International Standard

ISO/IEEE 11073-10471

Second edition 2024-09

Health informatics — Device interoperability —

Part 10471: Personal health device communication — Device specialization — Independent living activity hub

Informatique de santé — Interopérabilité des dispositifs —

Partie 10471: Communication entre dispositifs de santé personnels — Spécialisation des dispositifs — Concentrateur d'activités pour une vie autonome

Reference number
ISO/IEEE 11073-10471:2024(en)

Institute of Electrical and Electronics Engineers, Inc
3 Park Avenue, New York
NY 10016-5997, USA

Email: stds.ipr@ieee.org
Website: www.ieee.org

Published in Switzerland

Foreword

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

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

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

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

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

ISO/IEEE 11073-10471 was prepared by the IEEE 11073 Standards Committee of the IEEE Engineering in Medicine and Biology Society (as IEEE Std 11073-10471) and drafted in accordance with its editorial rules. It was adopted, under the “fast-track procedure” defined in the Partner Standards Development Organization cooperation agreement between ISO and IEEE, by Technical Committee ISO/TC 215, *Health informatics*.

This second edition cancels and replaces the first edition (ISO/IEEE 11073-10471:2010), which has been technically revised.

The main changes are as follows:

- added current value sensors (temperature, humidity, utility usage);
- added humidity alert sensor;
- added support for location sensors;
- added support for Base-Offset-Time;

- added new location identifiers;
- incorporated location identifiers from ZigBee Home Automation and Health Care profiles;
- added new appliance identifiers;
- changed from bit string to simple OIDs to report events;
- updated the example to simple OID;
- changed supplementary types to have upper 11 bits for location and lower 5 bits for number to support more codes within partition;
- updated the version of this device specialization;
- updated the association details based on new version;
- updated the wording in 6.3 regarding the Not Allowed and Observational;
- added systematic names;
- tamper detected flag made a general health status flag for all sensors;
- enumeration codes provided to other systems (ZigBee);
- location RefId discriminator changed from parentheses to underscore ((3)->_3);
- encounter object added.

A list of all parts in the ISO 11073 series can be found on the ISO website.

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

Abstract: Within the context of the ISO/IEEE 11073 family of standards for device communication, a normative definition of communication between personal telehealth independent living activity hub devices and compute engines (e.g., cell phones, personal computers, personal health appliances, set top boxes) in a manner that enables plug-and-play interoperability is established in this standard. Appropriate portions of existing standards, including ISO/IEEE 11073 terminology, information models, application profile standards, and transport standards are leveraged. The use of specific term codes, formats, and behaviors in telehealth environments restricting optionality in base frameworks in favor of interoperability are specified. A common core of communication functionality for personal telehealth independent living activity hubs is defined in this standard.

Keywords: IEEE 11073-10471™, independent living activity hub, medical device communication, personal health devices

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

PDF: ISBN 978-1-5044-9825-8 STD26246
Print: ISBN 978-1-5044-9826-5 STDPD26246

Important Notices and Disclaimers Concerning IEEE Standards Documents

IEEE Standards documents are made available for use subject to important notices and legal disclaimers. These notices and disclaimers, or a reference to this page (https://standards.ieee.org/ipr/disclaimers.html), appear in all standards and may be found under the heading "Important Notices and Disclaimers Concerning IEEE Standards Documents."

Notice and Disclaimer of Liability Concerning the Use of IEEE Standards Documents

IEEE Standards documents are developed within IEEE Societies and subcommittees of IEEE Standards Association (IEEE SA) Board of Governors. IEEE develops its standards through an accredited consensus development process, which brings together volunteers representing varied viewpoints and interests to achieve the final product. IEEE Standards are documents developed by volunteers with scientific, academic, and industry-based expertise in technical working groups. Volunteers are not necessarily members of IEEE or IEEE SA and participate without compensation from IEEE. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

IEEE makes no warranties or representations concerning its standards, and expressly disclaims all warranties, express or implied, concerning this standard, including but not limited to the warranties of merchantability, fitness for a particular purpose and non-infringement. In addition, IEEE does not warrant or represent that the use of the material contained in its standards is free from patent infringement. IEEE standards documents are supplied "AS IS" and "WITH ALL FAULTS."

Use of an IEEE standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity, nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

IN NO EVENT SHALL IEEE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: THE NEED TO PROCURE SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE PUBLICATION, USE OF, OR RELIANCE UPON ANY STANDARD, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE AND REGARDLESS OF WHETHER SUCH DAMAGE WAS FORESEEABLE.

Translations

The IEEE consensus development process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE is the approved IEEE standard.

Official statements

A statement, written or oral, that is not processed in accordance with the IEEE SA Standards Board Operations Manual shall not be considered or inferred to be the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that the presenter's views should be considered the personal views of that individual rather than the formal position of IEEE, IEEE SA, the Standards Committee, or the Working Group. Statements made by volunteers may not represent the formal position of their employer(s) or affiliation(s).

Comments on standards

Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE or IEEE SA. However, **IEEE does not provide interpretations, consulting information, or advice pertaining to IEEE Standards documents**.

Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its Societies and subcommittees of the IEEE SA Board of Governors are not able to provide an instant response to comments, or questions except in those cases where the matter has previously been addressed. For the same reason, IEEE does not respond to interpretation requests. Any person who would like to participate in evaluating comments or in revisions to an IEEE standard is welcome to join the relevant IEEE working group. You can indicate interest in a working group using the Interests tab in the Manage Profile & Interests area of the IEEE SA myProject system.[1] An IEEE Account is needed to access the application.

Comments on standards should be submitted using the Contact Us form.[2]

Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not constitute compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Data privacy

Users of IEEE Standards documents should evaluate the standards for considerations of data privacy and data ownership in the context of assessing and using the standards in compliance with applicable laws and regulations.

Copyrights

IEEE draft and approved standards are copyrighted by IEEE under US and international copyright laws. They are made available by IEEE and are adopted for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making these documents available for use and adoption by public authorities and private users, neither IEEE nor its licensors waive any rights in copyright to the documents.

Photocopies

Subject to payment of the appropriate licensing fees, IEEE will grant users a limited, non-exclusive license to photocopy portions of any individual standard for company or organizational internal use or individual, non-commercial use only. To arrange for payment of licensing fees, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400; https://www.copyright.com/. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Updating of IEEE Standards documents

Users of IEEE Standards documents should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect.

[1] Available at: https://development.standards.ieee.org/myproject-web/public/view.html#landing.

[2] Available at: https://standards.ieee.org/content/ieee-standards/en/about/contact/index.html.

Every IEEE standard is subjected to review at least every 10 years. When a document is more than 10 years old and has not undergone a revision process, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE standard.

In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit IEEE Xplore or contact IEEE.[3] For more information about the IEEE SA or IEEE's standards development process, visit the IEEE SA Website.

Errata

Errata, if any, for all IEEE standards can be accessed on the IEEE SA Website.[4] Search for standard number and year of approval to access the web page of the published standard. Errata links are located under the Additional Resources Details section. Errata are also available in IEEE Xplore. Users are encouraged to periodically check for errata.

Patents

IEEE Standards are developed in compliance with the IEEE SA Patent Policy.[5]

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE SA Website at https://standards.ieee.org/about/sasb/patcom/patents.html. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

IMPORTANT NOTICE

IEEE Standards do not guarantee or ensure safety, security, health, or environmental protection, or ensure against interference with or from other devices or networks. IEEE Standards development activities consider research and information presented to the standards development group in developing any safety recommendations. Other information about safety practices, changes in technology or technology implementation, or impact by peripheral systems also may be pertinent to safety considerations during implementation of the standard. Implementers and users of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.

[3] Available at: https://ieeexplore.ieee.org/browse/standards/collection/ieee.
[4] Available at: https://standards.ieee.org/standard/index.html.
[5] Available at: https://standards.ieee.org/about/sasb/patcom/materials.html.

Participants

At the time this standard was completed, the Personal Health Devices Working Group had the following membership:

Daidi Zhong, *Chair*
Malcolm Clarke, *Vice Chair*
Raymond Krasinski, *Secretary*

Karsten Aalders
Charles R. Abbruscato
Nabil Abujbara
Maher Abuzaid
James Agnew
Manfred Aigner
Jorge Alberola
David Aparisi
Lawrence Arne
Diego B. Arquillo
Serafin Arroyo
Muhammad Asim
Kit August
Doug Baird
David Baker
Anindya Bakshi
Ananth Balasubramanian
Sunlee Bang
M. Jonathan Barkley
Gilberto Barrón
David Bean
John Bell
Olivia Bellamou-Huet
Rudy Belliardi
Daniel Bernstein
George A. Bertos
Chris Biernacki
Ola Björsne
Thomas Blackadar
Thomas Bluethner
Douglas P. Bogia
Xavier Boniface
Shannon Boucousis
Lyle G. Bullock, Jr.
Bernard Burg
Chris Burns
Jeremy Byford-Rew
Satya Calloji
Carole C. Carey
Craig Carlson
Santiago Carot-Nemesio
Seungchul Chae
Yao Chen
Jing Cheng
Peggy Chien
David Chiu
Jinyong Choi
Chia-Chin Chong
Jinhan Chung
John A. Cogan
John T. Collins
Cory Condek
Todd H. Cooper
Sandra Costanzo
Douglas Coup

Nigel Cox
Hans Crommenacker
Tomio Crosley
Allen Curtis
Jesús Daniel Trigo
David Davenport
Russell Davis
Sushil K. Deka
Ciro de la Vega
Jim Dello Stritto
Kent Dicks
Hyoungho Do
Fangjie Dong
Jonathan Dougherty
Xiaolian Duan
Sourav Dutta
Jakob Ehrensvard
Fredrik Einberg
Javier Escayola Calvo
Mark Estes
Leonardo Estevez
Bosco T. Fernandes
Christoph Fischer
Morten Flintrup
Russell Foster
Eric Freudenthal
Matthias Frohner
Ken Fuchs
Jing Gao
Marcus Garbe
John Garguilo
Liang Ge
Rick Geimer
Igor Gejdos
Ferenc Gerbovics
Alan Godfrey
Nicolae Goga
Julian Goldman
Raul Gonzalez Gomez
Chris Gough
Channa Gowda
Charles M. Gropper
Amit Gupta
Jeff Guttmacher
Rasmus Haahr
Christian Habermann
Michael Hagerty
Jerry Hahn
Robert Hall
Shu Han
Nathaniel Hamming
Rickey L. Hampton
Sten Hanke
Aki Harma
Jordan Hartmann

