DEUTSCHE NORM November 2006

DIN EN ISO 11073-10201

ICS 35.240.80

Medizinische Informatik – Kommunikation patientennaher medizinischer Geräte – Teil 10201: Bereichs-Informationsmodell (ISO/IEEE 11073-10201:2004); Englische Fassung EN ISO 11073-10201:2005

Health informatics –
Point-of-care medical device communication –
Part 10201: Domain information model (ISO/IEEE 11073-10201:2004);
English version EN ISO 11073-10201:2005

Informatique de santé –
Communication entre dispositifs médicaux sur le site des soins –
Partie 10201: Modèle d'information du domaine (ISO/IEEE 11073-10201:2004);
Version anglaise EN ISO 11073-10201:2005

Gesamtumfang 177 Seiten

Normenausschuss Medizin (NAMed) im DIN

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

Nationales Vorwort

Diese Norm enthält unter Berücksichtigung des Präsidialbeschlusses 13/1983 den englischen Originaltext der Europäischen Norm EN ISO 11073-10201:2005. Die Europäische Norm wurde in der WG IV „Technology for Interoperability“ des CEN/TC 251 „Medizinische Informatik“ übernommen und im Rahmen eines gemeinsamen Projektes zwischen ISO und IEEE erarbeitet. Der NA 063-07-02 AA „Interoperabilität“ des Fachbereichs 7 „Medizinische Informatik“ des Normenausschusses Medizin im DIN hat an der Erarbeitung mitgewirkt.

Für die im Abschnitt 2 zitierten Normen wird im Folgenden auf die entsprechenden Deutschen Normen hingewiesen:

ISO 639-1	siehe DIN 2335
ISO 3166-1	siehe DIN EN ISO 3166-1
ISO 3166-2	siehe DIN ISO 3166-2
ISO 3166-3	siehe DIN ISO 3166-3
ISO 8601	siehe E DIN ISO 8601
ISO 15225	siehe DIN EN ISO 15225
ISO/IEC 2022	siehe DIN ISO/IEC 2022
ISO/IEC 8859-1	siehe DIN 66303
ISO/IEC 9545	siehe DIN ISO/IEC 9545
ISO/IEEE 11073-10101	siehe DIN EN ISO 11073-10101

Zusammenfassung

Die ISO/IEEE11073-Standardfamilie ermöglicht die Datenübertragung zwischen Medizingeräten und Computersystemen. Sie erlaubt die automatisierte, detaillierte Erfassung von patientenbezogenen Vitaldaten und Funktionsparametern durch die Verwendung von Überwachungs- und Therapiegeräten — beispielsweise auf einer Intensivstation oder im Operationssaal. Die primären Ziele dabei sind:

- Bereitstellung von „Plug-and-Play“-Funktionalität und Interoperabilität mit Patienten verbundener Medizingeräte für Realzeitanwendungen.
- Vereinfachung des effizienten Austausches gemessener Vitaldaten von Patienten und Betriebsdaten der verwendeten Medizingeräte.

„Echtzeit“ bedeutet, dass Daten von mehreren Medizingeräten zeitgenau erfasst und in Bruchteilen von Sekunden dargestellt oder verarbeitet werden können, werden können. „Plug and Play“ beinhaltet, dass eine medizinische Fachkraft lediglich die Verbindung zum Gerät herstellen muss, worauf das System automatisch — ohne weiteren menschlichen Eingriff — das neu angeschlossene Medizingerät erkennt, Kommunikationsparameter sowie gegebenenfalls die Applikation konfiguriert und danach die Kommunikation von Daten aufnimmt.

Diese Norm dient innerhalb der ISO/IEEE-11073-Standardfamilie der Definition und Strukturierung der Information, die zwischen Anwendungsinstanzen kommuniziert oder in der Kommunikation referenziert wird. Er stellt eine gemeinsame syntaxunabhängige Beschreibung aller in den Anwendungsprozessen verschiedener Geräte vorhandenen Anwendungsinstanzen zur Verfügung. Einem objektorientierten Ansatz folgend werden Daten, z. B. Messwerte oder Zustandsinformation, Informationsobjekten zugeordnet, auf die über entsprechende Dienste zugegriffen werden kann. Die meisten der definierten Objekte repräsentieren verallgemeinerte Vitaldaten und unterstützende Information. Spezialisierungen dieser Objekte können durch die Definition zusätzlicher Attribute erfolgen. Gerätekonfigurationen und -funktionalitäten werden durch Objekthierarchien und -relationen beschrieben. Das resultierende Domänen-Informationsmodell (DIM) definiert umfassend die für die Kommunikation medizinischer Geräte benötigten Informationsobjekte mit ihren Attributen, Methoden und Zugriffsfunktionen.

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

Für die Objektorientierte Modellierung wird die „Unified Modelling Language“ (UML) verwendet. Um Handhabbarkeit und Übersichtlichkeit zu erhöhen, wird die gesamte Domäne in verschiedene Pakete unterteilt. Jedes Paket wird in Form eines separaten Objektdiagramms definiert. Für die Definition von Objekten wird die Textform benutzt; sämtliche Objekte werden kurz beschrieben. Attribute werden in Definitionstabellen festgelegt, gleiches gilt für Objektverhalten und Benachrichtigungen, die von Objekten erzeugt werden. Attribut-Datentypen werden unter Verwendung der Abstract Syntax Notation One (ASN.1) definiert.

Gegenstand dieser Norm sind auch Informationsmanagement-Objekte, die Konzepte und Fähigkeiten zur kosteneffektiven Kommunikation realisieren (insbesondere Objekte zur Datenkumulation), sowie Objekte zur Unterstützung der Realzeitkommunikation.

Interoperabilitätsaspekte, insbesondere die unteren ISO/OSI-Protokollschichten, Zeitsynchronisation zwischen verschiedenen Geräten etc. sind nicht Gegenstand dieser Norm.

Nationaler Anhang NA
(informativ)

Literaturhinweise

DIN 2335:1986-10, *Sprachenzeichen*

DIN 66303:2000-06, *Informationstechnik — 8-Bit-Code*

DIN EN ISO 3166-1:1998-04, *Codes für die Namen von Ländern und deren Untereinheiten — Teil 1: Codes für Ländernamen (ISO 3166-1:1997; Deutsche Fassung EN ISO 3166-1:1997)*

DIN EN ISO 11073-10101:2006-05, *Medizinische Informatik — Kommunikation patientennaher medizinischer Geräte — Teil 10101: Nomenklatur (ISO/IEEE 11073-10101:2004); Deutsche Fassung EN ISO 11073-10101:2005*

DIN EN ISO 15225:2005-09, *Nomenklatur — Spezifikation für ein Nomenklatursystem für Medizinprodukte zum Zweck des regulativen Datenaustauschs (ISO 15225:2000 + Amd 1:2004); Deutsche Fassung EN ISO 15225:2000 + A1:2004 + A2:2005*

DIN ISO/IEC 2022:1996-11, *Informationstechnik — Zeichencodestruktur und Erweiterungstechniken*

DIN ISO 3166-2:2001-04, *Codes für die Namen von Ländern und deren Untereinheiten — Teil 2: Code für Namen von Länderuntereinheiten (ISO 3166-2:1998)*

DIN ISO 3166-3:2001-02, *Information und Dokumentation — Codes für die Namen von Ländern und deren Untereinheiten — Teil 3: Code für früher gebräuchliche Ländernamen (ISO 3166-3:1999)*

E DIN ISO 8601:2005-11, *Datenelemente und Austauschformate — Informationsaustausch — Darstellung von Datum und Uhrzeit (ISO 8601:2004)*

DIN ISO/IEC 9545:1996-11, *Informationstechnik — Kommunikation Offener Systeme — Struktur der Anwendungsschicht (ISO/IEC 9545:1994)*

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

— Leerseite —

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN ISO 11073-10201

August 2005

ICS 35.240.80

English Version

Health informatics - Point-of-care medical device communication - Part 10201: Domain information model (ISO/IEEE 11073-10201:2004)

Informatique de santé - Communication entre dispositifs médicaux sur le site des soins - Partie 10201: Modèle d'information du domaine (ISO/IEEE 11073-10201:2004)

Medizinische Informatik - Kommunikation patientennaher medizinischer Geräte - Teil 10201: Bereichs-Informationsmodell (ISO/IEEE 11073-10201:2004)

This European Standard was approved by CEN on 16 August 2005.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the Central Secretariat or to any CEN member.

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

CEN members are the national standards bodies of Austria, Belgium, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Slovakia, Slovenia, Spain, Sweden, Switzerland and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

Management Centre: rue de Stassart, 36 B-1050 Brussels

© 2005 CEN All rights of exploitation in any form and by any means reserved worldwide for CEN national Members.

Ref. No. EN ISO 11073-10201:2005: E

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

Contents

Page

Foreword ... 6

1. Scope ... 7

2. Normative references ... 7

3. Definitions ... 10

4. Abbreviations and acronyms ... 14

5. General requirements ... 15

6. DIM ... 15

6.1 General ... 15
6.1.1 Modeling concept ... 15
6.1.2 Scope of the DIM ... 17
6.1.3 Approach ... 17
6.1.4 Extension of the model ... 18
6.2 Package diagram–overview ... 18
6.3 Model for the Medical Package ... 18
6.3.1 VMO (i.e., virtual medical object) ... 20
6.3.2 VMD (i.e., virtual medical device) object ... 20
6.3.3 Channel object ... 21
6.3.4 Metric object ... 21
6.3.5 Numeric object ... 21
6.3.6 Sample Array object ... 21
6.3.7 Real Time Sample Array object ... 21
6.3.8 Time Sample Array object ... 21
6.3.9 Distribution Sample Array object ... 22
6.3.10 Enumeration object ... 22
6.3.11 Complex Metric object ... 22
6.3.12 PM-Store (i.e., persistent metric) object ... 22
6.3.13 PM-Segment object ... 22
6.4 Model for the Alert Package ... 23
6.4.1 Alert object ... 23
6.4.2 Alert Status object ... 24
6.4.3 Alert Monitor object ... 24
6.5 Model for the System Package ... 25
6.5.1 VMS (i.e., virtual medical system) object ... 25
6.5.2 MDS object ... 25
6.5.3 Simple MDS object ... 26
6.5.4 Hydra MDS object ... 26
6.5.5 Composite Single Bed MDS object ... 26
6.5.6 Composite Multiple Bed MDS object ... 26
6.5.7 Log object ... 26
6.5.8 Event Log object ... 26
6.5.9 Battery object ... 26
6.5.10 Clock object ... 26
6.6 Model for the Control Package ... 27
6.6.1 SCO ... 28
6.6.2 Operation object ... 28
6.6.3 Select Item Operation object ... 28

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

6.6.4 Set Value Operation object 29
6.6.5 Set String Operation object 29
6.6.6 Toggle Flag Operation object 29
6.6.7 Activate Operation object 29
6.6.8 Limit Alert Operation object 29
6.6.9 Set Range Operation object 29
6.7 Model for the Extended Services Package 30
6.7.1 Scanner object 30
6.7.2 CfgScanner (i.e., configurable scanner) object 31
6.7.3 EpiCfgScanner (i.e., episodic configurable scanner) object 31
6.7.4 PeriCfgScanner (i.e., periodic configurable scanner) object 31
6.7.5 FastPeriCfgScanner (i.e., fast periodic configurable scanner) object 32
6.7.6 UcfgScanner (i.e., unconfigurable scanner) object 32
6.7.7 Context Scanner object 32
6.7.8 Alert Scanner object 32
6.7.9 Operating Scanner object 32
6.8 Model for the Communication Package 32
6.8.1 Communication Controller object 33
6.8.2 DCC (i.e., device communication controller) object 33
6.8.3 BCC (i.e., bedside communication controller) object 34
6.8.4 Device Interface object 34
6.8.5 MibElement object 34
6.8.6 Specialized MibElement object 34
6.9 Model for the Archival Package 34
6.9.1 Multipatient Archive object 35
6.9.2 Patient Archive object 35
6.9.3 Session Archive object 36
6.9.4 Physician object 36
6.9.5 Session Test object 36
6.9.6 Session Notes object 36
6.9.7 Ancillary object 36
6.10 Model for the Patient Package 36
6.10.1 Patient Demographics object 37
6.11 DIM—dynamic model 37
6.11.1 General 37
6.11.2 MDS communication finite state machine (FSM) 37
6.11.3 Communicating systems—startup object interaction diagram 39
6.11.4 Communication Package—MibElement data access 40
6.11.5 Dynamic object relations 40

7. DIM object definitions 42

7.1 General 42
7.1.1 Notation 42
7.1.2 Common data types 43
7.2 Top object 51
7.2.1 Attributes 51
7.2.2 Behavior 51
7.2.3 Notifications 51
7.3 Objects in the Medical Package 52
7.3.1 VMO 52
7.3.2 VMD object 53
7.3.3 Channel object 55
7.3.4 Metric object 56

AVA Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

7.3.5 Numeric object ... 62
7.3.6 Sample Array object ... 64
7.3.7 Real Time Sample Array object ... 69
7.3.8 Time Sample Array object ... 71
7.3.9 Distribution Sample Array object ... 73
7.3.10 Enumeration object ... 75
7.3.11 Complex Metric object ... 78
7.3.12 PM-Store object ... 81
7.3.13 PM-Segment object ... 84
7.4 Objects in the Alert Package ... 86
7.4.1 Alert object ... 86
7.4.2 Alert Status object ... 88
7.4.3 Alert Monitor object ... 90
7.5 Objects in the System Package ... 92
7.5.1 VMS object ... 92
7.5.2 MDS object ... 95
7.5.3 Simple MDS object ... 99
7.5.4 Hydra MDS object ... 99
7.5.5 Composite Single Bed MDS object ... 99
7.5.6 Composite Multiple Bed MDS object ... 99
7.5.7 Log object ... 99
7.5.8 Event Log object ... 101
7.5.9 Battery object ... 103
7.5.10 Clock object ... 105
7.6 Objects in the Control Package ... 111
7.6.1 SCO ... 111
7.6.2 Operation object ... 114
7.6.3 Select Item Operation object ... 116
7.6.4 Set Value Operation object ... 118
7.6.5 Set String Operation object ... 119
7.6.6 Toggle Flag Operation object ... 120
7.6.7 Activate Operation object ... 122
7.6.8 Limit Alert Operation object ... 122
7.6.9 Set Range Operation object ... 124
7.7 Objects in the Extended Services Package ... 126
7.7.1 Scanner object ... 126
7.7.2 CfgScanner object ... 127
7.7.3 EpiCfgScanner object ... 129
7.7.4 PeriCfgScanner object ... 130
7.7.5 FastPeriCfgScanner object ... 131
7.7.6 UcfgScanner object ... 132
7.7.7 Context Scanner object ... 133
7.7.8 Alert Scanner object ... 135
7.7.9 Operating Scanner object ... 136
7.8 Objects in the Communication Package ... 138
7.8.1 Communication Controller object ... 138
7.8.2 DCC object ... 141
7.8.3 BCC object ... 141
7.8.4 Device Interface object ... 141
7.8.5 MibElement object ... 142
7.8.6 Device Interface MibElement object ... 143
7.8.7 General Communication Statistics MibElement object ... 144
7.9 Objects in the Archival Package ... 146
7.9.1 Multipatient Archive object ... 146

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

7.9.2 Patient Archive object 147
7.9.3 Session Archive object 148
7.9.4 Physician object 149
7.9.5 Session Test object 151
7.9.6 Session Notes object 152
7.10 Objects in the Patient Package 153
7.10.1 Patient Demographics object 153

8. Service model for communicating systems 157

8.1 General 157
8.2 Communicating systems 157
8.3 General service model overview 158
8.3.1 Conceptual architecture of communicating systems 159
8.4 General object management services definition 160
8.4.1 EVENT REPORT service 161
8.4.2 GET service 162
8.4.3 SET service 162
8.4.4 ACTION service 163
8.4.5 CREATE service 164
8.4.6 DELETE service 165

9. MDIB nomenclature 166

10. Conformance model 167

10.1 Applicability 167
10.2 Conformance specification 167
10.3 ICSs 168
10.3.1 General format 168
10.3.2 General ICS 168
10.3.3 Service Support ICS 170
10.3.4 DIM managed object class (MOC) ICS 171
10.3.5 MOC Attribute ICS 171
10.3.6 MOC Behavior ICS 171
10.3.7 MOC Notification ICS 171

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

Foreword

The text of ISO/IEEE 11073-10201:2004 has been prepared by Technical Committee ISO/TC 215 “Health informatics” of the International Organization for Standardization (ISO) and has been taken over as EN ISO 11073-10201:2005 by Technical Committee CEN/TC 251 “Health informatics”, the secretariat of which is held by NEN.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by February 2006, and conflicting national standards shall be withdrawn at the latest by February 2006.

