a) Sensing: Detection by a sensor (and transmission) of a spontaneous emission coming from the human body (for example, ECG or a visual inspection).

b) Probing: Detection by a sensor (and transmission) of an emission produced by an external source after having been modified because of interaction with the body (for example, standard X-ray imaging).

c) Probing-sensing: Detection by a sensor of an emission produced by the body after having been stimulated by an external emission (for example, magnetic resonance imaging).

d) Strictly non-invasive: The sensor (for example a video camera) does not touch the patient. However, for this application magnification, zoom and nudge facilities may be required, controlled by the remote medical expert facility.

a) Non-invasive: The patient is touched by the sensor (for example, an ECG electrode), with intact skin, no contact with internal mycoses and no passage between natural cavities or orifices of the body.

Others types of physical interactions will be covered by later parts of the ITU-T X.1080 Series of Recommendations.

A.3 Body position

Some types of sensing (such as an ECG) do not require the body position to be identified.

In other cases, the place on the body to be inspected (bowels, kidney, back, head, etc.) needs to be identified.

In all cases where a body position needs to be identified, this should be done by agreement prior to the sensing session. How this agreement is reached is outside the scope of this Recommendation.

A.4 Sensing session

A sensing instance, called a sensing session, is initiated by a set-up exchange specifying the overall conditions under which sensing shall be conducted. The set-up sequence is followed by one or more operations particular to the type of sensing. The sensing session is concluded by a termination exchange. The session set-up and termination are specified in [ITU-T X.1080.1].

Several sessions may be in progress simultaneously. Each sensing session instance is identified by an object identifier that identifies the type of session and an integer that identifies a particular session among sessions of the same type.

Each operation invoked within a sensing session is given a unique invoke identifier that together with the sensing session identifier uniquely identifies a particular operation in progress.

Annex B

Biology-to-machine (B2M) protocol in ASN.1

(This annex forms an integral part of this Recommendation.)