Kai Hassing
Avi Hauser
Nathaniel Heintzman
Charles Henderson
Jun-Ho Her
Timothy L. Hirou
Allen Hobbs
Alex Holland
Arto Holopainen
Kris Holtzclaw
Robert Hoy
Anne Huang
Guiling Huang
Haofei Huang
Zhiyong Huang
David Hughes
Robert D. Hughes
Jiyoung Huh
Hugh Hunter
Philip O. Isaacson
Atsushi Ito
Michael Jaffe
Praduman Jain
Hu Jin
Danny Jochelson
Akiyoshi Kabe
Steve Kahle
Tomio Kamioka
James J. Kang
Kei Kariya
Andy Kaschl
Junzo Kashihara
Ralph Kent
Laurie M. Kermes
Sanjay R. Kharche
Ahmad Kheirandish
Junhyung Kim
Minho Kim
Min-Joon Kim
Taekon Kim
Tetsuya Kimura
Michael J. Kirwan
Alfred Kloos
Edward Koch
Jeongmee Koh
Jean-Marc Koller
John Koon
Patty Krantz
Alexander Kraus
Ramesh Krishna
Geoffrey Kruse
Falko Kuester
Rafael Lajara
Pierre Landau
Jaechul Lee

JongMuk Lee
Kyong Ho Lee
Rami Lee
Sungkee Lee
Woojae Lee
Jing Li
Qiong Li
Xiangchen Li
Xiaoyu Li
Patrick Lichter
Jisoon Lim
Wei-Jung Lo
Charles Lowe
Ling Luo
Don Ludolph
Christian Luszick
Bob MacWilliams
Srikkanth Madhurbootheswaran
Miriam L. Makhlouf
M. Sabarimalai Manikandan
Romain Marmot
Sandra Martinez
Miguel Martínez de EsproncedaCámara
Peter Mayhew
Jim McCain
LászlóMeleg
Alexander Mense
Behnaz Minaei
Jinsei Miyazaki
Madhu Mohan
Erik Moll
Darr Moore
Chris Morel
Carsten Mueglitz
Soundharya Nagasubramanian
Alex Neefus
Trong-Nghia Nguyen-Dobinsky
Michael E. Nidd
Jim Niswander
Hiroaki Niwamoto
Thomas Norgall
Yoshiteru Nozoe
Abraham Ofek

Brett Olive
BegonyaOtal
Marco Paleari
Bud Panjwani
Carl Pantiskas
Harry P. Pappas
Hanna Park
Jong-Tae Park
Myungeun Park

Phillip E. Pash
Tong Bi Pei
Soren Petersen
James Petisce
Peter Piction
Michael Pliskin
Varshney Prabodh
Jeff Price
Harald Prinzhorn
Lifei Qian
Harry Qiu
Tanzilur Rahman
Lin Ran
Phillip Raymond
Terrie Reed
Barry Reinhold
Brian Reinhold
John G. Rhoads
Jeffrey S. Robbins
Chris Roberts
Moskowitz Robert
Stefan Robert
Scott M. Robertson
Timothy Robertson
Sean Rocke
David Rosales
Bill Saltzstein
Giovanna Sannino
Jose A. Santos-Cadenas
Stefan Sauermann
John Sawyer
Alois Schloegl
Paul S. Schluter
Mark G. Schnell
Richard A. Schrenker
Antonio Scorpiniti
KwangSeok Seo
Riccardo Serafin
Sid Shaw
Frank Shen
Min Shih
Mazen Shihabi
Redmond Shouldice
Sternly K. Simon
Marjorie Skubic
Robert Smith
Ivan Soh
Motoki Sone
Emily Sopensky
Rajagopalan Srinivasan
Nicholas Steblay
Lars Steubesand
John (Ivo) Stivoric

Hermanni Suominen
Lee Surprenant
Ravi Swami
Ray Sweidan
Na Tang
Haruyuyki Tatsumi
Isabel Tejero
Chn Jonas Tirén
Janet Traub
Gary Tschautscher
Masato Tsuchid
Ken Tubman
Akib Uddin
Sunil Unadkat
Fabio Urbani
Philipp Urbauer
Laura Vanzago
Alpo Värri
Andrei Vasilateanu
Dalimar Velez
Martha Velezis
Rudi Voon
Isobel Walker
David Wang
Jerry P. Wang
Shiwei Wang
Yao Wang
Yi Wang
Steve Warren
Fujio Watanabe
Toru Watsuji
Kathleen Wible
Paul Williamson
Jia-Rong Wu
Will Wykeham
Ariton Xhafa
Ricky Yang
Melanie S. Yeung
Qiang Yin
Done-Sik Yoo
Zhi Yu
Jianchao Zeng
Jason Zhang
Zerui Zhang
Shibai Zhao
Yu Zhao
Liang Zheng
Yuanhong Zhong
Qing Zhou
Miha Zoubek
Szymon Zyskoter

The following members of the individual Standards Association balloting group voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Bjoern Andersen
William Byrd
Randy Carroll
Keith Chow
Malcolm Clarke

Immanuel Freedman
David Fuschi
Randall Groves
Werner Hoelzl
Noriyuki Ikeuchi

Atsushi Ito
Raj Jain
Ghalib Muhammad Waqas Janjua
Piotr Karocki

Raymond Krasinski
H. Moll
Nick S.A. Nikjoo
Beth Pumo
Stefan Schlichting
Walter Struppler
Oren Yuen
Jia Zheng

When the IEEE SA Standards Board approved this standard on 30 March 2023, it had the following membership:

David J. Law, *Chair*
Ted Burse, *Vice Chair*
Gary Hoffman, *Past Chair*
Konstantinos Karachalios, *Secretary*

Sara R. Biyabani
Doug Edwards
Ramy Ahmed Fathy
Guido R. Hiertz
Yousef Kimiagar
Joseph L. Koepfinger*
Thomas Koshy
John D. Kulick
Joseph S. Levy
Howard Li
Gui Lin
Johnny Daozhuang Lin
Xiaohui Liu
Kevin W. Lu
Daleep C. Mohla
Andrew Myles
Paul Nikolich
Annette D. Reilly
Robby Robson
Lei Wang
F. Keith Waters
Karl Weber
Philip B. Winston
Don Wright

*Member Emeritus

Introduction

This introduction is not part of IEEE Std 11073-10471-2023, IEEE Standard for Health informatics—Device Interoperability—Part 10471: Personal health device communication—Device specialization—Independent living activity hub.

ISO/IEEE 11073 standards enable communication between medical devices and external computer systems. This document uses the optimized framework created in ISO/IEEE 11073:20601 and describes a specific, interoperable communication approach for the Independent Living Activity Hub of personal health devices. These standards align with and draw on the existing clinically focused standards to provide support for communication of data from clinical or personal health devices (PHDs).

Contents

Health Informatics—Device Interoperability— Part 10471: Personal Health Device Communication—Device Specialization—Independent Living Activity Hub

1. Overview

1.1 Scope

Within the context of the ISO/IEEE 11073 family of standards for device communication, this standard establishes a normative definition of the communication between independent living activity hubs and managers (e.g., cell phones, personal computers, personal health appliances, and set top boxes) in a manner that enables plug-and-play interoperability. It leverages appropriate portions of existing standards, including ISO/IEEE 11073 terminology and information models. It specifies the use of specific term codes, formats, and behaviors in telehealth environments restricting ambiguity in base frameworks in favor of interoperability. This standard defines a common core of communication functionality for independent living activity hubs. In this context, independent living activity hubs are defined as devices that communicate with simple situation monitors (binary sensors), normalize information received from the simple environmental monitors, and provide this normalized information to one or more managers. This information can be examined, for example, to determine when a person’s activities/behaviors have deviated significantly from what is normal for them such that relevant parties can be notified. Independent living activity hubs will normalize information from the following simple situation monitors (binary sensors) for the initial release of the proposed standard: fall sensor, motion sensor, door sensor, bed/chair occupancy sensor, light switch sensor, smoke sensor, (ambient) temperature threshold sensor, personal emergency response system (PERS), and enuresis sensor (bed-wetting).

1.2 Purpose

This standard addresses a need for an openly defined, independent standard for controlling information exchange to and from personal health devices and managers (e.g., cell phones, personal computers, personal health appliances, and set top boxes). Interoperability is the key to growing the potential market for these devices and to enabling people to be better informed participants in the management of their health.

1.3 Word usage

The word *shall* indicates mandatory requirements strictly to be followed in order to conform to the standard and from which no deviation is permitted (*shall* equals *is required to*).[6,7]

The word *should* indicates that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required (*should* equals *is recommended that*).

The word *may* is used to indicate a course of action permissible within the limits of the standard (*may* equals *is permitted to*).

The word *can* is used for statements of possibility and capability, whether material, physical, or causal (*can* equals *is able to*).

1.4 Context

See IEEE Std 11073-20601™ for an overview of the environment within which this standard is written.[8]

[6] The use of the word *must* is deprecated and cannot be used when stating mandatory requirements; *must* is used only to describe unavoidable situations.
[7] The use of *will* is deprecated and cannot be used when stating mandatory requirements; *will* is only used in statements of fact.
[8] Information on references can be found in Clause 2.

This document, IEEE Std 11073-10471™, defines the device specialization for the independent living activity hub, being a specific agent type, and provides a description of the device concepts, its capabilities, and its implementation according to this standard.

This standard is based on IEEE Std 11073-20601, which in turn draws information from both IEEE Std 11073-10201™ [B5] and ISO/IEEE 11073:20101 [B9].[9] The medical device encoding rules (MDERs) used within this standard are fully described in IEEE Std 11073-20601.

This standard reproduces relevant portions of the nomenclature found in ISO/IEEE 11073:10101 [B8] and adds new nomenclature codes for the purposes of this standard. Between this standard and IEEE Std 11073-20601, all required nomenclature codes for implementation are documented.

The object classes and attributes in this standard are identified by nomenclature codes. Each code consists of a reference identifier (RefID) string and an integer code value. By using a consistent nomenclature, interoperability is enhanced as all implementations maintain the same semantic meaning for the numeric codes. This standard leverages the existing nomenclature codes in ISO/IEEE 11073:10101.

This standard defines specialized nomenclature codes that will be collected in IEEE Std 11073-10101™. Between this standard, IEEE Std 11073-10101, IEEE Std 11073-20601, and other IEEE Std 11073-104xx, all required nomenclature codes for implementation are documented. New codes may be defined in newer versions/revisions of each of these documents. In the case of a conflict, where one term code has been assigned to two separate semantic concepts with different RefIDs, in general the oldest definition that is in actual use should take precedence. The same policy applies when one RefID has two different code values assigned in different specifications. The resolution of such conflicts will be determined through joint action by the responsible workgroups and other stakeholders and any corrective action published as corrigenda.

NOTE—In this standard, IEEE Std 11073-104zz is used to refer to the collection of device specialization standards that utilize IEEE Std 11073-20601, where zz can be any number from 01 to 99, inclusive.[10]

2. Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so that each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

IEEE Std 11073-20601™-2019, Health informatics—Personal health device communication—Part 20601: Application profile—Optimized Exchange Protocol.[11,12]

ISO/IEEE 11073:10101-2020, Health informatics—Point-of-care medical device communication—Part 10101: Nomenclature.[13]

See Annex A for all informative material referenced by this standard.