According to the CEN/CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Slovakia, Slovenia, Spain, Sweden, Switzerland and United Kingdom.

Endorsement notice

The text of ISO/IEEE 11073-10201:2004 has been approved by CEN as EN ISO 11073-10201:2005 without any modifications.

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

1. Scope

Within the context of the ISO/IEEE 11073 family of standards, this standard addresses the definition and structuring of information that is communicated or referred to in communication between application entities.

This standard provides a common representation of all application entities present in the application processes within the various devices independent of the syntax.

The definition of association control and lower layer communication is outside the scope of this standard.

2. Normative references

The following referenced documents are indispensable for the application of this standard. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

CEN EN 1064, Medical informatics — Standard communication protocol — computer-assisted electrocardiography.[1]

CEN ENV 12052, Medical informatics — Medical imaging communication (MEDICOM).

IEEE Std 1073™, IEEE Standard for Medical Device Communications—Overview and Framework.[2]

[1]CEN publications are available from the European Committee for Standardization (CEN), 36, rue de Stassart, B-1050 Brussels, Belgium (http://www.cenorm.be).

[2]IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, Piscataway, NJ 08854, USA (http://www.standards.ieee.org/).

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

IETF RFC 1155, Structure and Identification of Management Information for TCP/IP-Based Internets.[3]

ISO 639-1, Code for the representation of names of languages — Part 1: Alpha-2 code.[4]

ISO 639-2, Codes for the representation of names of languages — Part 2: Alpha-3 code.

ISO 3166-1, Codes for the representation of names of countries and their subdivisions — Part 1: Country codes.

ISO 3166-2, Codes for the representation of names of countries and their subdivisions — Part 2: Country subdivision code.

ISO 3166-3, Codes for the representation of names of countries and their subdivisions — Part 3: Code for formerly used names of countries.

ISO 8601, Data elements and interchange formats — Information interchange — Representation of dates and times.

ISO 15225, Nomenclature — Specification for a nomenclature system for medical devices for the purpose of regulatory data exchange.

ISO/IEC 646, Information technology — ISO 7-bit coded character set for information interchange.[5]

ISO/IEC 2022, Information technology — Character code structure and extension techniques.

ISO/IEC 5218, Information technology — Codes for the representation of human sexes.

ISO/IEC 7498-1, Information technology — Open systems interconnection — Basic reference model — Part 1: The basic model.

ISO/IEC 7498-2, Information processing systems — Open systems interconnection — Basic reference model — Part 2: Security architecture.

ISO/IEC 7498-3, Information processing systems — Open systems interconnection — Basic reference model — Part 3: Naming and addressing.

ISO/IEC 7498-4, Information processing systems — Open systems interconnection — Basic reference model — Part 4: Management framework.

ISO/IEC 8649, Information processing systems — Open systems interconnection — Service definition for the Association Control Service Element.

ISO/IEC 8650-1, Information technology — Open systems interconnection — Connection-Oriented Protocol for the Association Control Service Element — Part 1: Protocol.

---

[3]Internet requests for comment (RFCs) are available from the Internet Engineering Task Force at http://www.ietf.org/.

[4]ISO publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembé, CH-1211, Genève 20, Switzerland/Suisse (http://www.iso.ch/). ISO publications are also available in the United States from the Sales Department, American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (http://www.ansi.org/).

[5]ISO/IEC documents can be obtained from the ISO office, 1 rue de Varembé, Case Postale 56, CH-1211, Genève 20, Switzerland/Suisse (http://www.iso.ch/) and from the IEC office, 3 rue de Varembé, Case Postale 131, CH-1211, Genève 20, Switzerland/Suisse (http://www.iec.ch/). ISO/IEC publications are also available in the United States from the Sales Department, American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (http://www.ansi.org/).

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

ISO/IEC 8650-2, Information technology — Open systems interconnection — Protocol Specification for Association Control Service Element — Part 2: Protocol Implementation Conformance Statements (PICS) proforma.

ISO/IEC 8824-1, Information technology — Abstract Syntax Notation One (ASN.1) — Part 1: Specification of basic notation.

ISO/IEC 8824-2, Information technology — Abstract Syntax Notation One (ASN.1) — Part 2: Information object specification.

ISO/IEC 8859-*n*, Information processing — 8-bit single-byte coded graphic character sets — Part 1 to Part 15: Various alphabets.

ISO/IEC 9545, Information technology — Open systems interconnection — Application layer structure.

ISO/IEC 9595, Information technology — Open systems interconnection — Common management information service definition.

ISO/IEC 9596-1, Information technology — Open systems interconnection — Common Management Information Protocol — Part 1: Specification.

ISO/IEC 10040, Information technology — Open systems interconnection — Systems management overview.

ISO/IEC 10164-1, Information technology — Open systems interconnection — Systems management — Part 1: Object management function.

ISO/IEC 10164-2, Information technology — Open systems interconnection — Systems management — Part 2: State management function.

ISO/IEC 10164-3, Information technology — Open systems interconnection — System management — Part 3: Attributes for representing relationships.

ISO/IEC 10164-4, Information technology — Open systems interconnection — Systems management — Part 4: Alarm reporting function.

ISO/IEC 10164-5, Information technology — Open systems interconnection — Systems management — Part 5: Event management function.

ISO/IEC 10164-6, Information technology — Open systems interconnection — Systems management — Part 6: Log control function.

ISO/IEC 10164-7, Information technology — Open systems interconnection — Systems management — Part 7: Security alarm reporting function.

ISO/IEC 10164-8, Information technology — Open systems interconnection — Systems management — Part 8: Security audit trail function.

ISO/IEC 10164-9, Information technology — Open systems interconnection — Systems management — Part 9: Objects and attributes for access control.

ISO/IEC 10164-10, Information technology — Open systems interconnection — Systems management — Part 10: Usage metering function for accounting purposes.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

ISO/IEC 10164-11, Information technology — Open systems interconnection — Systems management — Part 11: Metric objects and attributes.

ISO/IEC 10164-12, Information technology — Open systems interconnection — Systems management — Part 12: Test management function.

ISO/IEC 10164-13, Information technology — Open systems interconnection — Systems management — Part 13: Summarization function.

ISO/IEC 10164-14, Information technology — Open systems interconnection — Systems management — Part 14: Confidence and diagnostic test categories.

ISO/IEC 10165-1, Information technology — Open systems interconnection — Structure of management information — Part 1: Management information model.

ISO/IEC 10165-2, Information technology — Open systems interconnection — Structure of management information — Part 2: Definition of management information.

ISO/IEC 10646-1, Information technology — Universal multiple-octet coded character set (UCS) — Part 1: Architecture and basic multilingual plane.

ISO/IEEE 11073-10101, Health informatics — Point-of-care medical device communication — Part 10101: Nomenclature.

ISO/IEEE 11073-20101, Health informatics — Point-of-care medical device communication — Part 20101: Application profiles – Base standard.

NEMA PS 3, Digital imaging and communications in medicine (DICOM).[6]

3. Definitions

For the purpose of this standard, the following definitions apply. *The Authoritative Dictionary of IEEE Standards Terms*, Seventh Edition, should be referenced for terms not defined in this clause.

3.1 agent: Device that provides data in a manager-agent communicating system.

3.2 alarm: Signal that indicates abnormal events occurring to the patient or the device system.

3.3 alert: Synonym for the combination of patient-related physiological alarms, technical alarms, and equipment-user advisory signals.

3.4 alert monitor: Object representing the output of a device or system alarm processor and as such the overall device or system alarm condition.

3.5 alert status: Object representing the output of an alarm process that considers all alarm conditions in a scope that spans one or more objects.

3.6 archival: Relating to the storage of data over a prolonged period.

[6]NEMA publications are available from Global Engineering Documents, 15 Inverness Way East, Englewood, Colorado 80112, USA (http://global.ihs.com/).

AVA Documents Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

3.7 association control service element (ACSE): Method used to establish logical connections between medical device systems (MDSs).

3.8 channel: Umbrella object in the object model that groups together physiological measurement data and data derived from these data.

3.9 class: Description of one or more objects with a uniform set of attributes and services including a description of how to create new objects in the class.

3.10 communication controller: Part of a medical device system (MDS) responsible for communications.

3.11 communication party: Actor of the problem domain that participates in the communication in that domain.

3.12 communication role: Role of a party in a communication situation defining the party's behavior in the communication. Associated with a communication role is a set of services that the party provides to other parties.

3.13 data agent: As a medical device, a patient data acquisition system that provides the acquired data for other devices.

3.14 data format: Arrangement of data in a file or stream.

3.15 data logger: A medical device that is functioning in its capacity as a data storage and archival system.

3.16 data structure: Manner in which application entities construct the data set information resulting from the use of an information object.

3.17 dictionary: Description of the contents of the medical data information base (MDIB) containing vital signs information, device information, demographics, and other elements of the MDIB.

3.18 discrete parameter: Vital signs measurement that can be expressed as a single numeric or textual value.

3.19 domain information model (DIM): The model describing common concepts and relationships for a problem domain.

3.20 event; A change in device status that is communicated by a notification reporting service.

3.21 event report: Service [provided by the common medical device information service element (CMDISE)] to report an event relating to a managed object instance.

3.22 framework: A structure of processes and specifications designed to support the accomplishment of a specific task.

3.23 graphic parameter: Vital signs measurement that requires a number of regularly sampled data points in order to be expressed properly.

3.24 host system: Term used as an abstraction of a medical system to which measurement devices are attached.

3.25 information object: An abstract data model applicable to the communication of vital signs information and related patient data. The attributes of an information object definition describe its properties. Each

AVA Documents Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

information object definition does not represent a specific instance of real-world data, but rather a class of data that share the same properties.

3.26 information service element: Instances in the medical data information base (MDIB).

3.27 instance: The realization of an abstract concept or specification, e.g., object instance, application instance, information service element instance, virtual medical device (VMD) instance, class instance, operating instance.

3.28 intensive care unit (ICU): The unit within a medical facility in which patients are managed using multiple modes of monitoring and therapy.

3.29 interchange format: The representation of the data elements and the structure of the message containing those data elements while in transfer between systems. The interchange format consists of a data set of construction elements and a syntax. The representation is technology specific.

3.30 interoperability: Idealized scheme where medical devices of differing types, models, or manufacturers are capable of working with each other, whether connected to each other directly or through a communication system.

3.31 latency: In a communications scenario, the time delay between sending a signal from one device and receiving it by another device.

3.32 lower layers: Layer 1 to Layer 4 of the International Organization for Standardization (ISO)/open systems interconnection (OSI) reference model. These layers cover mechanical, electrical, and general communication protocol specifications.

3.33 manager: Device that receives data in a manager-agent communicating system.

3.34 manager-agent model: Communication model where one device (i.e., agent) provides data and another device (i.e., manager) receives data.

3.35 medical data information base (MDIB): The concept of an object-oriented database storing (at least) vital signs information.

3.36 medical device: A device, apparatus, or system used for patient monitoring, treatment, or therapy, which does not normally enter metabolic pathways. For the purposes of this standard, the scope of medical devices is further limited to patient-connected medical devices that provide support for electronic communications.

3.37 medical device system (MDS): Abstraction for system comprising one or more medical functions. In the context of this standard, the term is specifically used as an object-oriented abstraction of a device that provides medical information in the form of objects that are defined in this standard.

3.38 monitor: A medical device designed to acquire, display, record, and/or analyze patient data and to alert caregivers of events needing their attention.

3.39 object: A concept, an abstraction, or a thing with crisp boundaries and a meaning for the problem at hand.

3.40 object attributes: Data that, together with methods, define an object.

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

3.41 object class: A descriptor used in association with a group of objects with similar properties (i.e., attributes), common behavior (i.e., operations), common relationships to other objects, and common semantics.

3.42 object diagram: Diagram showing connections between objects in a system.

3.43 object method: A procedure or process acting upon the attributes and states of an object class.

3.44 object-oriented analysis: Method of analysis where the problem domain is modelled in the form of objects and their interactions.

3.45 open system: A set of protocols allowing computers of different origins to be linked together.

3.46 operation: A function or transformation that may be applied to or by objects in a class (sometimes also called service).

3.47 problem domain; The field of health care under consideration in a modeling process.

3.48 protocol: A standard set of rules describing the transfer of data between devices. It specifies the format of the data and specifies the signals to start, control, and end the transfer.

3.49 scanner: An observer and "summarizer" of object attribute values.

3.50 scenario: A formal description of a class of business activities including the semantics of business agreements, conventions, and information content.

3.51 service: A specific behavior that a communication party in a specific role is responsible for exhibiting.

3.52 syntax (i.e., of an interchange format): The rules for combining the construction elements of the interchange format.

3.53 system: The demarcated part of the perceivable universe, existing in time and space, that may be regarded as a set of elements and relationships between these elements.

3.54 timestamp: An attribute or field in data that denotes the time of data generation.

3.55 top object: The ultimate base class for all other objects belonging to one model.

3.56 upper layers: Layer 5 to Layer 7 of the International Organization for Standardization (ISO)/open systems interconnection (OSI) reference model. These layers cover application, presentation, and session specifications and functionalities.

3.57 virtual medical device (VMD): An abstract representation of a medical-related subsystem of a medical device system (MDS).

3.58 virtual medical object (VMO): An abstract representation of an object in the Medical Package of the domain information model (DIM).

3.59 vital sign: Clinical information, relating to one or more patients, that is measured by or derived from apparatus connected to the patient or otherwise gathered from the patient.