This annex provides the formal definition of the B2M protocol defined by this Recommendation in the form of the ASN.1 module `Biology-to-Machine`.

```
Biology-to-Machine { joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2)
   modules(0) b2m(0) version1(1)}

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS

  SupportedGenInfo, SupportedUnits
    FROM
      InfoObjects {joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2)
                   modules(0) infoObjects(2) version1(1)}

  moving-detect
    FROM MonitorTypes
      {joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2)
      modules(0) monitor-types(1) monitors(0) version1(1)} ;

id-telebio          OBJECT IDENTIFIER ::= { joint-iso-itu-t(2) telebiometrics(42)
}
id-thprot           OBJECT IDENTIFIER ::= { id-telebio thprot(10) }
id-x1080-2          OBJECT IDENTIFIER ::= { id-thprot part2(2) }
id-x1080-2-modules OBJECT IDENTIFIER ::= { id-x1080-2 modules(0) }
id-mt               OBJECT IDENTIFIER ::= { id-x1080-2 monitor-type(1) }
id-un               OBJECT IDENTIFIER ::= { id-x1080-2 unit(2) }
id-gi               OBJECT IDENTIFIER ::= { id-x1080-2 gen-info(3) }

MONITOR-TYPE ::= TYPE-IDENTIFIER

Direction ::= ENUMERATED {in(0), out(1)}

DeviceInfo ::= SEQUENCE {
  vendor [0] UTF8String OPTIONAL,
  type   [1] UTF8String OPTIONAL,
  id     [2] UTF8String OPTIONAL,
  ... }

SupportedMonitorTypes MONITOR-TYPE ::= {moving-detect}

OPTIONS ::= CLASS {
  &Voice-options   OPTIONAL,
  &Video-options   OPTIONAL }
WITH SYNTAX {
  [VOICE OPTIONS   &Voice-options]
  [VIDEO OPTIONS   &Video-options] }

UNIT ::= CLASS {
  &name      PrintableString,
  &symbol    PrintableString,
  &quantity  PrintableString OPTIONAL,
  &Value,
  &id        OBJECT IDENTIFIER UNIQUE }
WITH SYNTAX {
```

```
  NAME       &name
  SYMBOL     &symbol
  [QUANTITY  &quantity]
  VALUE      &Value
  ID         &id }

Unit{UNIT:SupportedUnits} ::= SEQUENCE {
  id             UNIT.&id({SupportedUnits}),
  name           UNIT.&name({SupportedUnits}{@id}),
  symbol         UNIT.&symbol({SupportedUnits}{@id}),
  quantity   [0] UNIT.&quantity({SupportedUnits}{@id}) OPTIONAL,
  value      [1] UNIT.&Value }

GEN-INFO ::= CLASS {
  &Type,
  &id      OBJECT IDENTIFIER UNIQUE }
WITH SYNTAX {
  SYNTAX  &Type
  ID      &id }

Gen-Info {GEN-INFO:SupportedGenInfo} ::= SEQUENCE {
  type     GEN-INFO.&id({SupportedGenInfo}),
  value    GEN-INFO.&Type({SupportedGenInfo}{@type}) }

B2M-protocol ::= CHOICE {
  b2mInitReq    [0] B2MInitReq,
  b2mInitAcc    [1] B2MInitAcc,
  b2mInitRej    [2] B2MInitRej,
  b2mInitAbt    [3] B2MInitAbt,
  b2mTranf      [4] B2MDataTransfer,
  ... }

B2MInitReq ::= SEQUENCE {
  version           Version,
  requirements      ENUMERATED {
    none                                (0),
    with-2-way-voice                    (1),
    with-reverse-video-and-2-way-voice  (2),
    with-2-way-video-and-voice          (3),
    ... } DEFAULT none,
  voice-options [0] OPTIONS.&Voice-options OPTIONAL,
  video-options [1] OPTIONS.&Video-options OPTIONAL,
  monitorTypes      SEQUENCE SIZE (1..MAX) OF MONITOR-TYPE.&id
({SupportedMonitorTypes}),
  ... }

Version ::= BIT STRING {
  v1 (0)  -- version 1
  }

B2MInitAcc ::= SEQUENCE {
  version       Version,
  monitorTypes  SEQUENCE SIZE (1..MAX) OF MONITOR-TYPE.&id OPTIONAL,
  ... }

B2MInitRej ::= SEQUENCE {
  version Version,
  initErr InitError,
  ... }

InitError ::=  ENUMERATED {
  initiation-not-allowed      (0),
  versions-not-supported      (1),
  voice-option-not-available  (2),
```

```
  video-option-not-available  (3),
  monitor-types-not-supported (4),
  ... }

B2MInitAbt ::= SEQUENCE {
  version Version,
  accErr  AccError,
  ... }

AccError ::= ENUMERATED {
  only-one-version              (0),
  unexpected-version            (1),
  unexpected-monitorTypes       (2),
  critical-monitorTypes-missing (3),
  ... }

B2MDataTransfer ::= SEQUENCE {
  time               GeneralizedTime,
  genInfos       [0] SEQUENCE SIZE (1..MAX) OF CHOICE {
    genInfoType      GEN-INFO.&id({SupportedGenInfo}),
    genInfo          Gen-Info{{SupportedGenInfo}},
    ... } OPTIONAL,
  monTypes     [1] SEQUENCE SIZE (1..MAX) OF SEQUENCE SIZE (1..MAX) OF SEQUENCE {
    deviceInfo   [1] DeviceInfo OPTIONAL,
    mon          [2] MONITOR-TYPE.&Type ({SupportedMonitorTypes}) OPTIONAL,
    ... },
  ... }

InvokeID ::= INTEGER (0..MAX)

END -- Biology-to-Machine
```