3. Definitions, acronyms, and abbreviations

For the purposes of this standard, the following terms and definitions apply. *The IEEE Standard Dictionary Online* **Error! Reference source not found.** should be referenced for terms not defined in this clause.

3.1 Definitions

agent: A node that collects and transmits personal health data to an associated manager.

[9] The numbers in brackets correspond to those of the bibliography in Annex A.
[10] Notes in text, tables, and figures are given for information only and do not contain requirements needed to implement the standard.
[11] IEEE publications are available from The Institute of Electrical and Electronics Engineers, Inc. (http://standards.ieee.org/).
[12] The IEEE standards or products referred to in this clause are trademarks of The Institute of Electrical and Electronics Engineers, Inc.
[13] ISO publications are available from the International Organization for Standardization (https://www.iso.org/) and the American National Standards Institute (https://www.ansi.org/).

class: In object-oriented modeling, it describes the attributes, methods, and events that objects instantiated from the class utilize.

compute engine: *See:* **manager**.

device: A term used to refer to a physical apparatus implementing either an agent or a manager role.

handle: An unsigned 16-bit number that is locally unique and identifies one of the object instances within an agent.

manager: A node receiving data from one or more agent systems. Some examples of managers include a cellular phone, health appliance, set top box, or a computer system.

obj-handle: *See*: **handle**.

object: In object-oriented modeling, a particular instantiation of a class. The instantiation realizes attributes, methods, and events from the class.

personal health device: A device used in personal health applications.

personal telehealth device: *See*: **personal health device**.

sensor: An apparatus that measures physical properties. These comprise the primary inputs to an independent living activity hub agent.

3.2 Acronyms and abbreviations

APDU	application protocol data unit
ASN.1	Abstract Syntax Notation One
DIM	domain information model
EUI-64	extended unique identifier (64 bits)
ICS	implementation conformance statement
MDC	medical device communication
MDER	medical device encoding rules
MDS	medical device system
MOC	managed object class
OID	object identifier
PDU	protocol data unit
PERS	personal emergency response system
PHD	personal health device
PIR	Passive Infrared
RT-SA	real-time sample array
VMO	virtual medical object
VMS	virtual medical system

4. Introduction to ISO/IEEE 11073 personal health devices

4.1 General

This standard and the remainder of the series of ISO/IEEE 11073 personal health device (PHD) standards fit in the larger context of the ISO/IEEE 11073 series of standards. The full suite of standards enables agents to interconnect and interoperate with managers and with computerized health-care information systems. See IEEE Std 11073-20601 for a description of the guiding principles for this series of ISO/IEEE 11073 personal health device standards.

IEEE Std 11073-20601 supports the modeling and implementation of an extensive set of personal health devices. This standard defines aspects of the independent living activity hub device. It describes all aspects necessary to implement the application layer services and data exchange protocol between an ISO/IEEE 11073 PHD independent living activity hub agent and a manager. This standard defines a subset of the objects and functionality contained in IEEE Std 11073-20601 and extends and adds definitions where appropriate. All new definitions are given in Annex B in abstract syntax notation one (ASN.1) **Error! Reference source not found.**. All nomenclature codes referenced in this standard are collected in Annex C.

4.2 Introduction to IEEE Std 11073-20601 modeling constructs

4.2.1 General

The ISO/IEEE 11073 series of standards, and in particular IEEE Std 11073-20601, is based on an object-oriented systems management paradigm. The overall system model is divided into three principal components: the domain information model (DIM), the service model, and the communication model. See IEEE Std 11073-20601 for a detailed description of the modeling constructs.

4.2.2 Domain information model (DIM)

The DIM is a hierarchical model that describes an agent as a set of objects. These objects and their attributes represent the elements that control behavior and report on the status of the agent and data that an agent can communicate to a manager. Communication between the agent and the manager is defined by the application protocol in IEEE Std 11073-20601.

4.2.3 Service model

The service model defines the conceptual mechanisms for the data exchange services. Such services are mapped to messages that are exchanged between the agent and the manager. Protocol messages within the ISO/IEEE 11073 series of standards are defined in ASN.1. The messages defined in IEEE Std 11073-20601 can coexist with messages defined in other standard application profiles defined in the ISO/IEEE 11073 series of standards.

4.2.4 Communication model

In general, the communication model supports the topology of one or more agents communicating over logical point-to-point connections to a single manager. For each logical point-to-point connection, the dynamic system behavior is defined by a connection state machine, as specified in IEEE Std 11073-20601.

4.2.5 Implementing the models

An agent implementing this standard shall implement all mandatory elements of the information, service, and communication models as well as all conditional elements where the condition is met. The agent should implement the recommended elements, and it may implement any combination of the optional elements. A manager implementing this standard shall utilize at least one of the mandatory, conditional, recommended, or optional elements. In this context, “utilize” means to use the element as part of the primary function of the manager device. For example, a manager whose primary function is to display data would need to display a piece of data in the element in order to utilize it.

4.3 Compliance with other standards

Devices that comply with this standard may also be required to comply with other domain-and device specific standards that supersede the requirements of this standard with respect to issues including safety, reliability, and risk management. A user of this standard is expected to be familiar with all other such standards that apply and to comply with any higher specifications thus imposed.

Typically, medical devices should comply with the IEC 60601-1:2005 [B1] base standards with respect to electrical and mechanical safety as well as any device-specific standard as might be defined in the IEC 60601-2 [B2] series of standards. Software aspects may apply through standards such as IEC 62304:2006/EN 62304:2006 [B3]. Devices that comply with this standard implement higher layers of network software and utilize lower layers as appropriate to the application. The requirements on performance of such applications and conformance are defined elsewhere and are outside the scope of this standard. Moreover, the use of any medical equipment is subject to risk assessment and risk management appropriate to the application. Some relevant examples are ISO 14971:2007 [B7] and IEC 80001-1:2010 [B4]. The requirements of such risk assessment and risk management and conformance are outside the scope of this standard. The applicable versions of the referenced safety related standards may differ per country.

5. Independent living activity hub device concepts and modalities

5.1 General

This clause presents the general concepts of independent living activity hub devices. In the context of personal health devices in this family of standards, an independent living activity hub is a device that aggregates activity data sensor events from multiple sensor data sources, all of which are used in the support of the independent living of one or more occupants. The occupants' environment may vary greatly and encompass a varying mixture of sensors; therefore, the activity data sensor events reported by any particular agent have a corresponding variance.

5.2 Concepts

While many data generating sensors exist, they have a number of properties in common that influence the design of this standard. Note that these are only generalities employed in the design and there may be instances where an activity data generating sensor exceeds the following properties:

- Price: usually, they are inexpensive sensors.
- Power: typically, they are low power sensors.
- Communication: typically, they use an inexpensive communication technique and are quite often wireless.
- Frequency: typically, they communicate infrequently or only when an event occurs.
- Quantity: there may be a wide range of sensors and many instances of any one sensor type.

It is the responsibility of the independent living activity hub agent to manage all these sensors. Management of the sensors is likely to be within the proprietary purview of the agent both because there is no acceptable existing industry standard, and the desire to integrate existing legacy solutions with the IEEE 11073-20601 framework. Therefore, this responsibility is outside the scope of this standard. Only the communication between the independent living activity hub agent and the manager is covered by this standard.

Additionally, due to the many different types of sensors that may be employed in any particular installation, there is a range of functionality that the corresponding independent living activity hub agent presents to the manager. On the one hand, a fully functional independent living activity hub agent could represent a significant number of sensors and have a complex conversation with the associated manager. On the other hand, a subset of this protocol could be employed by just one sensor such that it would appear to a manager as an independent living activity hub agent with a single sensor.

Figure 1 shows a typical scenario with the type of sensors that might be fitted within a domicile of a person being monitored for their activities of daily living (ADL) and to ensure their safety. Sensors to determine activity and location within the room include passive infrared (PIR) motion, chair and sofa occupancy sensors, and a position sensor worn on the person to provide accurate tracking within the room. Sensors detect when appliances such as the table lamp and television are used. Sensors for safety include a smoke detector and temperature sensor, and a window sensor detects any unexpected opening that might indicate unauthorized entry. Sensors on the door can be used to determine exit and entry and warn of wander. The person also wears a sensor that can detect if they have fallen, and includes a button they can press (PERS) to summon assistance.

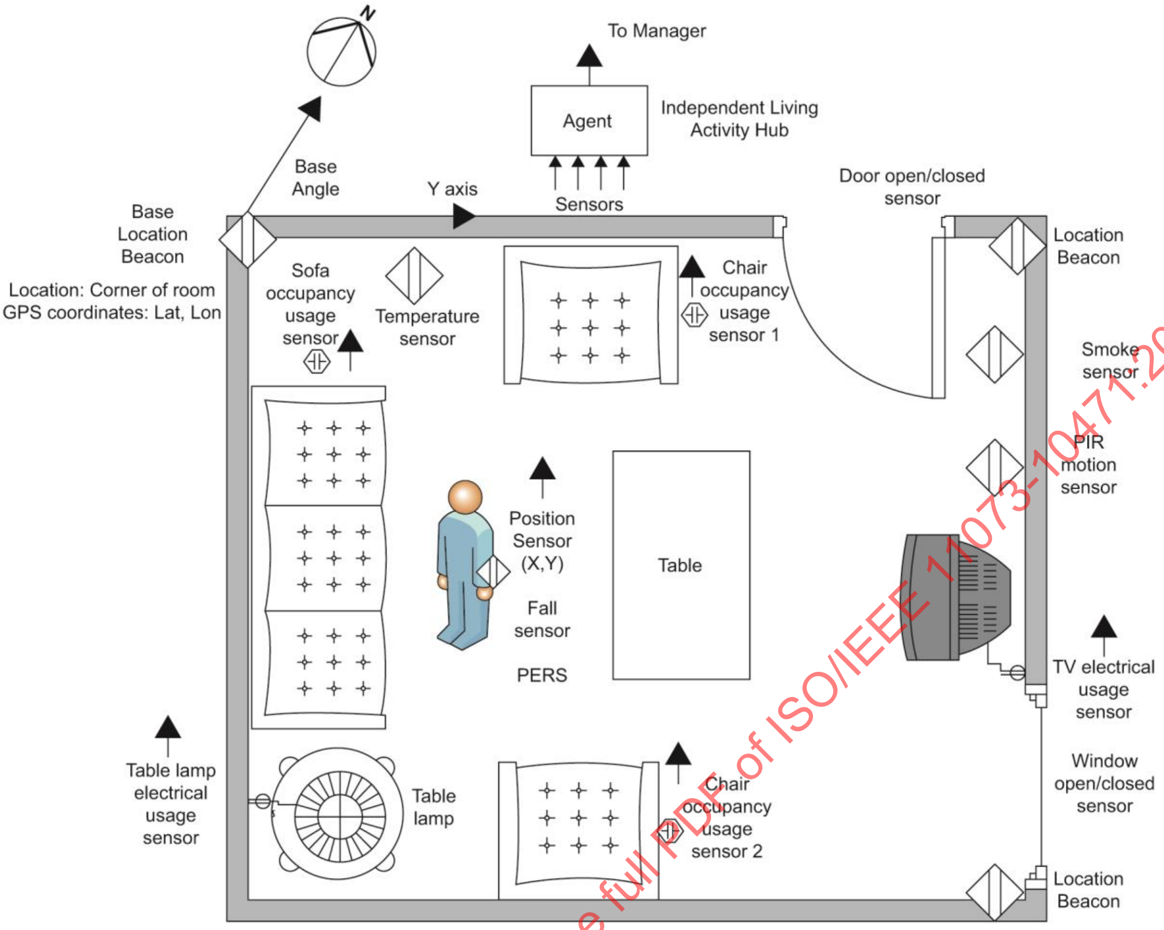

Figure 1—Independent living activity hub—typical scenario

In this scenario, sensors, such as PIR motion, temperature, and smoke are wired back to a common independent living activity hub (or use proprietary legacy wireless connection) that monitors each of the sensors and acts as a proxy to send their data and report their status to the manager. Other sensors, such as occupancy and usage, act as independent living activity hub agents that send their data and report their status direct to the manager.