3.60 waveform: Graphic data, typically vital signs data values varying with respect to time, usually presented to the clinician in a graphical form.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

4. Abbreviations and acronyms

ACSE	association control service element
ASN.1	Abstract Syntax Notation One
BCC	bedside communication controller
BER	basic encoding rules
BMP	basic multilingual plane
CMDIP	Common Medical Device Information Protocol
CMDISE	common medical device information service element
CMIP	Common Management Information Protocol
CMISE	common management information service element
DCC	device communication controller
DICOM	digital imaging and communications in medicine
DIM	domain information model
ECG	electrocardiogram
EEG	electroencephalogram
EBWW	eyeball and wristwatch
FSM	finite state machine
GMDN	Global Medical Device Nomenclature
GMT	Greenwich mean time
IANA	Internet Assigned Numbers Authority
ICS	implementation conformance statement
ICU	intensive care unit
ID	identifier or identification
LAN	local area network
LSB	least significant bit
MDIB	medical data information base
MDS	medical device system
MEDICOM	medical imaging communication
MIB or Mib	management information base
MOC	managed object class
OID	object identifier
OR	operating room
OSI	open systems interconnection
PC	personal computer
PDU	protocol data unit
PM	persistent metric
SCADA	supervisory control and data acquisition
SCP ECG	Standard Communications Protocol [for computer-assisted] Electrocardiography
SNMP	Simple Network Management Protocol
SNTP	Simple Network Time Protocol
UML	unified modeling language
UTC	coordinated universal time
VMD	virtual medical device
VMO	virtual medical object
VMS	virtual medical system

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

5. General requirements

The ISO/IEEE 11073 family of standards is intended to enable medical devices to interconnect and interoperate with other medical devices and with computerized healthcare information systems in a manner suitable for the clinical environment.

The ISO/IEEE 11073 family is based on an object-oriented systems management paradigm. Data (e.g., measurement, state) is modeled in the form of information objects that can be accessed and manipulated using an object access service protocol.

The domain information model (DIM) defines the overall set of information objects and their attributes, methods, and access functions needed for medical device communication.

The top-level user requirements are defined in IEEE Std 1073,[7] which also defines the user scenarios covered by the ISO/IEEE 11073 family of standards.

As a part of the ISO/IEEE 11073 family of standards, the primary requirements for the DIM are as follows:

— Define an object-oriented model representing the relevant information (i.e., data) and functions (e.g., device controls) encountered in the problem domain of medical device communication, including measurement information, contextual data, device control methods, and other relevant aspects.

— Provide detailed specification of the information objects defined in the object-oriented model, including their attributes and methods.

— Define a service model for communicating medical devices that provides access to the information objects, their attributes, and their methods.

— Use the nomenclature defined in ISO/IEEE 11073-10101 to identify all data elements in the model.

— Be usable for the definition of data communication protocols as well as for the definition of file storage formats.

— Define conformance requirements.

— Be extensible and expandable to incorporate future needs in the defined modeling framework.

6. DIM

6.1 General

6.1.1 Modeling concept

The DIM is an object-oriented model that consists of objects, their attributes, and their methods, which are abstractions of real-world entities in the domain of (vital signs information communicating) medical devices.

The information model and the service model for communicating systems defined and used in this standard are conceptually based on the International Organization for Standardization (ISO)/open systems interconnection (OSI) system management model. Objects defined in the information model are considered managed (here, medical) objects. For the most part, they are directly available to management (i.e., access) services provided by the common medical device information service element (CMDISE) as defined in this standard.

[7] Information on references can be found in Clause 2.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

For communicating systems, the set of object instances available on any medical device that complies with the definitions of this standard forms the medical data information base (MDIB). The MDIB is a structured collection of managed medical objects representing the vital signs information provided by a particular medical device. Attribute data types, hierarchies, and behavior of objects in the MDIB are defined in this standard.

The majority of objects defined here represent generalized vital signs data and support information. Specialization of these objects is achieved by defining appropriate attributes. Object hierarchies and relations between objects are used to express device configuration and device capabilities.

Example: A generalized object is defined to represent vital signs in the form of a real-time waveform. A set of object attributes is used to specify a particular waveform as an invasive arterial blood pressure. The position in the hierarchy of all objects defines the subsystem that derives the waveform.

Figure 6.1 shows the relation between managed medical objects, MDIB, CMDISE, application processes, and communication systems.

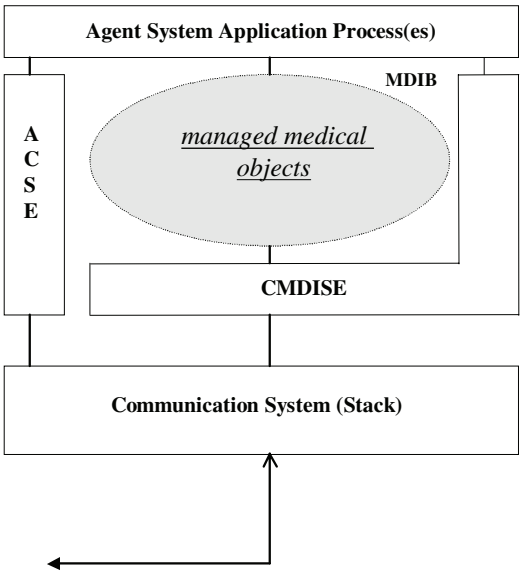

Figure 6.1—MDIB in communicating systems

In the case of communicating systems, managed medical objects are accessible only through services provided by CMDISE. The way that these objects are stored in the MDIB in any specific system and the way applications and the CMDISE access these objects are implementation issues and as such not normative.

In the case of a vital signs archived data format that complies with the definitions in this standard, object instances are stored, together with their dynamic attribute value changes, over a certain time period on archival media. Attribute data types and hierarchies of objects in the archival data format are again defined in this standard.

Figure 6.2 shows the relationship between managed medical objects, the data archive, and archive access services.

In the case of the archived data format, the way managed medical objects are stored on a medium is the subject of standardization. The access services are a local implementation issue and as such are not intended to be governed by this standard.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

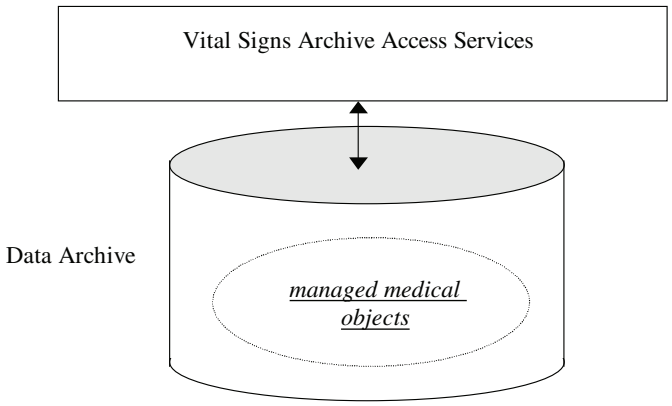

Figure 6.2—Managed medical objects in a vital signs archive

6.1.2 Scope of the DIM

6.1.2.1 General

Vital signs information objects that are defined in this standard encompass digitized biosignals that are derived by medical measurement devices used, for example, in anaesthesia, surgery, infusion therapy, intensive care, and obstetrical care.

Biosignal data within the scope of this standard include direct and derived, quantitative and qualitative measurements, technical and medical alarms, and control settings. Patient information relevant for the interpretation of these signals is also defined in the DIM.

6.1.2.2 Communicating systems

Communicating systems within the scope of this standard include physiological meters and analysers, especially systems providing real-time or continuous monitoring. Data processing capabilities are required for these systems.

Information management objects that provide capabilities and concepts for cost-effective communication (specifically data summarization objects) and objects necessary to enable real-time communication are also within the scope of the information model in this standard.

Interoperability issues, specifically lower communication layers, temporal synchronization between multiple devices, etc., are outside the scope of this standard.

6.1.2.3 Archived vital signs

Context information objects that describe the data acquisition process and organize a vital signs archive are within the scope of this standard.

6.1.3 Approach

For the object-oriented modeling, the unified modeling language (UML) technique is used. The domain is first subdivided into different packages, and this subdivision permits the organization of the model into smaller packages. Each package is then defined in the form of object diagrams. Objects are briefly introduced, and their relations and hierarchies are defined in the object diagram.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

For the object definitions, a textual approach is followed. Attributes are defined in attribute definition tables. Attribute data types are defined using Abstract Syntax Notation One (ASN.1). Object behavior and notifications generated by objects are also defined in definition tables. These definitions directly relate to the service model specified in Clause 8.

6.1.4 Extension of the model

It is expected that over time extensions of the model may be needed to account for new developments in the area of medical devices. Also, in special implementations, there may be a requirement to model data that are specific for a particular device or a particular application (and that are, therefore, not covered by the general model).

In some cases, it may be possible to use the concept of *external object relations*. Most objects defined in this standard provide an attribute group (e.g., the Relationship Attribute Group) that can be used to supply information about related objects that are not defined in the DIM. Supplying such information can be done by specifying a relation to an external object and assigning attributes to this relation (see 7.1.2.20).

In other cases, it may be necessary to define completely new objects or to add new attributes, new methods, or new events to already defined objects. These extensions are considered private or manufacturer-specific extensions. Dealing with these extensions is primarily a matter of an interoperability standard that is based on this standard on vital signs representation.

In general, in an interoperability format, objects, attributes, and methods are identified by nomenclature codes. The nomenclature code space (i.e., code values) leaves room for private extensions. As a general rule, an interoperability standard that is based on this DIM should be able to deal with private or manufacturer-specific extensions by ignoring objects, attributes, etc., with unknown identifiers (i.e., nomenclature codes).

6.2 Package diagram–overview

The package diagram organizes the problem domain into separate groups. It shows the major objects inside each package and defines the relationships between these packets.

The package diagram depicted in Figure 6.3 contains only a small subset of all objects defined in the DIM. Common base objects except the Top object are not shown in this diagram. Also, not all relations are shown between the different packages. Refer to the detailed package diagrams for more information.

The numbers in the packages refer to the corresponding subclauses in this clause about models and in Clause 7 about the object definitions.

The Top object is an abstract base class and at the same time the ultimate base class for all objects defined in the model. For editorial convenience, the modeling diagrams in this standard do not show this inheritance hierarchy.

The more detailed models for these packages are contained in 6.3 through 6.10.

6.3 Model for the Medical Package

The Medical Package deals with the derivation and representation of biosignals and contextual information that is important for the interpretation of measurements.

Figure 6.4 shows the object model of the Medical Package.