Annex C

Defined information objects in ASN.1

(This annex forms an integral part of this Recommendation.)

```
InfoObjects { joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2)
   modules(0) infoObjects(2) version1(1)}

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS

  GEN-INFO, id-gi, id-un, UNIT
    FROM Biology-to-Machine
      { joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2) modules(0)
b2m(0) version1(1)} ;


-- Unit information objects

SupportedUnits UNIT ::= {metre | kilogram | second | ampere | kelvin | mole |
                         candela | hertz | newton | pascal | joule | watt |
                         coulomb | volt}

metre UNIT ::= {
  NAME     "metre"
  SYMBOL   "m"
  QUANTITY "length"
  VALUE    REAL
  ID       id-un-metre }

kilogram UNIT ::= {
  NAME     "kilogram"
  SYMBOL   "kg"
  QUANTITY "mass"
  VALUE    REAL
  ID       id-un-kilogram }

second UNIT ::= {
  NAME     "second"
  SYMBOL   "s"
  QUANTITY "time"
  VALUE    REAL
  ID       id-un-second }

ampere UNIT ::= {
  NAME     "ampere"
  SYMBOL   "A"
  QUANTITY "electric-current"
  VALUE    REAL
  ID       id-un-ampere }

kelvin UNIT ::= {
  NAME     "kelvin"
  SYMBOL   "K"
  QUANTITY "thermodynamic-temperature"
  VALUE    REAL
  ID       id-un-kelvin }

mole UNIT ::= {
```

```
  NAME     "mole"
  SYMBOL   "mol"
  QUANTITY "amount-of-substance"
  VALUE    REAL
  ID       id-un-mole }

candela UNIT ::= {
  NAME     "candela"
  SYMBOL   "cd"
  QUANTITY "luminous-intensity"
  VALUE    REAL
  ID       id-un-candela }

-- Derived units

hertz UNIT ::= {
  NAME    "hertz"
  SYMBOL  "Hz"
  VALUE   REAL
  ID      id-un-hertz }

newton UNIT ::= {
  NAME    "newton"
  SYMBOL  "N"
  VALUE   REAL
  ID      id-un-newton }

pascal UNIT ::= {
  NAME    "pascal"
  SYMBOL  "Pa"
  VALUE   REAL
  ID      id-un-pascal }

joule UNIT ::= {
  NAME    "joule"
  SYMBOL  "J"
  VALUE   REAL
  ID      id-un-joule }

watt UNIT ::= {
  NAME     "watt"
  SYMBOL   "W"
  QUANTITY "power"
  VALUE    REAL
  ID       id-un-watt }

coulomb UNIT ::= {
  NAME    "coulomb"
  SYMBOL  "C"
  VALUE   REAL
  ID      id-un-coulomb }

volt UNIT ::= {
  NAME    "volt"
  SYMBOL  "V"
  VALUE   REAL
  ID      id-un-volt }

farad UNIT ::= {
  NAME    "farad"
  SYMBOL  "F"
  VALUE   REAL
  ID      id-un-farad }
```

```
ohm UNIT ::= {
  NAME    "ohm"
  SYMBOL  "Ω"
  VALUE   REAL
  ID      id-un-ohm }

siemens UNIT ::= {
  NAME    "siemens"
  SYMBOL  "S"
  VALUE   REAL
  ID      id-un-siemens }

weber UNIT ::= {
  NAME    "weber"
  SYMBOL  "Wb"
  VALUE   REAL
  ID      id-un-weber }

tesla UNIT ::= {
  NAME    "tesla"
  SYMBOL  "T"
  VALUE   REAL
  ID      id-un-tesla }

henry UNIT ::= {
  NAME    "henry"
  SYMBOL  "H"
  VALUE   REAL
  ID      id-un-henry }

degreeCelsius UNIT ::= {
  NAME    "degreeCelsius"
  SYMBOL  "°C"
  VALUE   REAL
  ID      id-un-degreeCelsius }

lumen UNIT ::= {
  NAME    "lumen"
  SYMBOL  "lm"
  VALUE   REAL
  ID      id-un-lumen }

lux UNIT ::= {
  NAME    "lux"
  SYMBOL  "lx"
  VALUE   REAL
  ID      id-un-lux }

becquerel UNIT ::= {
  NAME    "becquerel"
  SYMBOL  "Bq"
  VALUE   REAL
  ID      id-un-becquerel }