Position is based on a radio beacon system. The corner of the room is used as the origin for the coordinates, and the beacon at this location is designated as the datum base. The location of the base beacon may be reported as the exact geographic location, as a meaningful descriptive location, or no location, as appropriate for an application. The orientation of the room (base angle) may be reported as the angle between geographic north and the Y axis of the coordinate system.

5.3 Collected data

5.3.1 General

This clause provides an overview of the kinds of sensors and activity data that could be collected. This is not to imply that all independent living activity hub agents would necessarily report values for all of these sensors. Furthermore, this standard is not concerned with the form of the data nor the communication between any actual sensor and the independent living activity hub; rather, only the activity data sensor events derived as a result of that data are considered part of this standard. See Clause 6 for the normative definition of this derived data.

5.3.2 Fall sensor

This sensor is used to notify the monitoring system that a fall sensor event has taken place. This could take the form of a sensor of the type that detects a person’s fall and automatically generates the event. A pre-fall event can be sent if the sensor detects a stumble.

5.3.3 PERS sensor

The personal emergency response system (PERS) sensor is used to notify the monitoring system that a personal emergency sensor event has taken place. This would typically take the form of a button that the person presses to indicate some sort of perceived emergency (“panic button”).

5.3.4 Environmental alert sensors

These sensors generate a sensor event whenever they sense an environmental aspect that is beyond a preset threshold. Examples include smoke sensors, carbon monoxide sensors, water sensors, and natural gas/liquid propane (LP) gas sensors.

5.3.5 Motion sensor

This sensor generates a sensor event whenever it has sensed movement within its range above a preset level. This type of sensor is typically employed in two manners: general motion and intruder detection.

In the case of general motion, the detection of the motion causes an immediate generation of a sensor event and subsequent action. This may be used for tracking the activity level of the occupant to discern whether behavior patterns have altered.

In the case of intruder detection, the motion sensor event could be used to trigger intruder detection actions.

Optionally, in the case of the primary entrances to the building; for example, the subsequent action to the sensor event may be delayed for some period before taken. This is to allow for an authorized person to enter the premises and to disable the sensor event before the action is taken (e.g., in the case where a triggered sensor event will generate an intruder alert). Should the proper disabling not take place within the expected delay time period, the normal subsequent action would take place. There may also be a sensor event for the case when the motion sensor detects someone attempting to tamper with its function.

5.3.6 Property exit sensor

This sensor generates a sensor event whenever it has sensed the exit of an occupant from the premises. This is commonly employed when there is an occupant with some cognitive issues who would encounter difficulties in an unfamiliar environment. There may also be a sensor event for the case where the premises’ exit is left open.

5.3.7 Enuresis sensor

This sensor generates a sensor event when it detects occurrences of involuntary urination or bed-wetting. This sensor could be utilized in a range of settings such as a bed, chair, or any similar structure.

5.3.8 Contact closure sensor

This sensor issues a sensor event whenever a contact is opened or closed. This sensor reports the state of the contact after a transition, either from closed to open or from open to closed. Only a single sensor event is sent for each transition. Examples of where this sensor would be deployed are passageway doors, cupboard doors, drawers, windows, and pressure mats.

5.3.9 Usage sensor

This sensor issues a sensor event to denote the start of use (into a bed/chair, for example) or the end of use (out of a bed/chair, for example). It also issues a sensor event for an anticipated usage not occurring by an expected preset time (expected use start violation) as well as a sensor event for the usage continuing beyond an expected preset time (expected use stop violation). Additionally, there would be a sensor event generated if during an expected usage time, the usage is discontinued for longer than a preset period of time (intermittent absence violation). An example would be that during a normal sleep period, a person got out of the bed and did not return in an expected period of time.

5.3.10 Switch use sensor

This sensor issues a sensor event for a switch changing states either to the used state (ON) or to the unused state (OFF). Examples of this are light switches, fan switches, and other similar switches that control electrical apparatus.

5.3.11 Simple medication dispenser

The dispenser is a container that contains doses of one or more medications. The medications are to be taken in predetermined doses at predetermined times. A sensor event is generated for a presented dosage being taken from the dispenser (dosage taken) and/or for a dose not being taken after a predetermined amount of time (dosage missed). An event is generated when the dispenser becomes empty of medication and requires to be refilled.

NOTE—The sensor description presented here is of a conceptual model to aid in understanding. Be aware that the same derived sensor events could be generated through many other means (such as a user interaction with some screen interface), and devices may not be able to generate all of these events. This standard is only concerned with the generated sensor events.

5.3.12 Temperature alert sensor

This sensor monitors the temperature in an environment. It issues sensor events based on the sensor value being outside of a preset temperature limit. The sensor events could be that the ambient temperature has risen above a certain level (high threshold) or dropped below a certain level (low threshold), or that the rate-of-change is faster than a predetermined expected rate (rate-of-change). This can be used for detecting conditions such as the temperature of a dwelling being dangerously high/low, or that stove elements have been left on after cooking has been completed.

5.3.13 Relative humidity alert sensor

This sensor monitors the relative humidity in an environment. It issues sensor events based on the sensor value being outside of a preset humidity limit. The sensor events could be that the ambient relative humidity has risen above a certain level (high threshold). This can be used for detecting conditions such as the dangerously high humidity of a dwelling.

5.3.14 Ambient temperature sensor

This sensor monitors the temperature in an environment and reports the current temperature if capable, or the most recent temperature reading when the manager initiates a data request.

5.3.15 Ambient relative humidity sensor

This sensor monitors the humidity in an environment and reports the current relative humidity if capable, or the most recent relative humidity reading when the manager initiates a data request.

5.3.16 Ambient pressure sensor

This sensor monitors the pressure in an environment and reports the current pressure if capable, or the most recent pressure reading when the manager initiates a data request.

5.3.17 Pollen sensor

This sensor monitors the level of pollen in an environment and reports the level if capable, or the most recent pollen level reading when the manager initiates a data request. The pollen sensor may send readings when the level is elevated.

5.3.18 Dust particle sensor

This sensor monitors the level of dust particles in an environment and reports the current dust count if capable or the most recent dust particle count when the manager initiates a data request. The dust particle sensor may send readings when the level is elevated.

5.3.19 Utility accumulated usage sensor

This sensor reports on the total accumulated usage of a utility such electricity, gas, or water.

5.3.20 Wind speed

Wind speed is reported sepately as velocity and direction.

5.3.20.1 Wind velocity

This sensor reports on velocity of the wind.

5.3.20.2 Wind direction

This sensor reports on the direction from which the wind is blowing. Expressed in degrees, 0 degrees indicates the wind is from due north.

5.3.21 Utility accumulated usage

These sensors report on the accumulated usage of a utility, including electricity, gas, and water for the household or individual appliance.

5.3.21.1 Electricity accumulated usage sensor

This sensor reports the total accumulated usage of electricity and reports the current accumulated reading if capable or the most recent accumulated reading when the manager initiates a data request. Usage over a period of time is determined by taking successive readings. This sensor could also report readings at predefined intervals.

5.3.21.2 Gas accumulated usage sensor

This sensor reports the total accumulated usage of gas and reports the most reports the current accumulated reading if capable or the most recent accumulated reading when the manager initiates a data request. Usage over a period of time is determined by taking successive readings. This sensor could also report readings at predefined intervals.

5.3.21.3 Water accumulated usage sensor

This sensor reports the total accumulated usage of water and reports the current accumulated reading if capable or the most recent accumulated reading when the manager initiates a data request. Usage over a period of time is determined by taking successive readings. This sensor could also report readings at predefined intervals.

5.3.22 Utility instantaneous usage sensor

These sensors report on the instantaneous usage of a utility, including electricity, gas, and water for the household or individual appliance.

5.3.22.1 Electricity instantaneous usage sensor

This sensor reports the instantaneous usage of electricity and reports the current reading if capable, or the most recent reading when the manager initiates a data request. This sensor could also report readings at predefined intervals or based on predefined trigger events such as usage.

5.3.22.2 Gas instantaneous usage sensor

This sensor reports the instantaneous usage of gas and reports the current reading if capable, or the most recent reading when the manager initiates a data request. This sensor could also report readings at predefined intervals or based on predefined trigger events such as usage.

5.3.22.3 Water instantaneous usage sensor

This sensor reports the instantaneous usage of water and reports the current reading if capable, or the most recent reading when the manager initiates a data request. This sensor could also report readings at predefined intervals or based on predefined trigger events such as usage.

5.3.23 Solar accumulated generation sensor

This sensor reports on the total accumulated generation of energy by photovoltaic cells installed in a location and reports the current accumulated generation reading if capable, or the most recent accumulated reading when the manager initiates a data request. Usage over a period of time is determined by taking successive readings. This sensor could also report readings at predefined intervals.

5.3.24 Solar instantaneous generation sensor

This sensor reports on the instantaneous generation of energy by photovoltaic cells installed in a location and reports the current reading if capable, or the most recent reading when the manager initiates a data request. This sensor could also report readings at predefined intervals or based on predefined trigger events such as usage.

5.3.25 Location

This sensor reports its current location as an enumeration of one of the location codes and can indicate if the sensor is at or not at the location. The sensor may also emit a string that provides a descriptive annotation of the location such as a human readable string or might report the identity of a beacon in a location system.

5.3.26 Relative location coordinates

Location within a defined space, such as a room, building, or site are reported as the Cartesian coordinates {X, Y, Z} relative to a datum base coordinate (origin) or specified identified datum base location. If the datum base coordinate is specified, then geographic coordinates are used as the datum base X (latitude), Y (longitude), and Z (altitude) coordinates, together with the angle of the Y axis of the local coordinate space relative to geographic coordinates (±180°). Alternatively, the datum base coordinate may be specified as a well-defined location within the defined space together with the angle of the Y axis of the local coordinate space relative to geographic coordinates. The location may be specified as one of the location codes and/or a string to describe the location.