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

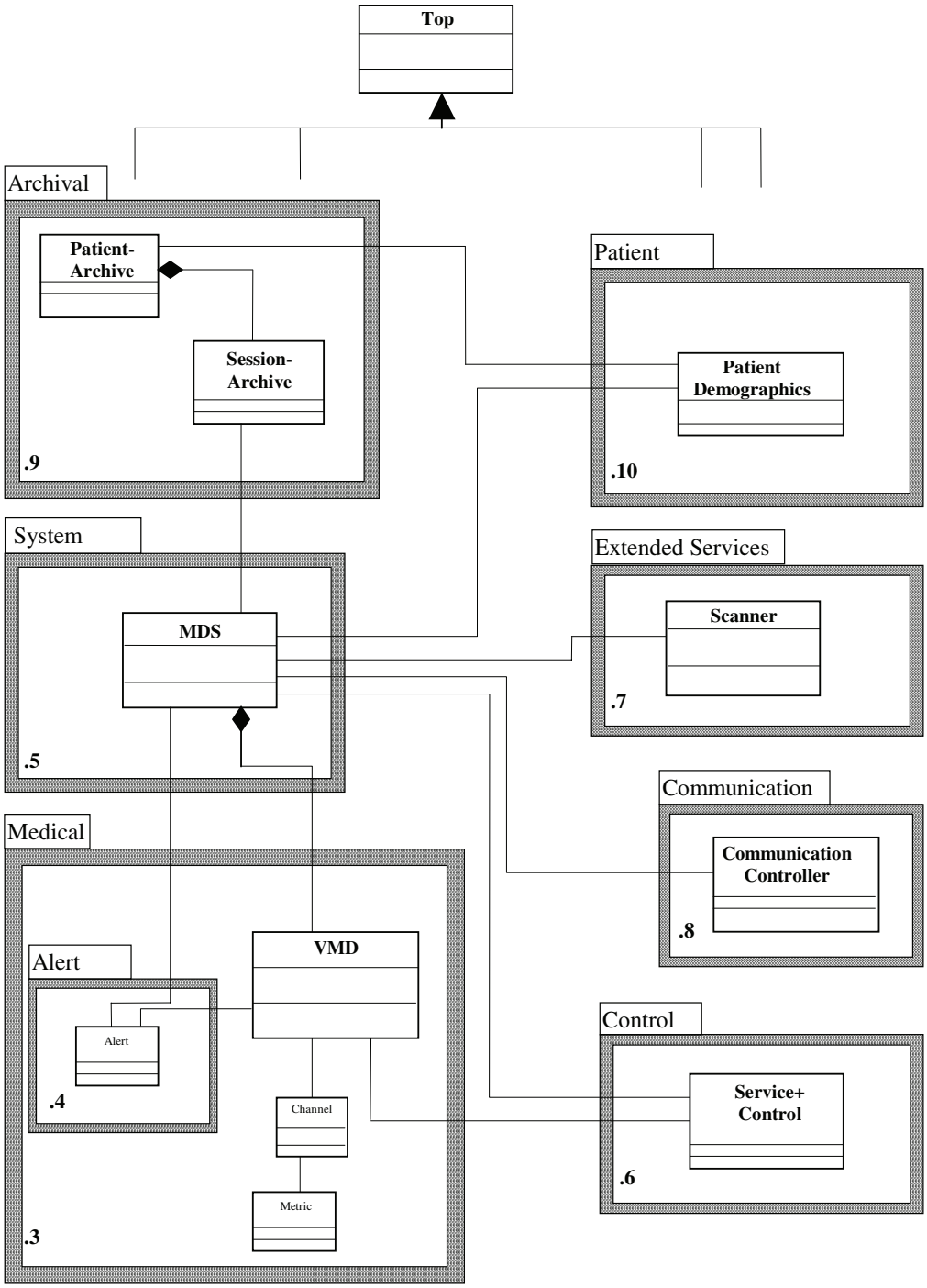

Figure 6.3—Packages of the DIM

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

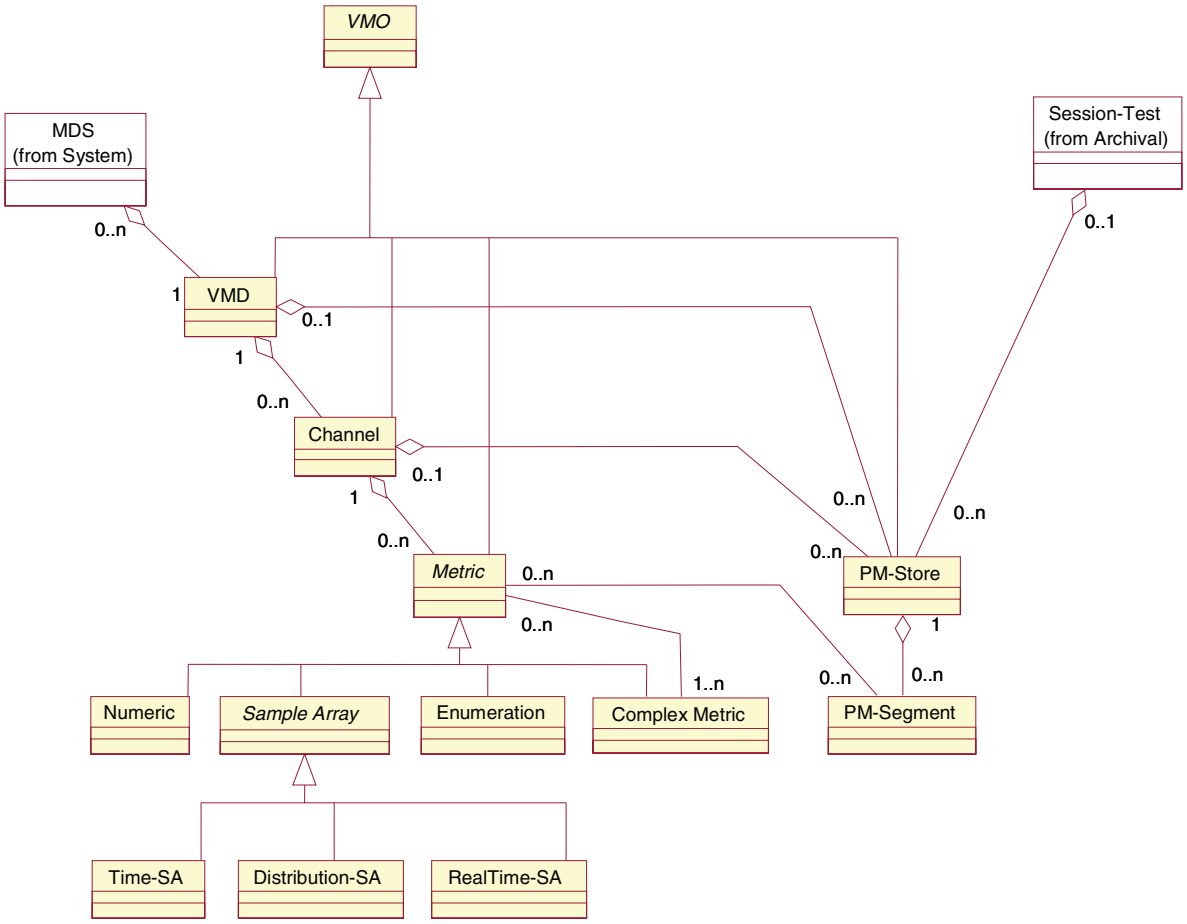

Figure 6.4—Medical Package model

NOTE—Instances of the Channel object and the PM-Store object shall be contained in exactly one superior object instance. Refer to the Alert Package for information about alarm-related objects.[8]

The Medical Package model contains the objects described in 6.3.1 through 6.3.13.

6.3.1 VMO (i.e., virtual medical object)

The VMO is the base class for all medical-related objects in the model. It provides consistent naming and identification across the Medical Package model.

As a base class, the VMO cannot be instantiated.

6.3.2 VMD (i.e., virtual medical device) object

The VMD object is an abstraction for a medical-related subsystem (e.g., hardware or even pure software) of a medical device. Characteristics of this subsystem (e.g., modes, versions) are captured in this object. At the same time, the VMD object is a container for objects representing measurement and status information.

[8]Notes in text, tables, and figures are given for information only, and do not contain requirements needed to implement the standard.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

Example: A modular patient monitor provides measurement modalities in the form of plug-in modules. Each module is represented by a VMD object.

6.3.3 Channel object

The Channel object is used for grouping Metric objects and, thus, allows hierarchical information organization. The Channel object is not mandatory for representation of Metric objects in a VMD.

Example: A blood pressure VMD may define a Channel object to group together all metrics that deal with the blood pressure (e.g., pressure value, pressure waveform). A second Channel object can be used to group together metrics that deal with heart rate.

6.3.4 Metric object

The Metric object is the base class for all objects representing direct and derived, quantitative and qualitative biosignal measurement, status, and context data.

Specializations of the Metric object are provided to deal with common representations (e.g., single values, array data, status indications) and presentations (e.g., on a display) of measurement data.

As a base class, the Metric object cannot be instantiated.

6.3.5 Numeric object

The Numeric object represents numerical measurements and status information, e.g., amplitude measures, counters.

Example: A heart rate measurement is represented by a Numeric object.

NOTE—A compound Numeric object is defined as an efficient model, for example, for arterial blood pressure, which usually has three associated values (i.e., systolic, diastolic, mean). The availability of multiple values in a single Numeric (or other Metric) object can be indicated in a special structure attribute in the Metric object.

6.3.6 Sample Array object

The Sample Array object is the base class for metrics that have a graphical, curve type presentation and, therefore, have their observation values reported as arrays of data points by communicating systems.

As a base class, the Sample Array object cannot be instantiated.

6.3.7 Real Time Sample Array object

The Real Time Sample Array object is a sample array that represents a real-time continuous waveform. As such, it has special requirements in communicating systems, e.g., processing power, low latency, high bandwidth. Therefore, it requires the definition of a specialized object.

Example: An electrocardiogram (ECG) real-time wave is represented as a Real Time Sample Array object.

6.3.8 Time Sample Array object

The Time Sample Array object is a sample array that represents noncontinuous waveforms (i.e., a wave snippet). Within a single observation (i.e., a single array of sample values), samples are equidistant in time.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

Example: Software for ST segment analysis may use the Time Sample Array object to represent snippets of ECG real-time waves that contain only a single QRS complex. Within this wave snippet, the software can locate the ST measurement points. It generates a new snippet, for example, every 15 s.

6.3.9 Distribution Sample Array object

The Distribution Sample Array object is a sample array that represents linear value distributions in the form of arrays containing scaled sample values. The index of a value within an observation array denotes a spatial value, not a time point.

Example: An electroencephalogram (EEG) application may use a Fourier transformation to derive a frequency distribution (i.e., a spectrum) from the EEG signal. It then uses the Distribution Sample Array object to represent that spectrum in the MDIB.

6.3.10 Enumeration object

The Enumeration object represents status information and/or annotation information. Observation values may be presented in the form of normative codes (that are included in the nomenclature defined in this standard or in some other nomenclature scheme) or in the form of free text.

Example: An ECG rhythm qualification may be represented as an Enumeration object. A ventilator may provide information about its current ventilation mode as an Enumeration object.

6.3.11 Complex Metric object

In special cases, the Complex Metric object can be used to group a larger number of strongly related Metric objects in one single container object for performance or for modeling convenience. The Complex Metric object is a composition of Metric objects, possibly recursive.

Example: A ventilator device may provide extensive breath analysis capabilities. For each breath, it calculates various numerical values (e.g., volumes, I:E ratio, timing information) as well as enumerated information (e.g., breath type classification, annotation data). For efficiency, all this information is grouped together in one Complex Metric object instance, which is updated upon each breath.

6.3.12 PM-Store (i.e., persistent metric) object

The PM-Store object provides long-term storage capabilities for metric data. It contains a variable number of PM-Segment objects that can be accessed only through the PM-Store object. Without further specialization, the PM-Store object is intended to store data of a single Metric object only.

Example: A device stores the numerical value of an invasive blood pressure on a disk. It uses the PM-Store object to represent this persistent information. Attributes of the PM-Store object describe the sampling period, the sampling algorithm, and the storage format. When the label of the pressure measurement is changed (e.g., during a wedge procedure), the storage process opens a new PM-Segment to store the updated context data (here: the label).

6.3.13 PM-Segment object

The PM-Segment object represents a continuous time period in which a metric is stored without any changes of relevant metric context attributes (e.g., scales, labels).

The PM-Segment object is accessible only through the PM-Store object (e.g., for retrieving stored data, the PM-Store object has to be accessed).

AVA Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

6.4 Model for the Alert Package

The Alert Package deals with objects that represent status information about patient condition and/or technical conditions influencing the measurement or device functioning. Alert-related information is often subject to normative regulations and, therefore, requires special handling.

In the model, all alarm-related object-oriented items are identified by the term *alert*. The term *alert* is used in this standard as a synonym for the combination of patient-related physiological alarms, technical alarms, and equipment user-advisory signals.

An alarm is a signal that indicates abnormal events occurring to the patient or the device system. A physiological alarm is a signal that either indicates that a monitored physiological parameter is out of specified limits or indicates an abnormal patient condition. A technical alarm is a signal that indicates a device system is either not capable of accurately monitoring the patient's condition or no longer monitoring the patient's condition.

The model defines three different levels of alarming. These levels represent different sets of alarm processing steps, ranging from a simple context-free alarm event detection to an intelligent device system alarm process. This process is required to prioritize all device alarms, to latch alarms if needed (a latched alarm does not stop when the alarm condition goes away), and to produce audible and visual alarm indications for the user.

For consistent system-wide alarming, a particular medical device may provide either no alarming capability or exactly one level of alarming, which is dependent on the capabilities of the device. Each level is represented by one specific object class. In other words, either zero or one alarm object class (e.g., only Alert or only Alert Status or only Alert Monitor; no combinations) is instantiated in the device containment tree. Multiple instances of a class are allowed.

NOTE—Medical device alarming is subject to various national and international safety standards (e.g., IEC 60601 series, ISO 9703 series). Considering requirements of current safety standards, objects in this standard define information contents only. Any implementation shall, therefore, follow appropriate standards for dynamic alarming behavior.

Figure 6.5 shows the object model of the Alert Package.

NOTE—Instances of objects in the Alert Package area shall be contained in exactly one superior object.

The Alert Package model contains the objects described in 6.4.1 through 6.4.3.

6.4.1 Alert object

The Alert object stands for the status of a simple alarm condition check. As such, it represents a single alarm only. The alarm can be either a physiological alarm or a technical alarm condition of a related object [e.g., MDS (i.e., medical device system), VMD, Metric]. If a device instantiates an Alert object, it shall not instantiate the Alert Status or the Alert Monitor object. A single Alert object is needed for each alarm condition that the device is able to detect.

The Alert object has a reference to an object instance in the Medical Package to which the alarm condition relates.

NOTE—An Alert object instance is not dynamically created or deleted in cases where alarm conditions start or stop. Rather, an existing Alert object instance changes attribute values in these cases.

Example: An Alert object may represent the status of a process that checks for a limit violation physiological alarm of the heart rate signal. In the case of a violation of the limit, the object generates an event (i.e., attribute update) that represents this alarm condition in the form of attribute value changes.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

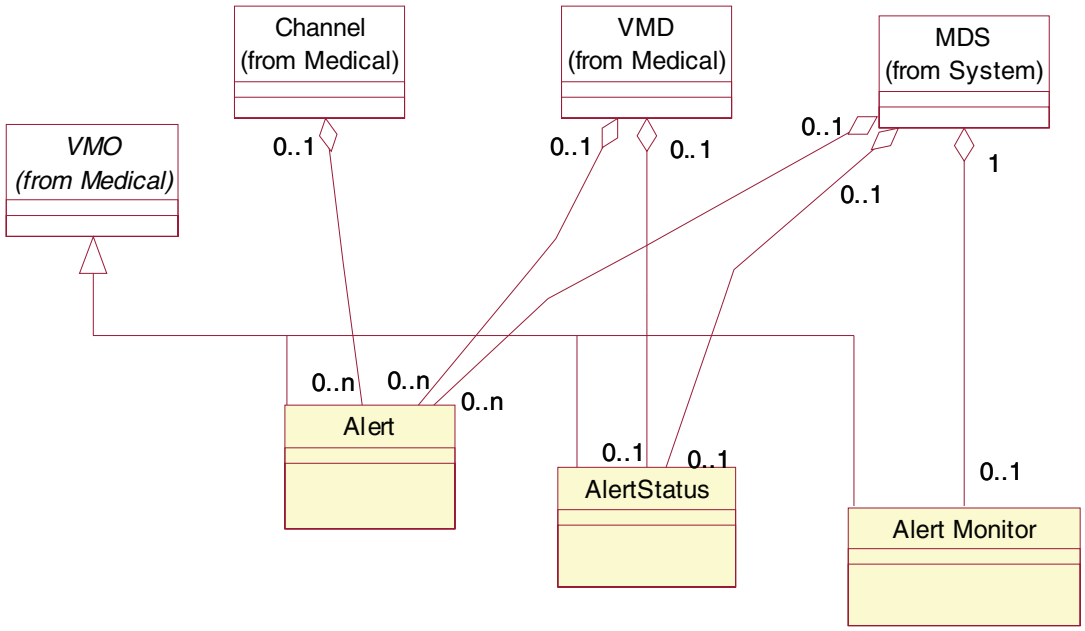

Figure 6.5—Alert Package model

6.4.2 Alert Status object

The Alert Status object represents the output of an alarm process that considers all alarm conditions in a scope that spans one or more objects. In contrast to the Alert object, the Alert Status object collects all alarm conditions related to a VMD object hierarchy or related to an MDS object and provides this information in list-structured attributes. Collecting all alarms together allows the implementation of first-level alarm processing where knowledge about the VMD or MDS can be used to prioritize alarm conditions and to suppress known false alarm indications.

For larger scale devices without complete alarm processing, the Alert Status object greatly reduces the overhead of a large number of Alert object instances.

If a device instantiates an Alert Status object, it shall not instantiate the Alert or the Alert Monitor object. Each VMD or MDS in the MDIB is able to contain at most one Alert Status object instance.

> **Example:** An ECG VMD derives a heart rate value. As the VMD is able to detect that the ECG leads are disconnected from the patient, its Alert Status object reports only a technical alarm and suppresses a heart rate limit violation alarm in this case.

6.4.3 Alert Monitor object

The Alert Monitor object represents the output of a device or system alarm processor. As such, it represents the overall device or system alarm condition and provides a list of all alarm conditions of the system in its scope. This list includes global state information and individual alarm state information that allows the implementation of a safety-standard-compliant alarm display on a remote system.

If a device instantiates an Alert Monitor object, it shall not instantiate the Alert or the Alert Status object. An MDS shall contain not more than one Alert Monitor object instance.

> **Example:** A patient-monitoring system provides alarm information in the form of an Alert Monitor object to a central station. Alert information includes the current global maximum severity of audible

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

and visual alarm conditions on the monitor display as well as a list of active technical and physiological alarm conditions. The alarm processor operates in a latching mode where physiological alarm conditions are buffered until they are explicitly acknowledged by a user.

6.5 Model for the System Package

The System Package deals with the representation of devices that derive or process vital signs information and comply with the definitions in this standard.

Figure 6.6 shows the object model for the System Package.