id-un-metre                    OBJECT IDENTIFIER ::= {id-un 1}
id-un-kilogram                 OBJECT IDENTIFIER ::= {id-un 2}
id-un-second                   OBJECT IDENTIFIER ::= {id-un 3}
id-un-ampere                   OBJECT IDENTIFIER ::= {id-un 4}
id-un-kelvin                   OBJECT IDENTIFIER ::= {id-un 5}
id-un-mole                     OBJECT IDENTIFIER ::= {id-un 6}
id-un-candela                  OBJECT IDENTIFIER ::= {id-un 7}
id-un-hertz                    OBJECT IDENTIFIER ::= {id-un 8}
id-un-newton                   OBJECT IDENTIFIER ::= {id-un 9}
id-un-pascal                   OBJECT IDENTIFIER ::= {id-un 10}
```

```
id-un-joule                    OBJECT IDENTIFIER ::= {id-un 11}
id-un-watt                     OBJECT IDENTIFIER ::= {id-un 12}
id-un-coulomb                  OBJECT IDENTIFIER ::= {id-un 13}
id-un-volt                     OBJECT IDENTIFIER ::= {id-un 14}
id-un-farad                    OBJECT IDENTIFIER ::= {id-un 15}
id-un-ohm                      OBJECT IDENTIFIER ::= {id-un 16}
id-un-siemens                  OBJECT IDENTIFIER ::= {id-un 17}
id-un-weber                    OBJECT IDENTIFIER ::= {id-un 18}
id-un-tesla                    OBJECT IDENTIFIER ::= {id-un 19}
id-un-henry                    OBJECT IDENTIFIER ::= {id-un 20}
id-un-degreeCelsius            OBJECT IDENTIFIER ::= {id-un 21}
id-un-lumen                    OBJECT IDENTIFIER ::= {id-un 22}
id-un-lux                      OBJECT IDENTIFIER ::= {id-un 23}
id-un-becquerel                OBJECT IDENTIFIER ::= {id-un 24}

-- Data item information objects

SupportedGenInfo GEN-INFO ::= { surname | givenName | initials |
generationQualifier |
                                serialNumber | pseudonym | uri | urn | url, ...
}

surname GEN-INFO ::= {
  SYNTAX   Surname
  ID       id-gi-surname }

Surname ::= UTF8String

givenName GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-givenName }

initials GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-initials }

generationQualifier GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-generationQualifier }

serialNumber GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-serialNumber }

pseudonym GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-pseudonym }

uri GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-uri }

urn GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-urn }

url GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-url }

dnsName GEN-INFO ::= {
  SYNTAX   DomainName
  ID       id-gi-dnsName }
```

```
DomainName ::= UTF8String (CONSTRAINED BY
  { -- Conforms to the format of a (internationalized) domain name. -- })

email GEN-INFO ::= {
  SYNTAX   IA5String
  ID       id-gi-email }

countryName GEN-INFO ::= {
  SYNTAX   CountryName
  ID       id-gi-countryName }

CountryName ::= PrintableString(SIZE (2))
  (CONSTRAINED BY { -- ISO 3166 alpha-2 codes only -- })

countryCode3c GEN-INFO ::= {
  SYNTAX   CountryCode3c
  ID       id-gi-countryCode3c }

CountryCode3c ::= PrintableString(SIZE (3))
  (CONSTRAINED BY { -- ISO 3166 alpha-3 codes only -- })

localityName GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-localityName }