If a datum base coordinate or the angle is not specified, then the relative coordinates are interpreted as appropriate for the location system within the defined space, such as a space defined by a wireless location system.

5.3.26.1 Relative Location coordinates

This sensor reports the {X, Y, Z} coordinates of the relative location of a device within a defined space.

5.3.26.2 Base coordinate

This metric object reports the geographic coordinates (latitude, longitude, and altitude) for the datum base coordinate (origin) of the defined space. The location is reported by two metric objects: longitude and latitude, and altitude.

5.3.26.3 Base angle

This metric object reports the angle of the Y axis of the local coordinate space relative to geographic coordinates.

5.3.26.4 Base coordinate location

This metric object reports the datum base coordinate (origin) of the defined space as a well-defined physical location. The location may be specified as one of the location codes and/or a string to describe the location.

5.3.27 GPS location coordinates

This sensor reports the location of the device as the GPS (geographic) coordinates of longitude, latitude, and altitude. In many cases only longitude and latitude are reported, and for this reason GPS location is reported by two metric objects: longitude and latitude, and altitude.

5.3.28 Speed and heading

These sensors report separately the speed and heading of the device.

5.3.28.1 Speed

This sensor reports the absolute speed of the device.

5.3.28.2 Heading

This sensor reports the heading relative to geographic coordinates

5.3.29 Identity

This object reports an opaque string that is a used as a locally unique identifier of the person or object associated with other sensors in an independent living activity hub agent. This could include such information as identification from an RFID tag, or patient identity number. If the identity object is present, then those sensors associated with a person shall have the Source-Handle-Reference set to the value of the handle of the identity object.

5.3.30 Encounter

This object reports an opaque string that is a used to provide a locally unique identifier of the person or object that has been encountered by this agent. This could include such information as identification from RFID tag, BlueTooth MAC address, or patient identity number. The form of encounter is not specified and could take the form of detection of being in close proximity.

5.4 Reporting data from multiple persons

5.4.1 Identification of a single person

IEEE 11073 contains several mechanisms that permit a device or observation to be associated with a specific person or object. The medical device system (MDS) of every IEEE 11073 device includes a unique extended unique identifier (64 bits) (EUI-64) that may be used to associate a specific device with a single user, and thereby associate all observations from that device with that patient or object.

5.4.2 Identification of multiple persons using multiple person dynamic data update

Where a single device is used to report observations from multiple persons, the multiple person form of the dynamic data update (6.5.3) may be used to report the observations of each individual. A person number is allocated to each individual and observations with that person number as header in the multiple persons dynamic data update are associated with that person. This approach has the disadvantage that the association of a sensor with a person is unknown until an observation is reported and the association of a sensor may be to multiple persons. There is a further disadvantage that the person number must be mapped a-priroi to each specific person for whom observations will be reported.

5.4.3 Identification of multiple persons using the identity object

The independent living activity hub may act as proxy for a sensor that is used by people who have previously not been configured in the sensor. This may be the case for entry systems where a card reader is used to identify the individual when they present a card to gain entry to a building. Information regarding which observation is associated with which person is provided by using the identity object. An identity object is instantiated and used to report the locally unique identification as an observation. If the Source-Handle-Reference attribute of a sensor object references the identity object, then the locally unique identity of a person can be associated to an observation from the sensor.

Figure 2 shows how an enumeration object references an associated identity object.

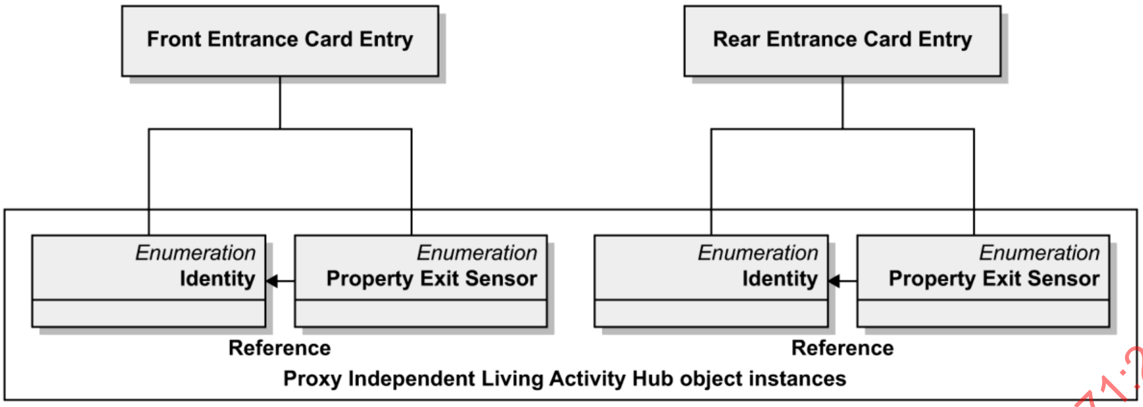

Figure 2—Identification of multiple persons using the identity object

Figure 3 provides an example of the use of the identity object to provide the identify of a person entering though the front door of a property and later leaving by the back door. An enumeration object references the associated identify object by using the value of handle of the identity object in the source reference handle field. The time stamp fields shall have the same value.

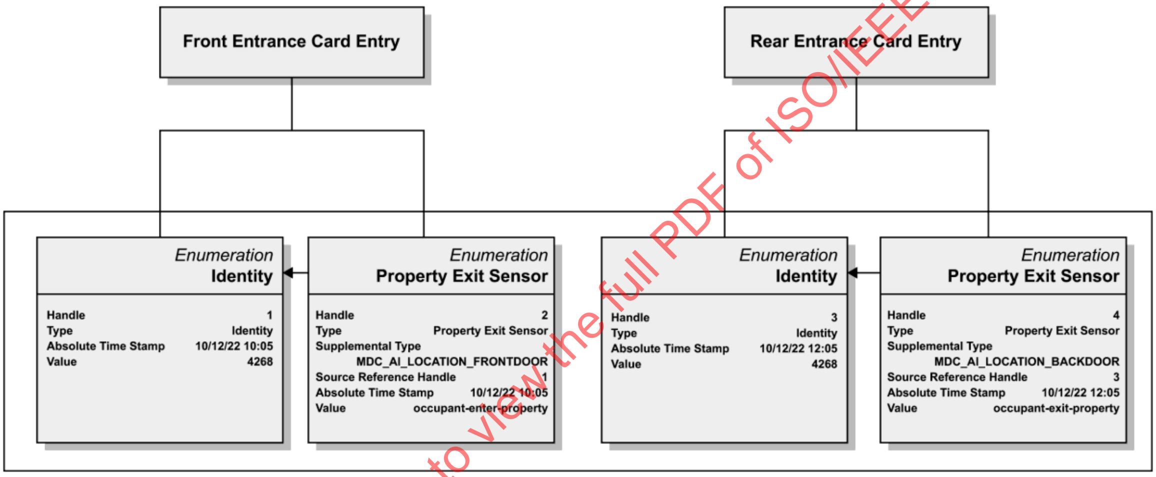
Front Entrance Card Entry
Rear Entrance Card Entry
Enumeration
Identity
Handle 1
Type Identity
Absolute Time Stamp 10/12/22 10:05
Value 4268
Enumeration
Property Exit Sensor
Handle 2
Type Property Exit Sensor
Supplemental Type MDC_AI_LOCATION_FRONTDOOR
Source Reference Handle 1
Absolute Time Stamp 10/12/22 10:05
Value occupant-enter-property
Enumeration
Identity
Handle 3
Type Identity
Absolute Time Stamp 10/12/22 12:05
Value 4268
Enumeration
Property Exit Sensor
Handle 4
Type Property Exit Sensor
Supplemental Type MDC_AI_LOCATION_BACKDOOR
Source Reference Handle 3
Absolute Time Stamp 10/12/22 12:05
Value occupant-exit-property

Figure 3—Example use of the identity object

6. Independent living activity hub domain information model

6.1 Overview

This clause describes the domain information model of the independent living activity hub.

6.2 Class extensions

In this standard, no class extensions are defined with respect to IEEE Std 11073-20601.

6.3 Object instance diagram

The object instance diagram of the independent living activity hub domain information model, defined for the purposes of this standard, is shown as Figure 4 and Figure 5.

The generic DIM of the independent living activity hub that is presented in Figure 4 and Figure 5 defines all possible data objects. However, it would be expected that most independent living activity hubs would implement only a restricted subset of the data objects. An independent living activity hub shall implement at least one sensor instance.

The objects of the DIM, as shown in Figure 4 and Figure 5, are described in 6.4 to 6.12. This includes the medical device system (MDS) object (see 6.5), the numeric objects (see 6.6), the real-time sample array (RT-SA) objects (see 6.7), the enumeration objects (see 6.8), the PM-store objects (see 6.9), and the scanner objects (see 6.10). See 6.12 for rules for extending the independent living activity hub information model beyond elements as described in this standard. Each clause that describes an object of the independent living activity hub contains the following information:

— The nomenclature code used to identify the class of the object. One example of where this code is used is the configuration event, where the object class is reported for each object. This allows the manager to determine whether the class of the object being specified is a numeric, real-time sample array, enumeration, scanner, or PM-store class.

— The attributes of the object. Each object has attributes that represent and convey information on the activity data generating sensor and its data sources. Each object has a Handle attribute that identifies the object instance within an agent. Attribute values are accessed and modified using communication services such as GET and SET. Attribute types are defined using ASN.1. The ASN.1 definitions for new attribute types specific to this standard are in Annex B, and the ASN.1 definitions for existing attribute types referenced in this standard are in IEEE Std 11073-20601.

— The methods available on the object.

— The potential events generated by the object. Data are sent to the manager using events.

— The available services such as getting or setting attributes.

The attributes for each class are defined in tables that specify the name of the attribute, its value, and its qualifier. The qualifiers mean M—Attribute is Mandatory, C—Attribute is Conditional and depends on the condition stated in the Remark or Value column (if IEEE Std 11073-20601 is referenced, then it contains the conditions), R—Attribute is Recommended, NR—Attribute is Not Recommended, and O—Attribute is Optional. Mandatory attributes shall be implemented by the agent. Conditional attributes shall be implemented if the condition applies and may be implemented otherwise. Recommended attributes should be implemented by the agent. Not recommended attributes should not be implemented by the agent. Optional attributes may be implemented by the agent. For attributes with qualifiers set to R or NR, underlying requirements stated in the Remark and Value column in IEEE Std 11073-20601 shall be followed. If any attribute (from the DIM of IEEE Std 11073-20601) is not included in the definition of that object in this standard, it shall not be included in that object by an implementation, unless it is a vendor-specific attribute extended according to 6.12.