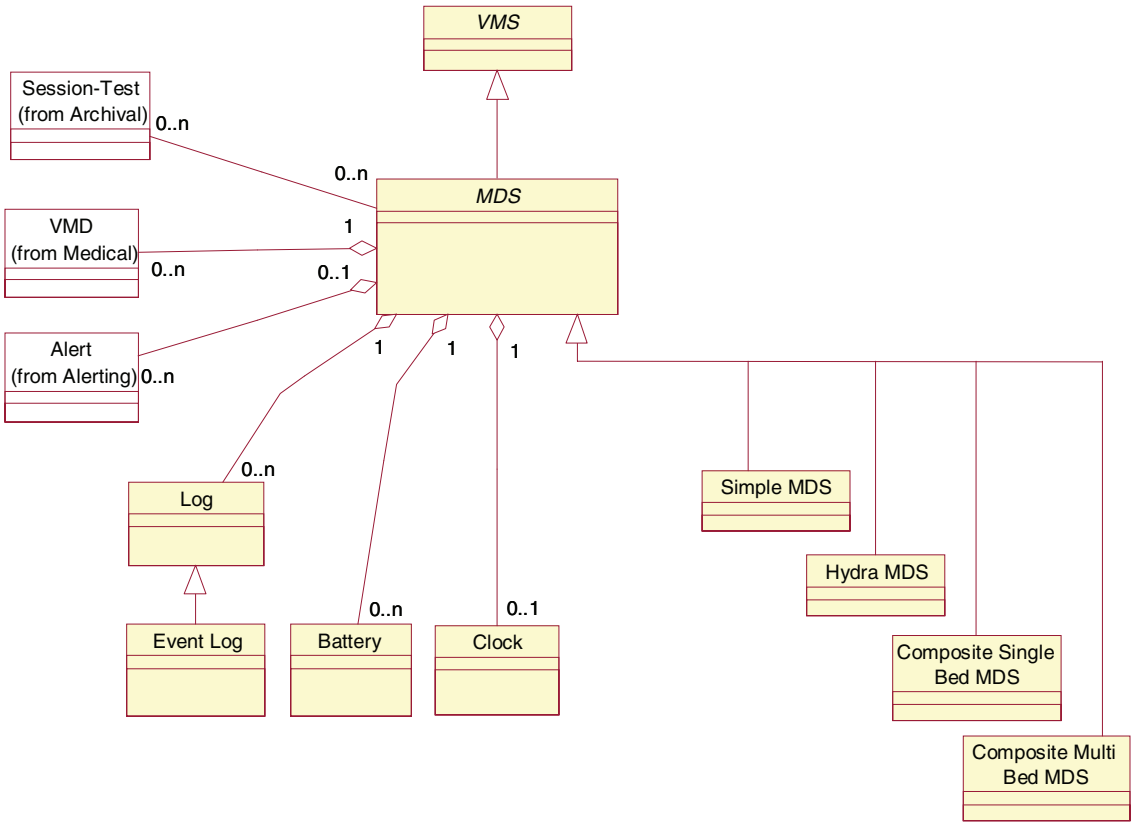

Figure 6.6—System Package model

The System Package model contains the objects described in 6.5.1 through 6.5.10.

6.5.1 VMS (i.e., virtual medical system) object

The VMS object is the abstract base class for all System Package objects in this model. It provides consistent naming and identification of system-related objects.

As a base class, the VMS object cannot be instantiated.

6.5.2 MDS object

The MDS object is an abstraction of a device that provides medical information in the form of objects that are defined in the Medical Package of the DIM.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

The MDS object is the top-level object in the device's MDIB and represents the instrument itself. Composite devices may contain additional MDS objects in the MDIB.

Further specializations of this class are used to represent differences in complexity and scope.

As a base class, the MDS object cannot be instantiated.

6.5.3 Simple MDS object

The Simple MDS object represents a medical device that contains a single VMD instance only (i.e., a single-purpose device).

6.5.4 Hydra MDS object

The Hydra MDS object represents a device that contains multiple VMD instances (i.e., a multipurpose device).

6.5.5 Composite Single Bed MDS object

The Composite Single Bed MDS object represents a device that contains (or interfaces with) one or more Simple or Hydra MDS objects at one location (i.e., a bed).

6.5.6 Composite Multiple Bed MDS object

The Composite Multiple Bed MDS object represents a device that contains (or interfaces with) multiple MDS objects at multiple locations (i.e., multiple beds).

6.5.7 Log object

The Log object is a base class that is a storage container for important local system notifications and events. It is possible to define specialized classes for specific event types.

As a base class, the Log object cannot be instantiated.

6.5.8 Event Log object

The Event Log object is a general Log object that stores system events in a free-text representation.

> **Example:** An infusion device may want to keep track of mode and rate changes by remote systems. When a remote operation is invoked, it creates an entry in its event log.

6.5.9 Battery object

For battery-powered devices, some battery information is contained in the MDS object in the form of attributes. If the battery subsystem is either capable of providing more information (i.e., a smart battery) or manageable, then a special Battery object is provided.

6.5.10 Clock object

The Clock object provides additional capabilities for handling date-related and time-related information beyond the basic capabilities of an MDS object. It models the real-time clock capabilities of an MDS object.

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

The Clock object is used in applications where precise time synchronization of medical devices is needed. This object provides resolution and accuracy information so that applications can synchronize real-time data streams between devices.

6.6 Model for the Control Package

The Control Package contains objects that allow remote measurement control and device control.

The model for remote control defined in this standard provides the following benefits:

- A system that allows remote control is able to explicitly register which attributes or features can be accessed or modified by a remote system.
- For attributes that can be remotely modified, a list of possible legal attribute values is provided to the controlling system.
- It is not mandatory that a remote-controllable item correspond to an attribute of an medical object.
- Dependence of a controllable item on internal system states is modeled.
- A simple locking transaction scheme allows the handling of transient states during remote control.

At least two different uses of remote control are considered:

- Automatic control may be done by some processes running on the controlling device. Such a process has to be able to discover automatically how it can modify or access the controllable items to provide its function.
- It is also possible to use remote control to present some form of control interface to a human operator. For this use, descriptions of functions, and possibly help information, need to be provided.

The basic concept presented here is based on Operation objects. An Operation object allows modification of a virtual attribute. This virtual attribute may, for example, be a measurement label, a filter state (on/off), or a gain factor. The attribute is called *virtual* because it need not correspond to any attribute in other objects instantiated in the system.

Different specializations of the Operation object define how the virtual attribute is modified. A Select Item operation, for example, allows the selection of an item from a given list of possible item values for the attribute. A Set Value operation allows the setting of the attribute to a value from a defined range with a specific step width (i.e., resolution).

The idea is that the Operation object provides all necessary information about legal attribute values. Furthermore, the Operation object defines various forms of text string to support a human user of the operation. It also contains grouping information that allows logical grouping of multiple Operation objects together when they are presented as part of a human interface.

Operation objects cannot directly be accessed by services defined in the service model in Clause 8. Instead, all controls shall be routed through the SCO (i.e., service and control object). This object supports a simple locking mechanism to prevent side effects caused by simultaneous calls.

The SCO groups together all Operation objects that belong to a specific entity (i.e., MDS, VMD). The SCO also allows feedback to a controlled device, for example, for a visual indication that the device is currently remote-controlled.

Figure 6.7 shows the object model for the Control Package:

The Control Package model contains the objects described in 6.6.1 through 6.6.9.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

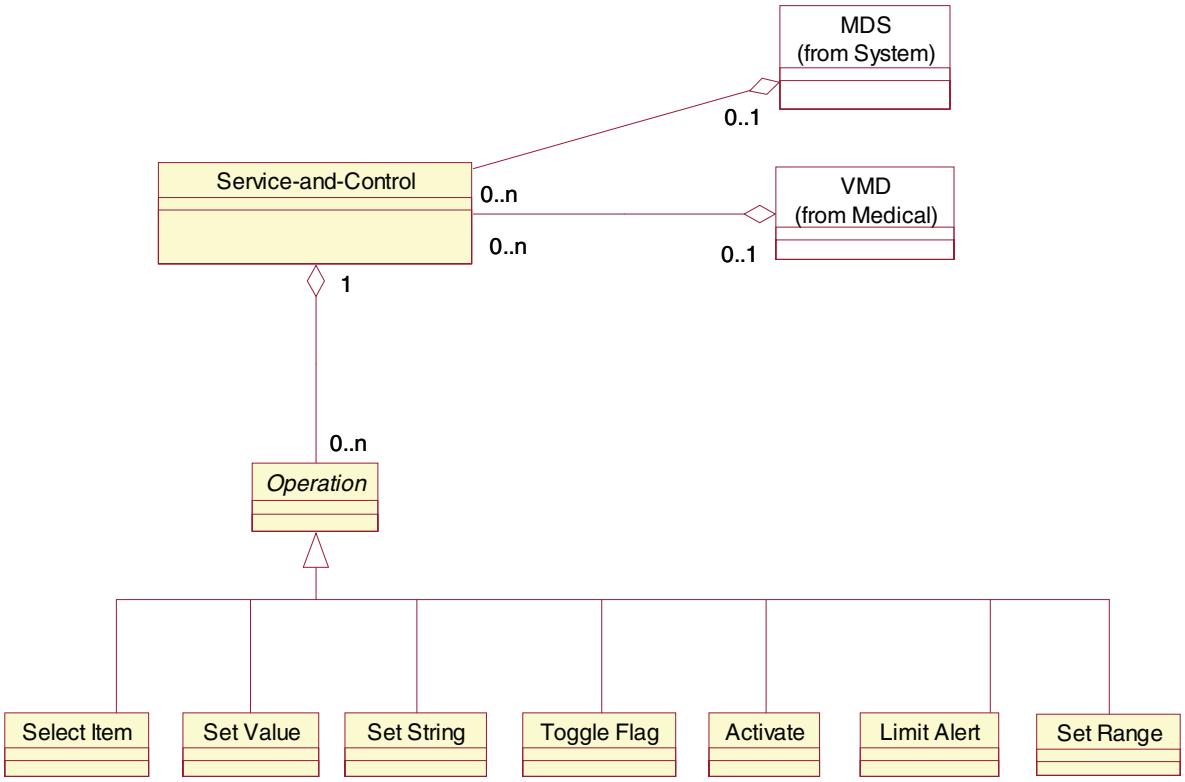

Figure 6.7—Control Package model

6.6.1 SCO

The SCO is responsible for managing all remote-control capabilities that are supported by a medical device.

Remote control in medical device communication is sensitive to safety and security issues. The SCO provides means for the following:

a) Simple transaction processing, which prevents inconsistencies when a device is controlled from multiple access points (e.g., local and remote) and during the processing of control commands.

b) State indications, which allows local and remote indication of ongoing controls.

6.6.2 Operation object

The Operation object is the abstract base class for classes that represent remote-controllable items. Each Operation object allows the system to modify some specific item (i.e., a virtual attribute) in a specific way defined by the Operation object. Operation objects are not directly accessible by services defined in the service model in Clause 8. All controls shall be routed through the SCO object (i.e., the parent) to allow a simple form of transaction processing.

The set of Operation objects instantiated by a particular medical device defines the complete remote control interface of the device. This way a host system is able to discover the remote control capabilities of a device in the configuration phase.

6.6.3 Select Item Operation object

The Select Item Operation object allows the selection of one item out of a given list.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

Example: The invasive pressure VMD may allow modification of its label. It uses a Select Item Operation object for this function. The list of legal values supplied by the operation may be, for example, {ABP, PAP, CVP, LAP}. By invoking the operation, a user is able to select one value out of this list.

6.6.4 Set Value Operation object

The Set Value Operation object allows the adjustment of a value within a given range with a given resolution.

Example: A measurement VMD may allow adjustment of a signal gain factor. It uses the Set Value Operation object for this function. The operation provides the supported value range and step width within this range.

6.6.5 Set String Operation object

The Set String Operation object allows the system to set the contents of an opaque string variable of a given maximum length and format.

Example: An infusion device may allow a remote system to set the name of the infused drug in free-text form to show it on a local display. It defines an instance of the Set String Operation object for this function. The operation specifies the maximum string length and the character format so that the device is able to show the drug name on a small display.

6.6.6 Toggle Flag Operation object

The Toggle Flag Operation object allows operation of a toggle switch (with two states, e.g., on/off).

Example: An ECG VMD may support a line frequency filter. It uses the Toggle Flag Operation object for switching the filter on or off.

6.6.7 Activate Operation object

The Activate Operation object allows a defined activity to be started (e.g., a zero pressure).

Example: The zero procedure of an invasive pressure VMD may be started with an Activate Operation object.

6.6.8 Limit Alert Operation object

The Limit Alert Operation object allows adjustment of the limits of a limit alarm detector and the switching of the limit alarm to on or off.

6.6.9 Set Range Operation object

The Set Range Operation object allows the selection of a value range by the simultaneous adjustment of a low and high value within defined boundaries.

Example: A measurement VMD may provide an analog signal input for which the signal input range can be adjusted with a Set Range Operation object.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

6.7 Model for the Extended Services Package

The Extended Services Package contains objects that provide extended medical object management services that allow efficient access to medical information in communicating systems. Such access is achieved by a set of objects that package attribute data from multiple objects in a single event message.

The objects providing extended services are conceptually derived from ISO/OSI system management services defined in the ISO/IEC 10164 family of standards (specifically Part 5 and Part 13). The definitions have been adapted to and optimized for specific needs in the area of vital signs communication between medical devices.

Figure 6.8 shows the object model for the Extended Services Package.

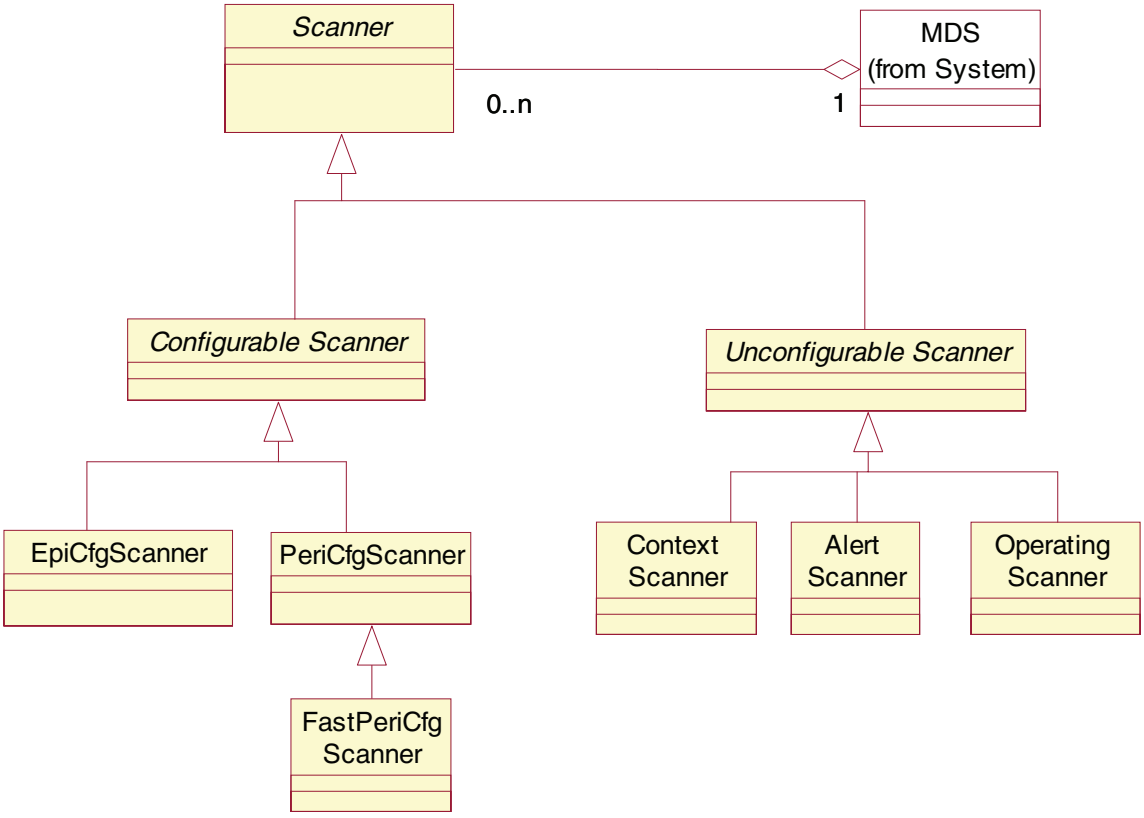

Figure 6.8—Extended Services Package model

The Extended Services Package model contains the objects described in 6.7.1 through 6.7.9.