streetAddress GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-streetAddress }

houseIdentifier GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-houseIdentifier }

utmCoordinates GEN-INFO ::= {
  SYNTAX   UtmCoordinates
  ID       id-gi-utmCoordinates }

UtmCoordinates ::= SEQUENCE {
  zone      PrintableString,
  easting   NumericString,
  northing  NumericString }

organizationName GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-organizationName }

organizationalUnitName GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-organizationalUnitName }

title GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-title }

organizationIdentifier GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-organizationIdentifier }

description GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-description }

businessCategory GEN-INFO ::= {
  SYNTAX   UTF8String
```

```
  ID       id-gi-businessCategory }

postalCode GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-postalCode }

postOfficeBox GEN-INFO ::= {
  SYNTAX   UTF8String
  ID       id-gi-postOfficeBox }

telephoneNumber GEN-INFO ::= {
  SYNTAX   TelephoneNumber
  ID       id-gi-telephoneNumber }

TelephoneNumber ::= PrintableString(SIZE (1..ub-telephone-number))
-- String complying with Rec. ITU-T E.123 only

ub-telephone-number INTEGER ::= 32

mobileNumber GEN-INFO ::= {
  SYNTAX   TelephoneNumber
  ID       id-gi-mobileNumber }

-- Allocation of object identifiers

id-gi-surname                 OBJECT IDENTIFIER ::= {id-gi 1}
id-gi-givenName               OBJECT IDENTIFIER ::= {id-gi 2}
id-gi-initials                OBJECT IDENTIFIER ::= {id-gi 3}
id-gi-generationQualifier     OBJECT IDENTIFIER ::= {id-gi 4}
id-gi-serialNumber            OBJECT IDENTIFIER ::= {id-gi 5}
id-gi-pseudonym               OBJECT IDENTIFIER ::= {id-gi 6}
id-gi-uri                     OBJECT IDENTIFIER ::= {id-gi 7}
id-gi-urn                     OBJECT IDENTIFIER ::= {id-gi 8}
id-gi-url                     OBJECT IDENTIFIER ::= {id-gi 9}
id-gi-dnsName                 OBJECT IDENTIFIER ::= {id-gi 10}
id-gi-email                   OBJECT IDENTIFIER ::= {id-gi 11}
id-gi-countryName             OBJECT IDENTIFIER ::= {id-gi 12}
id-gi-countryCode3c           OBJECT IDENTIFIER ::= {id-gi 13}
id-gi-localityName            OBJECT IDENTIFIER ::= {id-gi 14}
id-gi-streetAddress           OBJECT IDENTIFIER ::= {id-gi 15}
id-gi-houseIdentifier         OBJECT IDENTIFIER ::= {id-gi 16}
id-gi-utmCoordinates          OBJECT IDENTIFIER ::= {id-gi 17}
id-gi-organizationName        OBJECT IDENTIFIER ::= {id-gi 18}
id-gi-organizationalUnitName OBJECT IDENTIFIER ::= {id-gi 19}
id-gi-title                   OBJECT IDENTIFIER ::= {id-gi 20}
id-gi-organizationIdentifier OBJECT IDENTIFIER ::= {id-gi 21}
id-gi-description             OBJECT IDENTIFIER ::= {id-gi 22}
id-gi-businessCategory        OBJECT IDENTIFIER ::= {id-gi 23}
id-gi-postalCode              OBJECT IDENTIFIER ::= {id-gi 24}
id-gi-postOfficeBox           OBJECT IDENTIFIER ::= {id-gi 25}
id-gi-telephoneNumber         OBJECT IDENTIFIER ::= {id-gi 26}
id-gi-mobileNumber            OBJECT IDENTIFIER ::= {id-gi 27}

END -- InfoObjects
```