An attribute is further qualified as static, dynamic, or observational. Static attributes shall not change value during the life of an association. Dynamic attributes have a value that may change during the life of an association. The dynamic attribute value should be sent at configuration time and shall be sent at or before the time when the value would be needed for interpreting a reported observation. Observational attributes have a value that may change during the life of an association. When a set of observational attribute values are received, these values are combined with the available context information (i.e., all related dynamic and static attribute values) to represent the observation at the observation time.

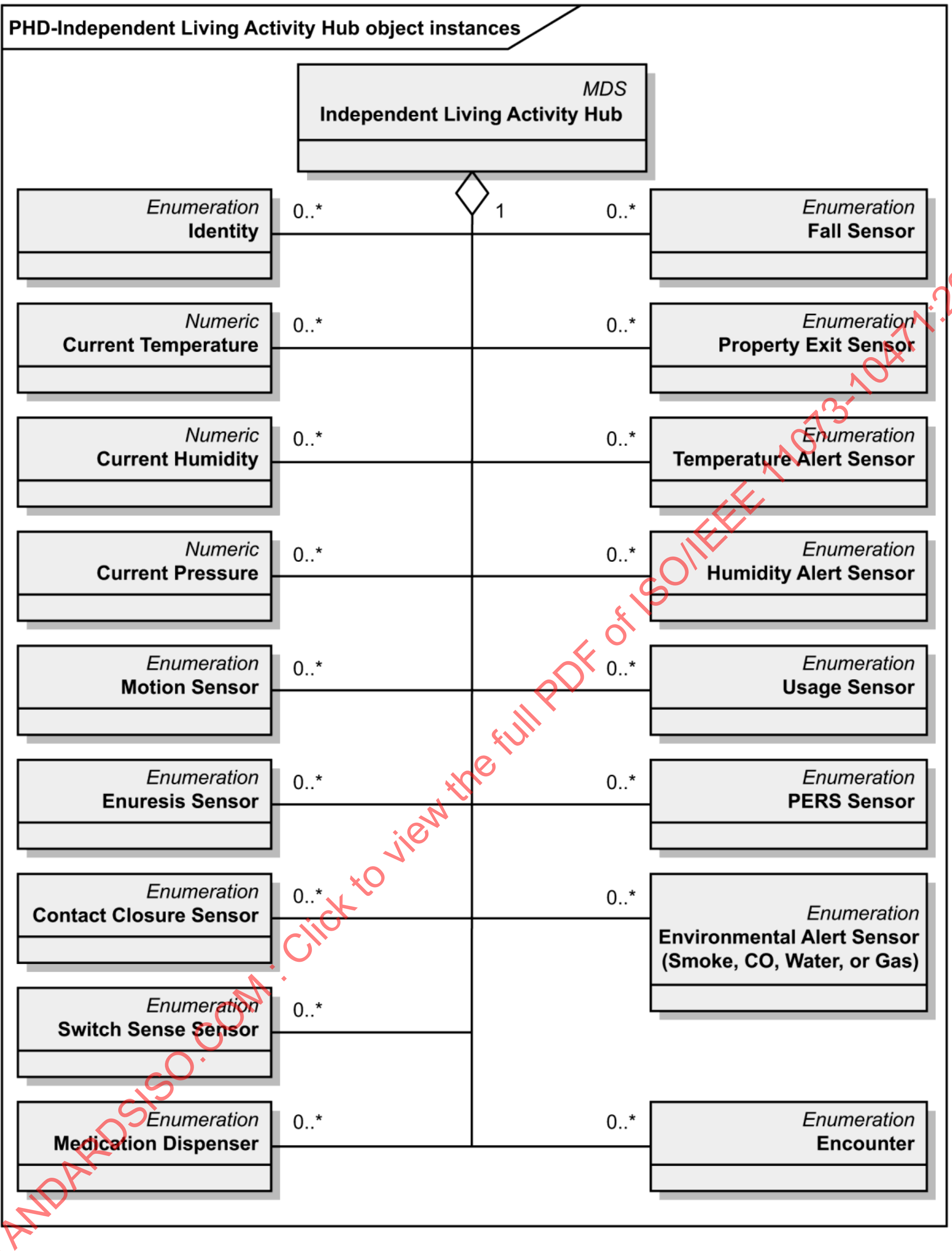

Figure 4—Independent living activity hub—object instances

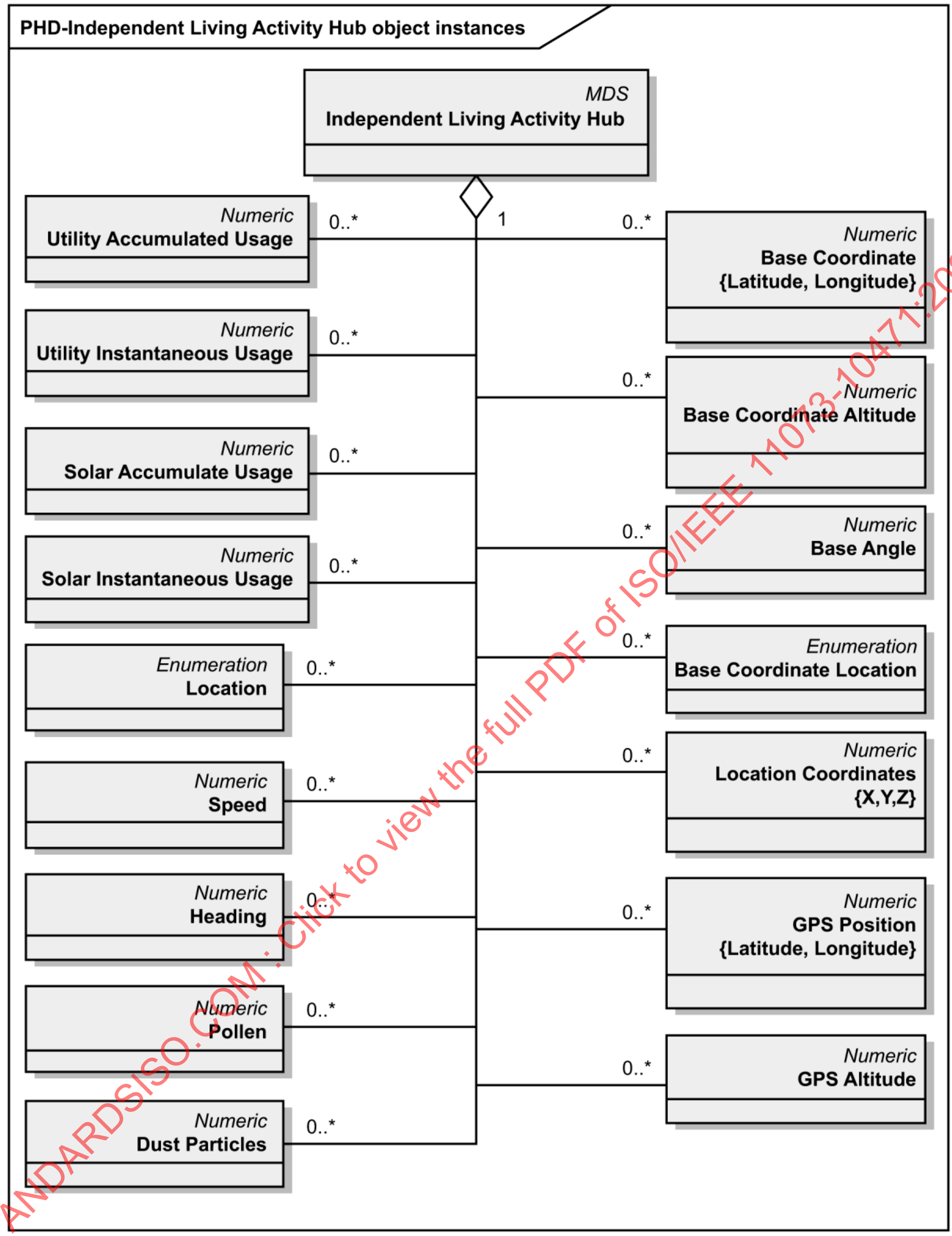

Figure 5—Independent living activity hub—object instances

6.4 Types of configuration

6.4.1 General

As specified in IEEE Std 11073-20601, there are two styles of configuration available. Subclauses 6.4.2 and 6.4.3 briefly introduce standard and extended configurations.

6.4.2 Standard configuration

This standard does not define any standard configurations since the set of sensors for each agent configuration is likely to vary significantly for each deployment scenario. Therefore, all configurations shall be specified as extended configurations.

6.4.3 Extended configuration

In extended configurations, the agent's configuration is not predefined in a standard. The agent determines which objects, attributes, and values that it wants to use in a configuration and assigns a configuration identifier. When the agent associates with a manager, it negotiates an acceptable configuration. Typically, the manager does not recognize the agent's configuration on the first connection, so the manager responds that the agent must send its configuration information as a configuration event report. However, if the manager already understands the configuration, either because it was preloaded in some way or the agent had previously associated with the manager, then the manager responds that the configuration is known and no further configuration information needs to be sent.

6.5 Medical device system object

6.5.1 MDS object attributes

Table 1 summarizes the attributes of the independent living activity hub MDS object. The nomenclature code to identify the MDS class is MDC_MOC_VMS_MDS_SIMP.

Table 1—MDS object attributes

Attribute name	**Value**	**Qual.**
Handle	0	M
System-Type-Spec-List	{MDC_DEV_SPEC_PROFILE_AI_ACTIVITY_HUB,2}. If an agent follows a profile within this specialization this profile shall be specified in the System-Type-Spec-List.	M
System-Type	Attribute not present. See IEEE Std 11073-20601.	NR
System-Model	{“Manufacturer”,”Model”}.	M
System-Id	EUI-64.	M
Dev-Configuration-Id	Extended configs: 0x4000–0x7FFF.	M
Attribute-Value-Map	See IEEE Std 11073-20601.	C
Production-Specification	See IEEE Std 11073-20601.	O
Mds-Time-Info	See IEEE Std 11073-20601.	C
Date-and-Time	See IEEE Std 11073-20601.	C
Base-Offset-Time	See IEEE Std 11073-20601.	R
Relative-Time	See IEEE Std 11073-20601.	NR
HiRes-Relative-Time	See IEEE Std 11073-20601.	O
Date-and-Time-Adjustment	See IEEE Std 11073-20601.	C
Power-Status	*onBattery* or *onMains*.	R
Battery-Level	See IEEE Std 11073-20601.	R
Remaining-Battery-Time	See IEEE Std 11073-20601.	R
Reg-Cert-Data-List	See IEEE Std 11073-20601.	O
Confirm-Timeout	See IEEE Std 11073-20601.	O
Transport-Timeout	See IEEE Std 11073-20601.	O
NOTE—See IEEE Std 11073-20601 for information on whether an attribute is static, or dynamic.		

NOTE—See 6.3 for a description of the qualifiers.

In the response to a Get MDS object command, only implemented attributes and their corresponding values are returned.

See IEEE Std 11073-20601 for descriptive explanations of the individual attributes as well as for information on attribute ID and attribute type.