6.7.1 Scanner object

A Scanner object is a base class that is an observer and "summarizer" of object attribute values. It observes attributes of managed medical objects and generates summaries in the form of notification event reports. These event reports contain data from multiple objects, which provide a better communication performance compared to separate polling commands (e.g., GET service) or multiple individual event reports from all object instances.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

Objects derived from the Scanner object may be instantiated either by the agent system itself or by the manager system (e.g., dynamic scanner creation by using the CREATE service).

As a base class, the Scanner object cannot be instantiated.

6.7.2 CfgScanner (i.e., configurable scanner) object

The CfgScanner object is a base class that has a special attribute (i.e., the ScanList attribute) that allows the system to configure which object attributes are scanned. The ScanList attribute may be modified either by the agent system (i.e., auto-configuration or pre-configuration) or by the manager system (i.e., full dynamic configuration by using the SET service).

A CfgScanner object may support different granularity for scanning:

a) Attribute group (i.e., a defined set of attributes): The ScanList attribute contains the identifiers (IDs) of attribute groups, and all attributes in the group are scanned.

b) Individual attribute: The ScanList attribute contains the IDs of all attributes that are scanned.

In order to deal efficiently with optional object attributes, the attribute group scan granularity is recommended for CfgScanner objects.

As a base class, the CfgScanner object cannot be instantiated.

6.7.3 EpiCfgScanner (i.e., episodic configurable scanner) object

The EpiCfgScanner object is responsible for observing attributes of managed medical objects and for reporting attribute changes in the form of unbuffered event reports.

The unbuffered event report is triggered only by object attribute value changes. If the EpiCfgScanner object uses attribute group scan granularity, the event report contains all attributes of the scanned object that belong to this attribute group if one or more of these attributes changed their value.

> **Example:** A medical device provides heart beat detect events in the form of an Enumeration object. A display application creates an instance of the EpiCfgScanner object and adds the observed value of the Enumeration object to the ScanList attribute. The scanner instance afterwards sends a notification when the Enumeration object reports a heart beat.

6.7.4 PeriCfgScanner (i.e., periodic configurable scanner) object

The PeriCfgScanner object is responsible for observing attributes of managed medical objects and for periodically reporting attribute values in the form of buffered event reports. A buffered event report contains the attribute values of all available attributes that are specified in the scan list, independent of attribute value changes.

If the scanner operates in a special superpositive mode, the buffered event report contains all value changes of attributes that occurred in the reporting period; otherwise, the report contains only the most recent attribute values.

> **Example:** A data logger creates an instance of the PeriCfgScanner object and configures the scanner so that it sends an update of the observed value attributes of all Numeric objects in the MDIB every 15 s.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

6.7.5 FastPeriCfgScanner (i.e., fast periodic configurable scanner) object

The FastPeriCfgScanner object is a specialized object class for scanning the observed value attribute of the Real Time Sample Array object. This special scanner object is further optimized for low-latency reporting and efficient communication bandwidth utilization, which is required to access real-time waveform data.

> **Example:** A real-time display application (e.g., manager system) wants to display ECG waveforms. It creates a FastPeriCfgScanner object on the agent system (e.g., server device) and requests periodic updates of all ECG leads.

6.7.6 UcfgScanner (i.e., unconfigurable scanner) object

The UcfgScanner object is a base class that scans a predefined set of managed medical objects that cannot be modified. In other words, an UcfgScanner object typically is a reporting object that is specialized for one specific purpose.

As a base class, the UcfgScanner object cannot be instantiated.

6.7.7 Context Scanner object

The Context Scanner object is responsible for observing device configuration changes. After instantiation, the Context Scanner object is responsible for announcing the object instances in the device's MDIB. The scanner provides the object instance containment hierarchy and static object attribute values.

In case of dynamic configuration changes, the Context Scanner object sends notifications about new object instances or deleted object instances.

> **Example:** A data logger creates an instance of the Context Scanner object in an agent MDIB to receive notifications about MDS configuration changes when new measurement modules are plugged in (i.e., new VMD instance) or when such a module is unplugged (i.e., VMD instance deleted).

6.7.8 Alert Scanner object

The Alert Scanner object is responsible for observing the alert-related attribute groups of objects in the Alert Package. As alarming in general is security-sensitive, the scanner is not configurable (i.e., all or no Alert objects are scanned).

The Alert Scanner object sends event reports periodically so that timeout conditions can be checked.

6.7.9 Operating Scanner object

The Operating Scanner object is responsible for providing all information about the operating and control system of a medical device.

In other words, the scanner maintains the configuration of Operation objects contained in SCOs (by sending CREATE notifications for Operation objects), it scans transaction-handling-related SCO attributes, and it scans Operation object attributes. Because SCOs and Operation objects may have dependencies, the scanner is not configurable.

6.8 Model for the Communication Package

The Communication Package deals with objects that enable and support basic communication.

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

Figure 6.9 shows the object model for the Communication Package.

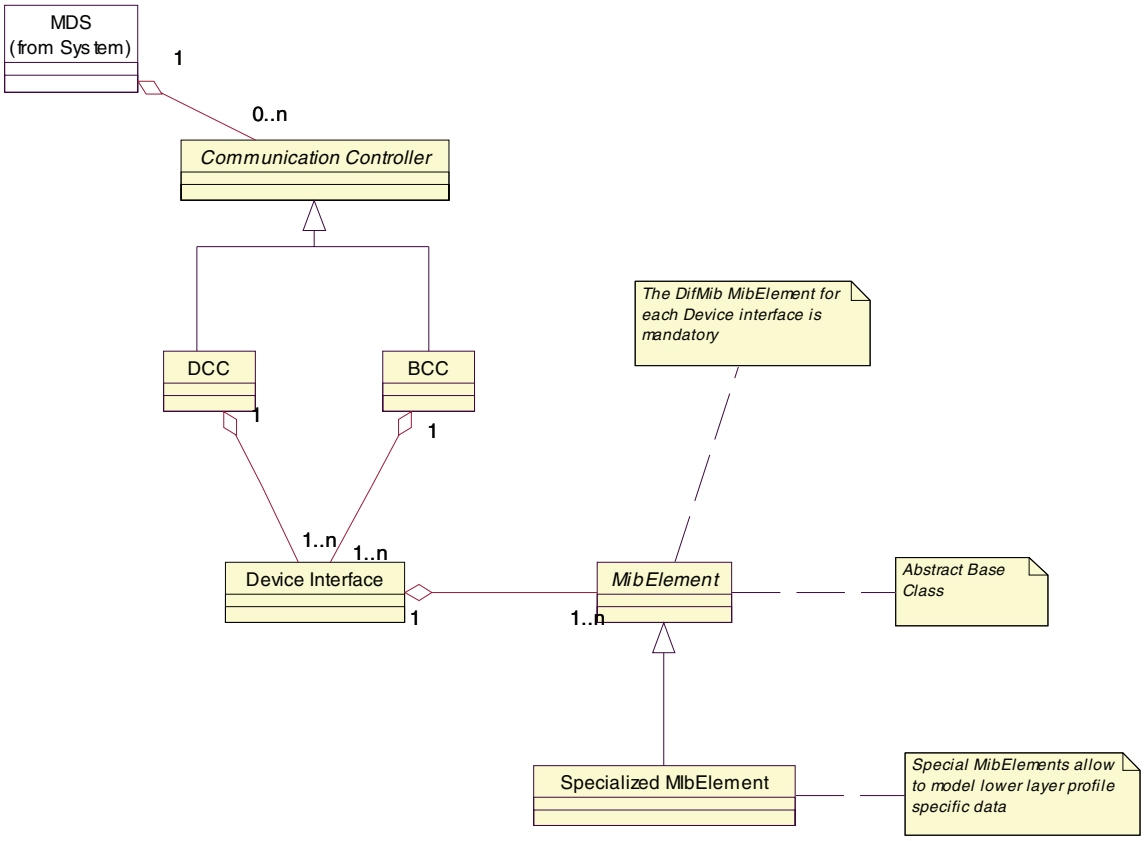

Figure 6.9—Communication Package model

The Communication Package model contains the objects described in 6.8.1 through 6.8.6.

6.8.1 Communication Controller object

The Communication Controller object represents the upper layer and lower layer communication profile of a medical device.

The Communication Controller object is the access point for retrieving Device Interface attributes and management information base element (MibElement) attributes for obtaining management information related to data communications.

As a base class, the Communication Controller object cannot be instantiated. Two instantiable specializations, however, are defined: BCC and DCC. A medical device MDIB contains either no Communication Controller object or one BCC object or one DCC object (depending on its role).

6.8.2 DCC (i.e., device communication controller) object

The DCC object is a Communication Controller object used by medical devices operating as agent systems (i.e., association responders).

The DCC object shall contain one or more Device Interface objects.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

6.8.3 BCC (i.e., bedside communication controller) object

The BCC object is a Communication Controller object used by medical devices operating as manager systems (i.e., association requestors).

The BCC object shall contain one or more Device Interface objects.

6.8.4 Device Interface object

The Device Interface object represents a particular interface, i.e., port. The port is either a logical or a physical end point of an association for which (e.g., statistical) data captured in the MibElement objects can be independently collected.

Both an agent system and a manager system can have multiple logical or physical ports, depending on the selected implementation of the lower layer communication system.

The Device Interface object is not accessible by CMDISE services. This object contains at least one MibElement object (i.e., the Device Interface MibElement object, which represents device interface properties), which can be accessed by a special method defined by the Communication Controller object.

6.8.5 MibElement object

The MibElement object contains statistics and performance data for one Device Interface object. The MibElement object is a base class for specialized MibElement objects only. It cannot be instantiated.

Various MibElement object types are defined to group management information in defined packages, which can be generic or dependent on specific transport profiles.

The MibElement object is not directly accessible. Its attributes can be accessed only through a Communication Controller object. The MibElement object is not part of the device's MDIB.

6.8.6 Specialized MibElement object

Management information for a communication link is dependent on the lower layers of the communication stack (i.e., lower layers profile).

This standard, however, defines only two generic MibElement objects:

— The mandatory Device Interface MibElement object, which describes properties of the device interface

— An optional general communication statistics MibElement object, which models typical communication statistics that are generally applicable

Specialized MibElement objects are defined in IEEE P1073.2.1.2.

6.9 Model for the Archival Package

The Archival Package deals with storage and representation of biosignals, status, and context information in an on-line or an off-line archive.

Figure 6.10 shows the object model of the Archival Package.

The Archival Package model contains the objects described in 6.9.1 through 6.9.7.

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

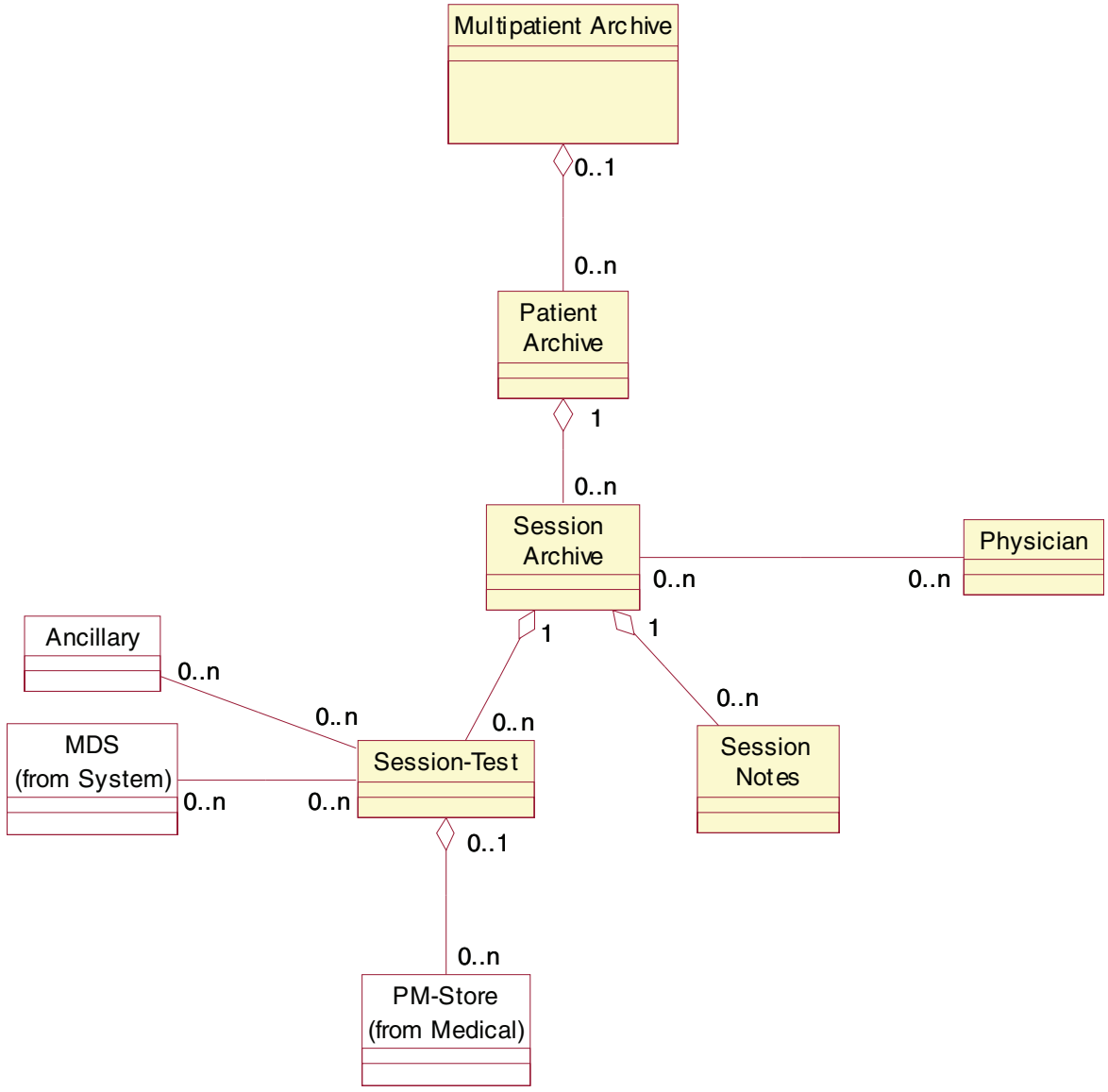

Figure 6.10—Archival Package model

6.9.1 Multipatient Archive object

The Multipatient Archive object groups together multiple Patient Archive ojbects referring to different patients.