Annex D

Monitor types in ASN.1

(This annex forms an integral part of this Recommendation.)

```
MonitorTypes { joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2)
   modules(0) monitor-types(1) monitors(0) version1(1)}

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS

  id-mt, MONITOR-TYPE
    FROM Biology-to-Machine
    {joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2) modules(0) b2m(0)
version1(1)}

  MotionPDU
    FROM Moving-detection
      {joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2)
      modules(0) monitor-types(1) moving-detect(1) version1(1)}

 ;

moving-detect MONITOR-TYPE ::= {
  MotionPDU
  IDENTIFIED BY id-moving-detect }


id-moving-detect  OBJECT IDENTIFIER ::= { id-mt moving-detect(1) }

END
```

Annex E

Motion detection monitor type in ASN.1

(This annex forms an integral part of this Recommendation.)

```
Moving-detection { joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2)
   modules(0) monitor-types(1) moving-detect(1) version1(1)}

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS

  DeviceInfo, Direction, MONITOR-TYPE, --Quantity{},-- Unit{}
    FROM Biology-to-Machine
      {joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2) modules(0)
b2m(0) version1(1)}

  id-moving-detect
    FROM MonitorTypes { joint-iso-itu-t(2) telebiometrics(42) thprot(10)
part2(2)
   modules(0) monitor-types(1) monitors(0) version1(1)}

  watt
    FROM InfoObjects
      { joint-iso-itu-t(2) telebiometrics(42) thprot(10) part2(2) modules(0)
        infoObjects(2) version1(1)} ;

MotionPDU ::= CHOICE {
  powerOnReq    [0]  MotionPowerOnReq,
  powerOnAcc    [1]  MotionPowerOnAcc,
  powerOnErr    [2]  MotionPowerOnErr,
  reportReq     [3]  MotionReportReq,
  motionStopped [4]  MotionStopped,
  reportAck     [5]  MotionReportAck,
  reportErr     [6]  MotionReportErr,
  powerOffReq   [7]  MotionPowerOffReq,
  powerOffAcc   [8]  MotionPowerOffAcc,
  powerOffErr   [9]  MotionPowerOffErr,
  walkTestOnReq [10] MotionWalkTestOnReq,
  walkTestOnAcc [11] MotionWalkTestOnAcc,
  walkTestOnErr [12] MotionWalkTestOnErr,
  testReportReq [13] MotionWalkTestReportReq,
  reportStopped [14] MotionWalkTestReportStopped,
  testReportAck [15] MotionWalkTestReportAck,
  testReportErr [16] MotionWalkTestReportErr,
  testOffReq    [17] MotionWalkTestOffReq,
  testOffAcc    [18] MotionWalkTestOffAcc,
  testOffErr    [19] MotionWalkTestOffErr,
  ... }

MotionPowerOnReq ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

InvokeID ::= INTEGER (0..MAX)

MotionPowerOnAcc ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

```
MotionPowerOnErr ::= SEQUENCE {
  invokeID    InvokeID,
  err         MotionPowerOnError,
  ... }

MotionPowerOnError ::= ENUMERATED {
  outOfSequence                 (0),
  operationFailed               (1),
  sensorTemporarilyUnavailable (2),
  sensorPermanentlyUnavailable   (3),
  walkTest                      (4),
  tampered                      (5),
  ... }

MotionReportReq ::= SEQUENCE {
  invokeID    InvokeID,
--  quantity    Quantity {{power}},
  unit        Unit {{watt}},
  direction   Direction (out),
  ... }

MotionStopped ::= SEQUENCE {
  invokeID    InvokeID,
  stopped     StopReason,
  ... }

StopReason ::= SEQUENCE {
  exceptions   ENUMERATED {
    walkTestON                    (0),
    walkTestOff                   (1),
    tampered                      (2),
    poweredOff                    (3),
    sensorTemporarilyUnavailable (4),
    sensorPermanentlyUnavailable (5),
    ... },
  ... }

MotionReportAck ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

MotionReportErr ::= SEQUENCE {
  invokeID    InvokeID,
  error       MotionReportError,
  ... }