The Dev-Configuration-Id attribute holds a locally unique 16-bit identifier that identifies the device configuration. For an independent living activity hub agent with extended configuration, this identifier is chosen in the range of extended-config-start to extended-config-end (see IEEE Std 11073-20601) as shown in Table 1.

The agent sends the Dev-Configuration-Id during the Associating state (see 8.3) to identify its configuration for the duration of the association. If the manager already holds the configuration information relating to the Dev-Configuration-Id, it recognizes the Dev-Configuration-Id. Then the Configuring state (see 8.4) is skipped; the agent and manager then enter the Operating state. If the manager does not recognize the Dev-Configuration-Id, the agent and manager enter the Configuring state.

If an agent implements multiple IEEE 11073-104zz specializations, System-Type-Spec-List is a list of type/version pairs, each referencing the respective device standard and version of that standard.

6.5.2 MDS object methods (actions)

Table 2 defines the methods (actions) of the MDS object. These methods are invoked using the Action service. In Table 2, the Subservice type name column defines the name of the method. The Mode column defines whether the method is invoked as an unconfirmed action (i.e., roiv-cmip-action from IEEE Std 11073-20601) or a confirmed action (i.e., roiv-cmip-confirmed-action). The Subservice type (action-type) column defines the nomenclature code used in the action-type field of an action request and response (see IEEE Std 11073-20601). The Parameters (action-info-args) column defines the associated ASN.1 data structure (see IEEE Std 11073-20601 for ASN.1 definitions) to use in the action message for the action-info-args field of the request. The Results (action-info-args) column defines the structure to use in the action-info-args of the response.

Table 2—MDS object methods (actions)

Service	Subservice type name	Mode	Subservice type (action-type)	Parameters (action-info-args)	Results (action-info-args)
ACTION	Set-Time	Confirmed	MDC_ACT_SET_TIME	SetTimeInvoke	—
	Set-Base-Offset-Time	Confirmed	MDC_ACT_SET_BO_TIME	SetBOTimeInvoke	—
	MDS-Data-Request	Confirmed	MDC_ACT_DATA_REQUEST	DataRequest	DataResponse

- **Set-Time:**
  This method allows the manager to set a real-time clock in the agent with the absolute time. The agent indicates whether the Set-Time command is valid using the mds-time-capab-set-clock bit in the Mds-Time-Info attribute (see IEEE Std 11073-20601). Agents with an internal real-time clock shall indicate this capability by also setting the mds-time-capab-real-time-clock bit in the Mds-Time-Info attribute. The Set-Time method can only be supported if the Absolute-Time-Stamp attribute is supported.
- **Set-Base-Offset-Time:**
  This method allows the manager to set a real-time clock in the agent with the base time and offset. The agent indicates whether the Set-Base-Offset-Time command is valid using the mds-time-capab-set-clock bit in the Mds-Time-Info attribute (see IEEE Std 11073-20601). The Set-Base-Offset-Time method can only be supported if the Base-Offset-Time-Stamp attribute is supported.
- **MDS-Data-Request:**
  This method allows the manager system to request measurement data from an agent (see IEEE Std 11073-20601 for a description).

Agents following only this device specialization and no others shall send event reports (see 6.5.3) using agent-initiated measurement and/or manager-initiated measurement data transmission. Agents following this device specialization shall send event reports or request data in the appropriate fashion. During the association procedure (see 8.3), DataReqModeCapab shall be set to the appropriate value for the event report or data request style. As a result, the manager shall assume that if the independent living activity hub agent supports MDS-Data-Request, then it may access the object value only if the object's Metric-Spec-Small attribute has its mss-acc-manager-initiated bit set (see IEEE Std 11073-20601 for additional information).

6.5.3 MDS object events

Table 3 defines the events sent by the independent living activity hub MDS object:

Table 3—Independent living activity hub MDS object events

Service	**Subservice type name**	**Mode**	**Subservice type (event-type)**	**Parameters (event-info)**	**Results (event-reply-info)**
EVENT REPORT	MDS-Configuration-Event	Confirmed	MDC_NOTI_CONFIG	ConfigReport	ConfigReportRsp
	MDS-Dynamic-Data-Update-Var	Confirmed	MDC_NOTI_SCAN_REPORT_VAR	ScanReportInfoVar	—
	MDS-Dynamic-Data-Update-Fixed	Confirmed	MDC_NOTI_SCAN_REPORT_FIXED	ScanReportInfoFixed	—
	MDS-Dynamic-Data-Update-MP-Var	Confirmed	MDC_NOTI_SCAN_REPORT_MP_VAR	ScanReportInfoMPVar	—
	MDS-Dynamic-Data-Update-MP-Fixed	Confirmed	MDC_NOTI_SCAN_REPORT_MP_FIXED	ScanReportInfoMPFixed	—

— **MDS-Configuration-Event:**
This event is sent by the independent living activity hub agent during the configuring procedure if the manager does not already know the independent living activity hub agent's configuration from past associations or because the manager has not been implemented to recognize the configuration according to the independent living activity hub device standard. The event provides static information about the supported measurement capabilities of the independent living activity hub agent.

— **MDS-Dynamic-Data-Update-Var:**
This event provides dynamic measurement data from the independent living activity hub agent for the independent living activity data objects. These data are reported using a generic attribute list variable format. The event is sent as an unsolicited message by the agent (i.e., an agent-initiated measurement data transmission). See 8.5.3 for more information on unsolicited event reporting.

— **MDS-Dynamic-Data-Update-Fixed:**
This event provides dynamic measurement data from the independent living activity hub agent for the independent living activity data objects. These data are reported in the fixed format defined by the Attribute-Value-Map attribute of the object(s). The event is sent as an unsolicited message by the agent (i.e., an agent-initiated measurement data transmission). See 8.5.3 for more information on unsolicited event reporting.

— **MDS-Dynamic-Data-Update-MP-Var:**
This is the same as MDS-Dynamic-Data-Update-Var, but allows inclusion of data from multiple people.

— **MDS-Dynamic-Data-Update-MP-Fixed:**
This is the same as MDS-Dynamic-Data-Update-Fixed, but allows inclusion of data from multiple people.

NOTE—IEEE Std 11073-20601 requires that managers support all of the MDS object events listed above.

6.5.4 Other MDS services

6.5.4.1 GET service

An independent living activity hub agent shall support the GET service to retrieve the values of all implemented attributes of the MDS object. The GET service can only be invoked after the manager has confirmed selection of the configuration of the Agent.

The GET request for all attributes shall be supported. An attribute-id-list parameter may be supported.

The manager may request the MDS object attributes of the independent living activity hub agent; in which case, the manager shall send the "Remote Operation Invoke | Get" message (see roiv-cmip-get in IEEE Std 11073-20601) with the reserved MDS handle value of 0. The independent living activity hub agent shall report its MDS object attributes to the manager using the "Remote Operation Response | Get" message (see rors-cmip-get in IEEE Std 11073-20601). See Table 4 for a summary of the GET service including some message fields.

Table 4—Independent living activity hub MDS object GET service

Service	Subservice type name	Mode	Subservice type	Parameters	Results
GET	<na>	<implied confirmed>	<na>	GetArgumentSimple=(obj-handle=0), attribute-id-list *<optional>*	GetResultSimple=(obj-handle=0), attribute-list

See 8.5.2 for details on the procedure for getting the MDS object attributes.

6.5.4.2 SET service

The independent living activity hub standard does not require an implementation to support the MDS object SET service.

6.6 Numeric objects

6.6.1 General

The independent living activity hub requires one numeric data object for each supported sensor instance. Objects instantiated from the numeric class are used to report the environmental data. The nomenclature code to identify the enumeration class is MDC_MOC_VMO_METRIC_NU. The attributes of a numeric data object are shown in Table 5. The attribute structure shown in Table 5 is common to all sensor types. Later clauses define the precise definitions for each sensor type.

Sometimes, the interpretation of one attribute value in an object depends on other attribute values in the same object. For example, Unit-Code and Unit-LabelString provide context for the observed values. Whenever a contextual attribute changes, the agent shall report these changes to the manager using an MDS object event (see 6.5.3) prior to reporting any of the dependent values.

Table 5—Activity data numeric object attributes

Attribute name	Extended configuration	
	Value	Qual.
Handle	See IEEE Std 11073-20601.	M
Type	Any of the MDC_PART_PHD_AI, MDC_AI_TYPE_* ID's. See Annex C.	M

Attribute name	Extended configuration	
Supplemental-Types	The value that denotes the sensor location. This location is a 16-bit value constructed where the high 11 bits are one of the MDC_AI_LOCATION family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). As a list, this allows for a hierarchical location, such as {building_3, bedroom_5, chair_2}. All of these values are from the Partition ID: MDC_PART_PHD_AI. If the sensor reports general household values or is not in a static location (e.g., GPS location of a person), then its use is optional. If reporting values for a specific location or appliance, then its use is mandatory.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate (See note)	M
Metric-Structure-Small	See IEEE Std 11073-20601.	NR
Measurement-Status	See IEEE Std 11073-20601.	C
Metric-Id	See IEEE Std 11073-20601.	NR
Metric-Id-List	See IEEE Std 11073-20601.	C
Metric-Id-Partition	See IEEE Std 11073-20601.	C
Unit-Code	See IEEE Std 11073-20601.	C
Attribute-Value-Map	See IEEE Std 11073-20601.	C
Source-Handle-Reference	The value references the handle of the identity object to which the sensor is associated if applicable.	C
Label-String	See IEEE Std 11073-20601.	O
Unit-LabelString	See IEEE Std 11073-20601.	O
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
Relative-Time-Stamp	See IEEE Std 11073-20601.	NR
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Measure-Active-Period	See IEEE Std 11073-20601.	NR
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Accuracy	See IEEE Std 11073-20601.	O
Context-Id	See IEEE Std 11073-20601.	NR

NOTE 1—See IEEE Std 11073-20601 for information on whether an attribute is static, dynamic, or observational.

NOTE 2—The settings for Metric-Spec-Small with respect to agent-initiated data transfer and manager initiated request for mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate and associated settings will depend on implementation and explicit use of the sensor.

NOTE—See 6.3 for a description of the qualifiers.

The numeric object does not support any methods, events, or other services.

See IEEE Std 11073-20601 for descriptive explanations on the individual attributes as well as for information on attribute ID and attribute type.

6.6.2 Ambient temperature

Table 6 summarizes the attributes of the ambient temperature numeric object.

Table 6—Ambient temperature numeric object attributes

Attribute name	**Extended configuration**	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_SCADA, MDC_TEMP_ROOM }	M
Supplemental-Types	The value that denotes the sensor location. This location is a 16-bit value constructed where the high 11 bits are one of the MDC_AI_LOCATION family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_DEGC or MDC_DIM_FAHR	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.3 Ambient relative humidity

Table 7 summarizes the attributes of the current humdity numeric object.