Example: A drug study may be documented in the form of a Multipatient Archive object containing multiple Patient Archive objects that show how the drug affected the monitored vital signs.

6.9.2 Patient Archive object

The Patient Archive object groups patient-related information (e.g., vital signs data, treatment data, and patient demographics) together in a single archive object. This object relates to static (i.e., invariant) data in a Patient Demographics object only.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

Example: A hospital may store data about multiple visits of a single patient in a Patient Archive object that contains a number of Session Archive objects, each documenting vital signs information recorded during a specific visit in a hospital department.

6.9.3 Session Archive object

The Session Archive object represents a patient visit or a continuous stay in the a hospital or hospital department. Diagnostic treatments performed during this time period are represented by Session Test objects contained in the Session Archive object. The Session Archive object refers to dynamic (i.e., variant) data in a Patient Demographics object.

6.9.4 Physician object

The Physician object represents the physician responsible for the set of diagnostic and therapeutic activities during the time period represented by the Session Archive object.

6.9.5 Session Test object

The Session Test object contains vital signs information of a single patient that is recorded during a single examination or diagnostic treatment. This object contains vital signs metrics in form of PM-Store objects. It also may contain information about equipment that was used for recording (in the form of relations to MDS and Ancillary objects).

Example: Vital signs information recorded during a ECG stress test examination is organized in a Session Test object.

6.9.6 Session Notes object

The Session Notes object is a container for diagnostic data, patient care details, and treatment-related information in the form of textual data.

6.9.7 Ancillary object

The Ancillary object is not further defined in this standard. This object is present in the model to indicate that information from sources other than devices within the scope of this standard are permitted to be included in (or referenced by) the Session Test object.

Example: Image data that complies with the MEDICOM (CEN ENV 12052) or DICOM (NEMA PS 3) standard are permitted to be included in the Session Test object as ancillary data.

6.10 Model for the Patient Package

The Patient Package deals with all patient-related information that is relevant in the scope of this standard, but is not vital signs information modeled in the Medical Package.

Figure 6.11 shows the object model for the Patient Package:

The Patient Package model contains one object (see 6.10.1).

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

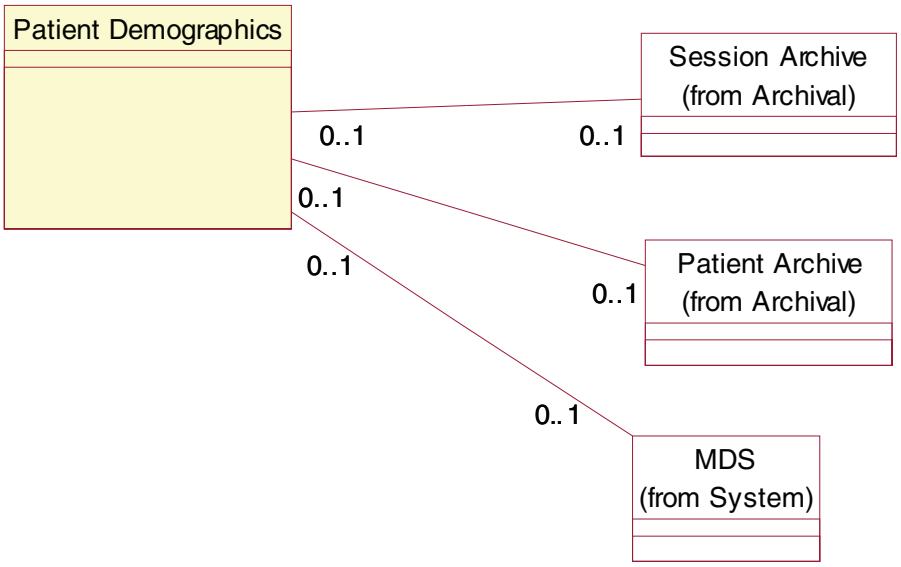

Figure 6.11—Patient Package model

6.10.1 Patient Demographics object

The Patient Demographics object stores patient census data.

This standard provides minimal patient information as typically required by medical devices. A complete patient record is outside the scope of this standard.

6.11 DIM—dynamic model

6.11.1 General

Subclause 6.11 defines global dynamic system behavior.

Note that dynamic object behavior resulting from invocation of object management services is part of the object definitions in Clause 7.

6.11.2 MDS communication finite state machine (FSM)

Figure 6.12 shows the MDS FSM for a communicating medical device that complies with the definitions in this standard. The FSM is used to synchronize the operational behavior of manager (i.e., client) systems and agent (i.e., server) systems.

After power-up, the device performs all necessary local initializations (i.e, boot phase) and ends up in the disconnected state, where it waits for connection events.

When a connection event is detected, the device tries to establish a logical connection (i.e, an association) with the other device. A manager (i.e., client) system is the association requester, and an agent (i.e, server) system is the association responder. Basic compatibility checks are performed in the associating state.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

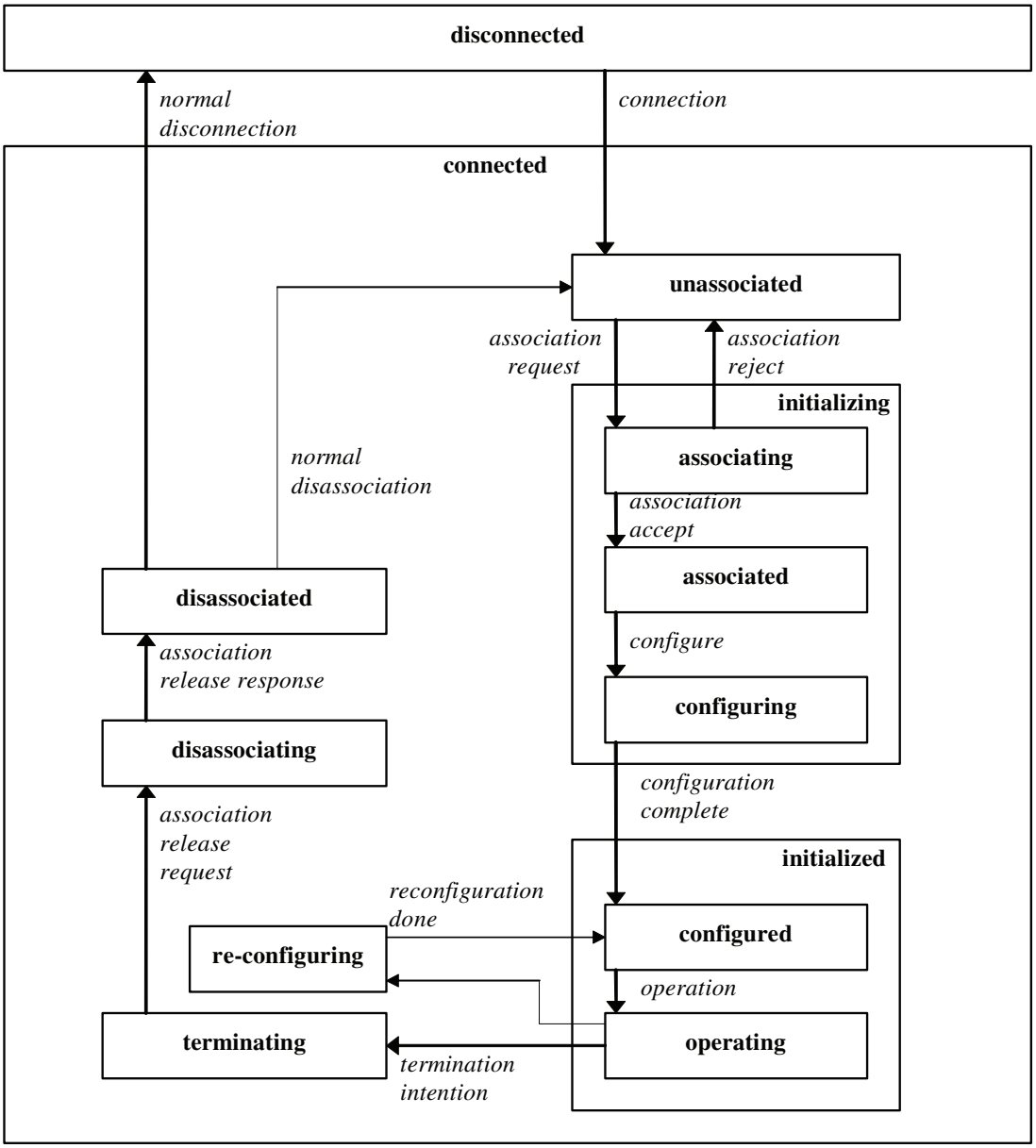

Figure 6.12—MDS FSM

After successful association, configuration data (i.e., the MDIB structure) is exchanged by the use of services and extended services (in particular, the Context Scanner object) as defined in this standard. Additional information (e.g., MDS attributes) is supplied that allows further compatibility and state checks.

After configuration, medical data are exchanged by using services and extended services as defined in this standard. Dynamic reconfiguration is allowed in the operating state. If the device or the type of reconfiguration does not allow dynamic handling in the operating state, a special “reconfiguring” state is provided.

If an event indicates an intention to disconnect, the disassociating state is entered.

The diagram does not show error events. Fatal error events take the state machine out of the operating state.

NOTE—This state machine describes the behavior of the MDS communication system only. Usually the device must perform its medical function independent of the communication system.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

The FSM is considered a part of the MDS object. The MDS Status attribute reflects the state of the machine. The MDS may announce state changes in the form of attribute change event reports.

Specific application profiles shall use this state machine as a general guideline, but they may define specific deviations to fulfill specific profile-dependent requirements or assumptions.

6.11.3 Communicating systems—startup object interaction diagram

Figure 6.13 presents the object interaction diagram that visualizes the startup phase after connecting two devices.

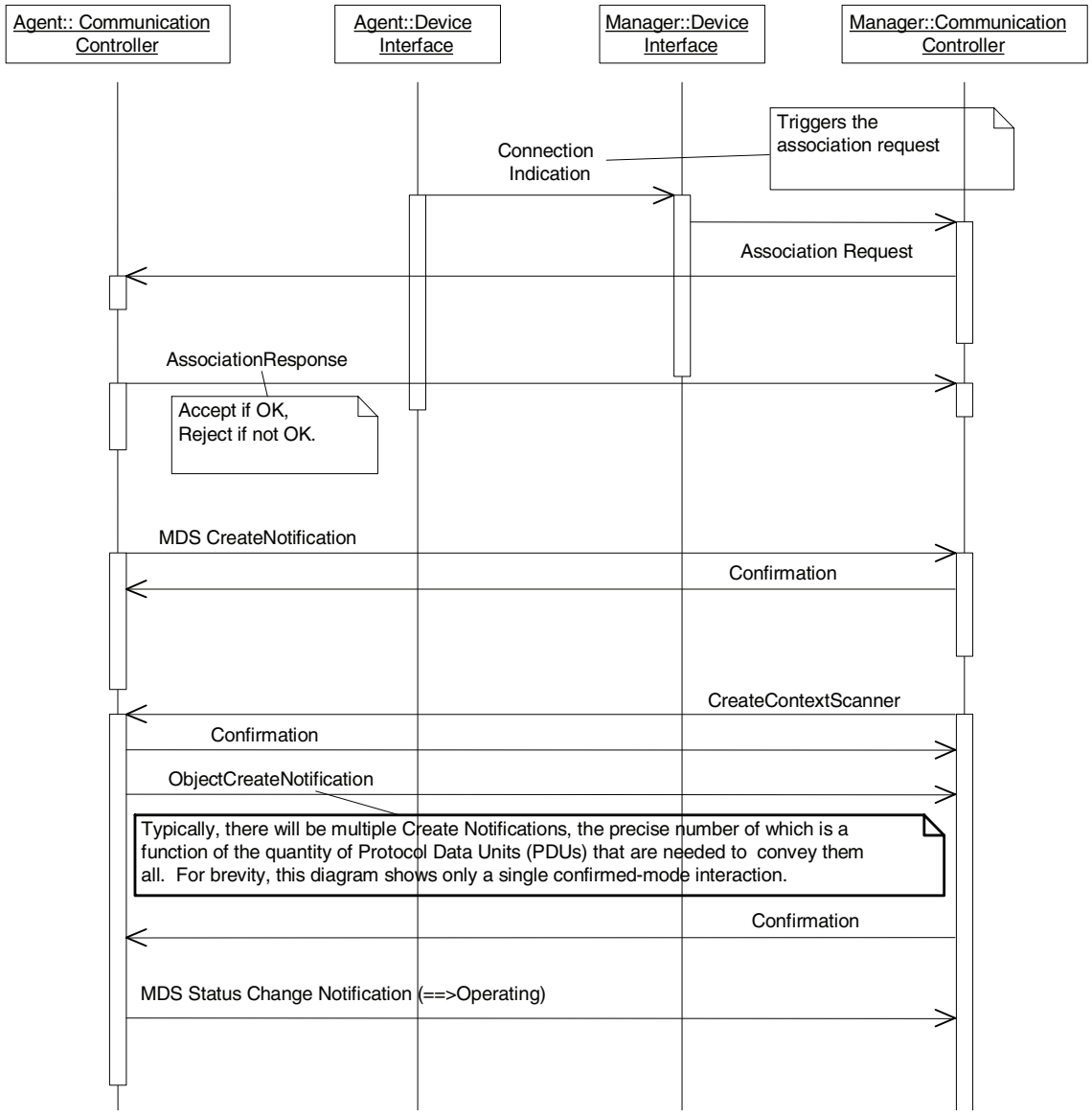

Figure 6.13—Startup after connection

It is assumed here that, conceptually, messages are exchanged between the Communication Controller objects (using the device interface).

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

Some form of connection indication is necessary to make the manager system aware of a new agent on the network. This mechanism is dependent on the specific lower layer implementation; therefore, it is not further defined in this standard.

Specific application profiles shall use this interaction diagram as a general guideline, but they may define specific deviations to fulfill specific profile-dependent requirements or assumptions.

6.11.4 Communication Package—MibElement data access

Figure 6.14 presents the object interaction diagram that shows how a manager system accesses the MibElement data using the objects defined in the Communication Package (see 6.8).