MotionReportError ::= ENUMERATED {
  outOfSequence (0),
  ... }

MotionPowerOffReq ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

MotionPowerOffAcc ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

MotionPowerOffErr ::= SEQUENCE {
  invokeID    InvokeID,
  err         MotionPowerOffError,
  ... }

MotionPowerOffError ::= ENUMERATED {
```

```
  outOfSequence  (0),
  ... }

MotionWalkTestOnReq ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

MotionWalkTestOnAcc ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

MotionWalkTestOnErr ::= SEQUENCE {
  invokeID    InvokeID,
  error       MotionWalkTestOnError,
  ... }

MotionWalkTestOnError ::= ENUMERATED {
  notSupported                  (0),
  voiceNotProvided              (1),
  operationFailed               (2),
  sensorTemporarilyUnavailable  (3),
  sensorPermanentlyUnavailable  (4),
  tampered                      (5),
  outOfSequencece               (6),
  ... }

MotionWalkTestReportReq ::= SEQUENCE {
  invokeID    InvokeID,
  unit        Unit {{watt}},
  direction   Direction (out),
  ... }

MotionWalkTestReportStopped ::= SEQUENCE {
  invokeID    InvokeID,
  stopped     WalkTestReportStopped,
  ... }

WalkTestReportStopped ::= ENUMERATED {
  tampered,
  poweredOff,
  sensorTemporarilyUnavailable,
  sensorPermanentlyUnavailable,
  ... }

MotionWalkTestReportAck ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

MotionWalkTestReportErr ::= SEQUENCE {
  error       WalkTestError,
  invokeID    InvokeID,
  ... }

WalkTestError ::= ENUMERATED {
  outOfSequence (0),
  ... }

MotionWalkTestOffReq ::= SEQUENCE {
  invokeID    InvokeID,
  ... }

MotionWalkTestOffAcc ::= SEQUENCE {
  invokeID    InvokeID,
  ... }
```

```
MotionWalkTestOffErr ::= SEQUENCE {
  error       WalkTestOffError,
  invokeID    InvokeID,
  ... }

WalkTestOffError ::= ENUMERATED {
  outOfSequence (0),
  ... }

END -- Moving-detection
```

Bibliography

[b-ITU-T E.123] Recommendation ITU-T E.123 (2002), *Notation for national and international telephone numbers, e-mail addresses and web addresses*.

[b-ITU-T X.1081] Recommendation ITU-T X.1081 (2011), *The telebiometric multimodal model – A framework for the specification of security and safety aspects of telebiometrics*.

[b-ITU-T X.1082] Recommendation ITU-T X.1082 (2007), *Telebiometrics related to human physiology*.

[b-ITU-T X.1082 Amd.1] Recommendation ITU-T X.1082 Amendment 1 (2009), *Telebiometrics related to human physiology Amendment 1: Object identifier assignments under the telebiometrics arc*.

[b-BIPM] International Bureau of Weights and Measures (2019), *The International System of Units (SI)*.
https://www.bipm.org/utils/common/pdf/si-brochure/SI-Brochure-9-EN.pdf

[b-ISO 3166-1] ISO 3166-1:2013, *Codes for the representation of names of countries and their subdivisions – Part 1: Country codes*.

[b-ISO/IEC 10646] ISO/IEC 10646 (2017), *Information technology – universal coded character set (UCS)*.

[b-RFC 3406] IETF RFC 3406 (2002), *Uniform Resource Names (URN) Namespace Definition Mechanisms*.

[b-RFC 3492] IETF RFC 3492 (2003), *Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)*.

[b-RFC 3986] IETF RFC 3986 (2005), *Uniform Resource Identifier (URI): Generic Syntax*.

[b-RFC 5892] IETF RFC 5892 (2010), *The Unicode Code Points and Internationalized Domain Names for Applications (IDNA)*.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2021