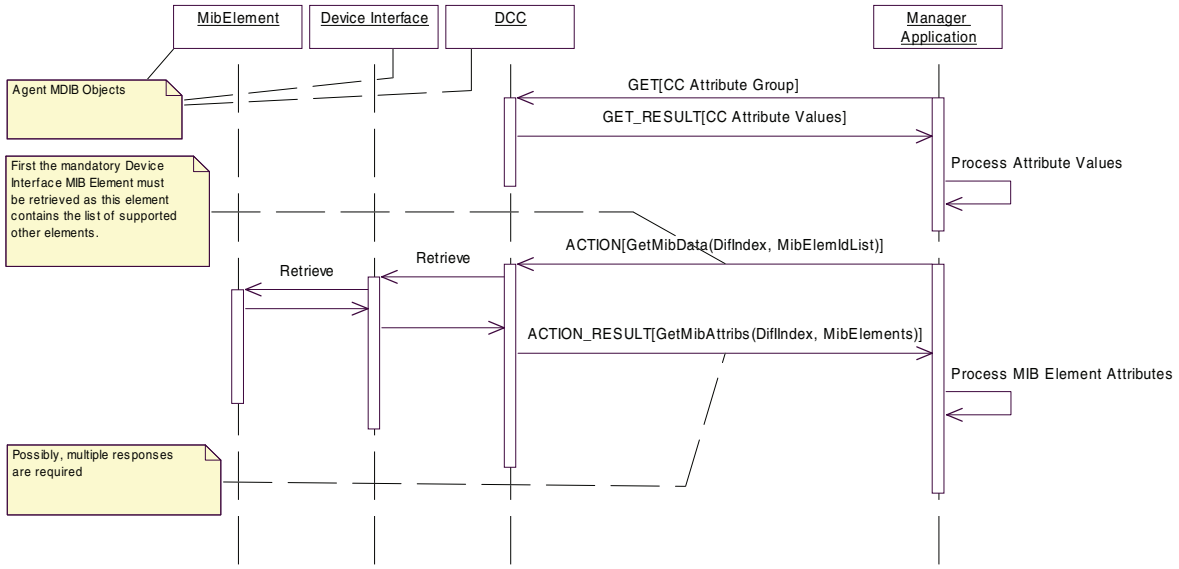

Figure 6.14—MibElement data access

The diagram assumes the following:

- An association is established between the agent and the manager.
- The configuration phase is finished, and the manager has an image of the agent's MDIB.
- The DCC object is part of the agent's MDIB.

The manager first uses the GET service to retrieve all DCC object attributes and their values. The attributes specify how many Device Interface objects exist.

The manager uses the ACTION service with the Communication Controller object-defined method to retrieve the attributes of the mandatory Device Interface MibElement object. The MibElement attribute variables specify if any additional MibElement objects are available for the interface.

If so, the manager can use the same ACTION command to retrieve the additional management information represented in the MibElement objects.

6.11.5 Dynamic object relations

This subclause deals with relations between managed medical objects (i.e., objects that are defined as managed objects in this standard).

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

Generally, the relationships between object instances that are defined in the package models are dynamic.

> **Example:** A modular patient monitor is modeled as an MDS. Measurement modules are modeled as VMDs. If a new module is connected to the monitor, there is also a new relationship between the MDS and the new VMD instance.

Communicating agent systems (i.e., agents) use services defined in this standard to announce configuration change events to other connected systems. These manager systems (i.e., managers) modify their view of the agent MDIB.

Not only does a vital signs information archive have to update its configuration, but it also has to permanently store these connection and disconnection events.

> **Example:** An instance of the Session Archive object represents the stay of a patient in the intensive care unit (ICU). During that period, new devices are connected to the patient to increase the number of recorded vital signs. They are removed again as soon as the patient's condition stabilizes. The Session Archive object shall not delete recorded data when the recording device is disconnected.

Thus, in certain applications (e.g., archival applications), object instance relationships have associated information that must be captured.

When required, the relationships themselves can be considered to be special managed objects as shown in Figure 6.15.

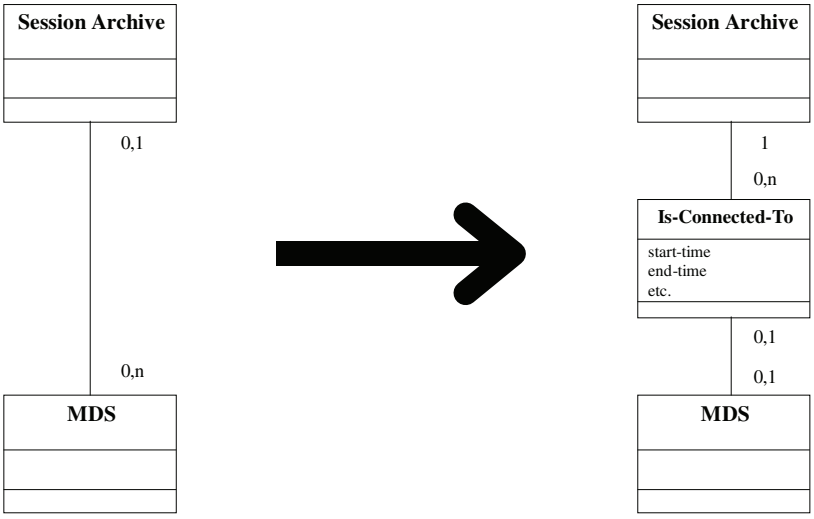

Figure 6.15—Example of a relationship represented by an object

The example in Figure 6.15 shows a relation between a Session Archive object and an MDS object. The relation is represented as an object. This object has attributes that provide information, for example, about time of connection and disconnection.

Modeling the relations as objects has the advantage that information can be defined in the form of attributes. It is not necessary to assign the attributes to one or both objects that are related.

How dynamic object relations are handled by systems that comply with the definitions in this standard is defined in 6.11.5.1 and 6.11.5.2.

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

6.11.5.1 Dynamic object relations in communicating systems

Relations between object instances that are defined in the package models are considered configuration information. The Context Scanner object provides configuration information in the form of object create notifications and object delete notifications. No means for persistent storage of past (i.e., old) configuration information is defined for communicating systems.

Other relations between object instances (e.g., to reference a source signal in derived data) are specified in the form of attributes of the corresponding objects (e.g., the Vmo-Source-List attribute of the Metric object). Dynamic changes of these attributes are announced by attribute change notification events.

6.11.5.2 Dynamic object relations in archival systems

An archival system may need to provide persistent storage for configuration information. In this case, the corresponding relations are considered to be represented by objects, as shown in 6.11.5.

An archival system that uses a data format in compliance with the definitions in this standard has to provide means to store (i.e., archive) the attributes of dynamic relationship objects.

7. DIM object definitions

7.1 General

Clause 7 contains the object definitions for all objects in the DIM. The packages defined in the model are used to categorize objects. Attributes, behavior, and notifications are defined for each object class.

7.1.1 Notation

Each object is defined in a separate subclause (see 7.2 through 7.10). Further subclauses define attributes, behavior, and notifications for the objects.

The object is defined in a subclause as follows:

Object:	Defines the name of the object.
Description:	"Gives a short, informative textual description of the object."
Derived From:	Defines potential base classes of the object.
Name Binding:	Defines the attribute that uniquely identifies an instance of the object in a given context. For manageable objects, this definition is the Handle attribute and the context is the device system (i.e., single MDS context). See also 7.1.2.5.
Registered As:	Defines a term that is defined in the nomenclature to allow unique identification [e.g., object identifier (OID), code] of the object.

Each object attribute is defined in an attribute subclause. Tables define attribute names, unique attribute IDs, attribute data types, and certain qualifiers. The qualifiers have the following meaning:

M attribute is mandatory
O attribute is optional
C attribute is conditional; availability of attribute depends on a predefined condition

Unless otherwise noted, the attribute definition tables do not show inherited attributes again. In other words, the attribute lists of all base classes have to be checked for a complete list of object attributes.

Attributes are assigned to, or grouped together in. attribute groups so attributes can be classified according to their use (e.g., static context information, dynamic context information, value observations). The grouping also makes it possible to effectively deal with optional attributes: A GET service facilitates the retrieval of

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

all members of the group so an application is able to determine which attributes are actually present in a particular object instance.

Attribute groups may be extensible. In other words, a derived object class is able to add additional members to an inherited attribute group.

Attribute groups are also defined in tables that specify group identification and the list of group members. Inherited attribute groups are not shown in the attribute group tables again unless these groups are extensible.

Special methods or functions that are provided by an object class are defined in a behavior subclause. These methods can be invoked by the CMDISE ACTION service.

Events generated by an object class (other than a generic attribute change notification) are defined in a notifications subclause. An object reports these events by using the CMDISE EVENT REPORT service.

7.1.2 Common data types

This subclause defines a set of ASN.1 data types that are used in the object definitions.

7.1.2.1 Integer and bit string data types

For representing integer numbers, the object definitions use fixed-size data types only. The bit string data type represents a bit field where each single bit has a defined meaning (i.e., flag fields). The following integer data types and bit string data types are used:

```
--
-- 8-bit unsigned integer
--
INT-U8 ::= INTEGER (0..255)
--
-- 8-bit signed integer
--
INT-I8 ::= INTEGER (-128..127)
--
-- 16-bit unsigned integer
--
INT-U16 ::= INTEGER (0..65535)
--
-- 16-bit signed integer
--
INT-I16 ::= INTEGER (-32768..32767)
--
-- 32-bit unsigned integer
--
INT-U32 ::= INTEGER (0..4294967295)
--
-- 32-bit signed integer
--
INT-I32 ::= INTEGER (-2147483648..2147483647)

--
-- 16-bit bit string
--
BITS-16 ::= BIT STRING (SIZE(16))
--
-- 32-bit bit string
--
BITS-32 ::= BIT STRING (SIZE(32))
```


Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

NOTES

1—When interpreting integer numbers, the representation (e.g., little endian versus big endian) has to be considered. Communicating systems negotiate this representation at association (i.e., transfer syntax). Archival data formats have to provide a mechanism to uniquely identify integer representation (e.g., a field in a specification header).

2—In the object definitions, integer and bit string data types with named constants or named bits also use the above notation for simplicity. The above notation is illegal ASN.1 syntax, but it can be easily transformed to the correct syntax.

7.1.2.2 Identification data type

All elements (e.g., classes, objects, measurement types) that need unique identification are assigned an OID. The set of valid OIDs for this standard is defined in ISO/IEEE 11073-10101. The nomenclature is split into a set of partitions, and each partition has its own range of 16-bit codes. In other words, the 16-bit code is context-sensitive.

The 16-bit identification data type is defined as follows:

```
--
-- OID type as defined in nomenclature
-- (do not confuse with ASN.1 OID)
--
OID-Type ::= INT-U16                    -- 16-bit integer type
```

For IDs that are not part of the standard nomenclature (i.e., private or manufacturer-specific codes), a special type is defined as follows:

```
--
-- Private OID
--
PrivateOid ::= INT-U16
```

7.1.2.3 Handle data type

The handle data type is used for efficient, locally unique identification of all managed medical object instances. (*Locally unique* means unique within one MDS context.) This data type is defined as follows:

```
--
-- handle
--
HANDLE ::= INT-U16
```

7.1.2.4 Instance number data type

The instance number data type is used to distinguish class or object instances of the same type or object instances that are not directly manageable (i.e., used, e.g., as the Name Binding attribute for Operation objects). This data type is defined as follows:

```
--
-- Instance Number
--
InstNumber ::= INT-U16
```

7.1.2.5 Global object identification

Handle and instance number data types must be unique inside one specific naming context (e.g., handles are unique within at least one MDS context). This uniqueness allows the identification of an object instance within its naming context by one single, small identifier.

Documents
Document Sharing Hub

Full document details: DIN EN ISO 11073-10201:2006-11

To address larger scale systems, a context ID field at the MDS level within an MDIB is added to the handle data type so that multiple device systems can be distinguished. This global handle data type is defined as follows:

```
--
-- MDS Context ID
--
MdsContext ::= INT-U16

--
-- Global handle allows identification of an object in a larger scale system
--
GLB-HANDLE ::= SEQUENCE {
    context-id                  MdsContext,
    handle                      HANDLE
}

--
-- Managed OID as a type for complete global object identification
--
ManagedObjectId ::= SEQUENCE {
    m-obj-class                 OID-Type,
    m-obj-inst                  GLB-HANDLE
}
```

Example: A medical device may interface with further medical devices (i.e., sub-devices). In the MDIB, this device may model these sub-devices as individual MDS objects with their own naming context. In this way, name space collisions (e.g., duplicate handle values, duplicate nomenclature codes) can be avoided without reassigning handle values. A manager system needs to interpret the MDS context IDs together with handle values to uniquely identify object instances within this composite MDIB. The context IDs are assigned when the MDIB is created by Context Scanner object create notifications.

Assumptions and possible restrictions about different naming contexts within an MDIB are profile dependent. They are, therefore, defined in the ISO/IEEE P11073-202xx series.

7.1.2.6 Type ID data type

The type ID data type is used in the VMOs and VMS objects to provide specific static information about the type of an object instance (e.g., blood pressure could be the type of a Numeric object). Codes defined in the nomenclature are used. The nomenclature contains a number of partitions, and code values are unique only within one partition. As the type ID data type should be context-free, the partition of the nomenclature code is also provided. This data type is defined as follows:

```
--
-- Type ID
--
TYPE ::= SEQUENCE {
    partition                   NomPartition,
    code                        OID-Type
}

--
-- The following nomenclature partitions exist
--
NomPartition ::= INT-U16 {
    nom-part-unspec(0),
    nom-part-obj(1),                    -- object-oriented partition
    nom-part-metric(2),                 -- metric [supervisory control and data acquisition
                                        -- (SCADA)] partition
    nom-part-alert(3),                  -- alerts/events partition
```


Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11

```
    nom-part-dim(4),                -- dimensions partition
    nom-part-vattr(5),              -- virtual attribute partition for Operation objects
    nom-part-pgrp(6),               -- parameter group ID partition
    nom-part-sites(7),              -- measurement and body site locations
    nom-part-infrastruct(8),        -- infrastructure elements partition
    nom-part-fef(9)                 -- file exchange format partition
    nom-part-ecg-extn(10),          -- ECG extensions partition
    nom-part-ext-nom(256),          -- IDs of other nomenclatures and dictionaries
    nom-part-priv(1024)             -- private partition
}
```

7.1.2.7 Attribute value assertion data type

A number of services defined in the service model in Clause 8 provide access to object attributes (e.g., GET, SET). Typically, the attribute has to be identified by means of an attribute ID. The attribute data type itself is dependent on this ID. The attribute value assertion data type represents this ID-value pair and is defined as follows:

```
--
AVA-Type ::= SEQUENCE {
    attribute-id            OID-Type,
    attribute-value         ANY DEFINED BY attribute-id
}
```

7.1.2.8 Attribute list data type

Frequently, a list of attribute ID–attribute value pairs is needed. The attribute list data type is a special data type that is provided for this situation and is defined as follows:

```
--
AttributeList ::= SEQUENCE OF AVA-Type
```

7.1.2.9 Attribute ID list data type

Frequently, a list of attribute IDs is used. The attribute ID list data type is a special type that is provided for convenience and is defined as follows:

```
--
AttributeIdList ::= SEQUENCE OF OID-Type
```

7.1.2.10 Floating point type data type

For performance and efficiency, the object definitions use the floating point type data type, which is a special data type for representing floating point numbers. It is assumed that this data type is 32 bits. This data type is defined as follows:

```
--
-- 32-bit float type; the integer type is a placeholder only
--
FLOAT-Type ::= INT-U32
```

The concrete floating point number format is either explicitly negotiated at association or implicitly defined by the association application context.

7.1.2.11 Relative time data type

The relative time data type is a high-resolution time definition relative to some event (e.g., a synchronization event at startup). This data type is used to position events relative to each other. It is defined as follows:

Documents
Document Sharing Hub
Full document details: DIN EN ISO 11073-10201:2006-11