Table 7—Ambient relative humidity numeric object attributes

Attribute name	**Extended configuration**	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_SCADA, MDC_REL_HUMIDITY_AMBIENT }	M

Attribute name	Extended configuration	
Supplemental-Types	The value that denotes the sensor location. This location is a 16-bit value constructed where the high 11 bits are one of the MDC_AI_LOCATION family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-periodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_PERCENT	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.4 Ambient pressure

Table 8 summarizes the attributes of the current pressure numeric object.

Table 8—Ambient pressure numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_SCADA, MDC_PRESS_AIR_AMBIENT}	M
Supplemental-Types	The value that denotes the sensor location. This location is a 16-bit value constructed where the high 11 bits are one of the MDC_AI_LOCATION family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_KILO_PASCAL, MDC_DIM_BAR, MDC_DIM_MILLI_BAR, MDC_DIM_MMHG, MDC_DIM_INHG, MDC_DIM_PSI.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.5 Pollen

Table 9 summarizes the attributes of the current pollen numeric object.

Table 9—Pollen numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**

Attribute name	**Extended configuration**	
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_SCADA, MDC _LEVEL_POLLEN}	M
Supplemental-Types	The value that denotes the sensor location. This location is a 16-bit value constructed where the high 11 bits are one of the MDC_AI_LOCATION family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_PER_DECI_L, MDC_DIM_MILLI_G_PER_M_CUBE, MDC_DIM_MICRO_G_PER_M_CUBE	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.6 Dust Count

Table 10 summarizes the attributes of the current dust count numeric object.

Table 10 —Dust count numeric object attributes

Attribute name	**Extended configuration**	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_SCADA, MDC_LEVEL_DUST}	M
Supplemental-Types	The value that denotes the sensor location. This location is a 16-bit value constructed where the high 11 bits are one of the MDC_AI_LOCATION family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_PER_DECI_L, MDC_DIM_MILLI_G_PER_M_CUBE, MDC_DIM_MICRO_G_PER_M_CUBE	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.7 Wind velocity

Table 11 summarizes the attributes of the current wind velocity numeric object.

Table 11 —Wind velocity numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{ MDC_PART_SCADA, MDC_WIND_VELOCITY}	M
Supplemental-Types	The value that denotes the sensor location. This location is a 16-bit value constructed where the high 11 bits are one of the MDC_AI_LOCATION family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_M_PER_SEC, MDC_DIM_KILO_M_PER_HR MDC_DIM_MPH	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.8 Wind direction

Table 12 summarizes the attributes of the current wind direction numeric object.

Table 12 —Wind direction numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_SCADA, MDC_WIND_DIRECTION}	M
Supplemental-Types	The value that denotes the sensor location. This location is a 16-bit value constructed where the high 11 bits are one of the MDC_AI_LOCATION family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_ANG_DEG	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.9 Electricity accumulated usage

Table 13 summarizes the attributes of the electricity accumulated usage numeric object.

Table 13—Electricity accumulated usage numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_ELECTRICITY_ACCUMULATED_USAGE}	M
Supplemental-Types	The value that denotes the appliance being monitored. This appliance is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that appliance. For example, the high 11 bits could be the value for television, and the lower 5 bits could be the value for television 1 (1 being the first television). A second supplemental type may be added to denote the location of the appliance, such as in the kitchen, and is one of the MDC_AI_LOCATION family of constants. All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_KILO_WATT_HR	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.10 Gas accumulated usage

Table 14 summarizes the attributes of the gas accumulated usage numeric object.

Table 14—Gas accumulated usage numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_GAS_ACCUMULATED_USAGE}	M
Supplemental-Types	The value that denotes the appliance being monitored. This appliance is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that appliance. For example, the high 11 bits could be the value for heater, and the lower 5 bits could be the value for heater 3 (1 being the first heater). A second supplemental type may be added to denote the location of the appliance, such as in the bedroom, and is one of the MDC_AI_LOCATION family of constants. All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_CUBIC_M	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C

Attribute name	Extended configuration	
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.11 Water accumulated usage

Table 15 summarizes the attributes of the water accumulated usage numeric object.

Table 15—Water accumulated usage numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_WATER_ACCUMULATED_USAGE}	M
Supplemental-Types	The value that denotes the appliance being monitored. This appliance is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that appliance. For example, the high 11 bits could be the value for washing machine, and the lower 5 bits could be the value 0. A second supplemental type may be added to denote the location of the appliance, such as in the utility room, and is one of the MDC_AI_LOCATION family of constants. All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_CUBIC_M	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.12 Solar accumulated generation

Table 16 summarizes the attributes of the solar accumulated generation numeric object.

Table 16—Solar accumulated generation numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_SOLAR_ACCUMULATED}	M
Supplemental-Types	See IEEE Std 11073-20601.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_KILO_WATT_HR	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C

Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.13 Electricity instantaneous usage

Table 17 summarizes the attributes of the electricity instantaneous usage numeric object.

Table 17—Electricity instantaneous usage numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_ELECTRICITY_INSTANTANEOUS_USAGE}	M
Supplemental-Types	The value that denotes the appliance being monitored. This appliance is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that appliance. For example, the high 11 bits could be the value for television, and the lower 5 bits could be the value for television 1 (1 being the first television). A second supplemental type may be added to denote the location of the appliance, such as in the kitchen, and is one of the MDC_AI_LOCATION family of constants. All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_WATT	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.14 Gas instantaneous usage

Table 18 summarizes the attributes of the gas instantaneous usage numeric object.

Table 18—Gas instantaneous usage numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_GAS_INSTANTANEOUS_USAGE}	M
Supplemental-Types	The value that denotes the appliance being monitored. This appliance is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that appliance. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for heater 3 (1 being the first heater). A second supplemental type may be added to denote the location of the appliance, such as in the bedroom, and is one of the MDC_AI_LOCATION family of constants. All of these values are from the Partition ID: MDC_PART_PHD_AI.	C

Attribute name	Extended configuration	
	Value	**Qual.**
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_CUBIC_M_PER_HR	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.15 Water instantaneous usage

Table 19 summarizes the attributes of the water instantaneous usage numeric object.

Table 19—Water instantaneous usage numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_WATER_INSTANTANEOUS_USAGE}	M
Supplemental-Types	The value that denotes the appliance being monitored. This location is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that appliance. For example, the high 11 bits could be the value for washing machine. A second supplemental type may be added to denote the location of the appliance, such as in the utility room, and is one of the MDC_AI_LOCATION family of constants. All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_L_PER_MIN	M
Attribute-Value-Map	See IEEE Std 11073-20601.	C
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.16 Solar instantaneous generation

Table 20 summarizes the attributes of the electricity instantaneous generation numeric object.

Table 20—Solar instantaneous generation numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_SOLAR_INSTANTANEOUS }	M
Supplemental-Types	See IEEE Std 11073-20601.	C

Attribute name	Extended configuration	
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_WATT	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.17 Base coordinate, Longitude and Latitude

Table 21 summarizes the attributes of the base coordinate (longitude and latitude) numeric object.

Table 21—Base coordinate, longitude and latitude, numeric object attributes

Attribute name	Extended configuration	
	Value	Qual.
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_BASE_COORD}	M
Supplemental-Types	See IEEE Std 11073-20601.	NR
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Metric-Id-List	This attribute shall identify the individual elements that are present in the compound observed value. The order of the Metric-Id-List shall correspond to the order of the elements in the compound observed value. It shall include as elements: {MDC_AI_TYPE_SENSOR_BASE_COORD_X, MDC_AI_TYPE_SENSOR_BASE_COORD_Y}.	M
Unit-Code	MDC_DIM_ANG_DEG	C
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Compound-Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Event-Context-Id	A unique value within an association shall be allocated to each observation when accompanied by an observation of base coordinate altitude, which shall be allocated the same unique value.	C

6.6.18 Base coordinate, Altitude

Table 22 summarizes the attributes of the base coordinate (altitude) numeric object.

Table 22—Base coordinate, Altitude, numeric object attributes

Attribute name	Extended configuration	
	Value	Qual.
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_BASE_COORD_Z}	M
Supplemental-Types	See IEEE Std 11073-20601.	NR
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M

Attribute name	Extended configuration	
Unit-Code	MDC_DIM_M	C
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C
Event-Context-Id	A unique value within an association shall be allocated to each observation when accompanied by an observation of base longitude and latitude, which shall be allocated the same unique value.	C

6.6.19 Base coordinate angle

Table 23 summarizes the attributes of the base coordinate orientation numeric object.

Table 23—Base coordinate angle numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_BASE_COORD_ANGLE}	M
Supplemental-Types	See IEEE Std 11073-20601.	NR
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_ANG_DEG	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.20 Relative coordinates

Table 24 summarizes the attributes of the relative coordinates numeric object.

Table 24—Relative coordinates numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_REL_COORD}	M
Supplemental-Types	The value that denotes the location or appliance to which the sensor is attached. This location is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_LOCATION or MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for wheelchair, and the lower 5 bits could be the value for wheelchair 3 (1 being the first wheelchair). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M

Attribute name	Extended configuration	
Metric-Id-List	This attribute shall identify the individual elements that are present in the compound observed value. The order of the Metric-Id-List shall correspond to the order of the elements in the compound observed value. It can include as elements: MDC_AI_TYPE_SENSOR_REL_COORD_X, MDC_AI_TYPE_SENSOR_REL_COORD_Y, MDC_AI_TYPE_SENSOR_REL_COORD_Z.	
Unit-Code	MDC_DIM_M	C
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Compound-Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.21 GPS location, Latitude and Longitude

Table 25 summarizes the attributes of the GPS location (latitude and longitude) numeric object.

Table 25 —GPS location, Latitude and Longitude, numeric object

Attribute name	Extended configuration	
	Value	**Qual**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_GPS_LOCATION}	M
Supplemental-Types	The value that denotes the location or appliance to which the sensor is attached. This location is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_LOCATION or MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for wheelchair, and the lower 5 bits could be the value for wheelchair 3 (1 being the first wheelchair). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Metric-Id-List	This attribute shall identify the individual elements that are present in the compound observed value. The order of the Metric-Id-List shall correspond to the order of the elements in the compound observed value. It shall include as elements: MDC_AI_TYPE_SENSOR_GPS_LONGITUDE, MDC_AI_TYPE_SENSOR_GPS_LATITUDE.	
Unit-Code	MDC_DIM_ANG_DEG	C
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Compound-Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Compound-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Event-Context-Id	A unique value within an association shall be allocated to each observation when accompanied by an observation of GPS altitude, which shall be allocated the same unique value.	C

6.6.22 GPS location, Altitude

Table 26 summarizes the attributes of the GPS altitude.

Table 26 —GPS location, Altitude, numeric object

Attribute name	**Extended configuration**	
	Value	**Qual**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_GPS_ALTITUDE}	M
Supplemental-Types	The value that denotes the location or appliance to which the sensor is attached. This location is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_LOCATION or MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for wheelchair, and the lower 5 bits could be the value for wheelchair 3 (1 being the first wheelchair). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_M	C
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C
Event-Context-Id	A unique value within an association shall be allocated to each observation when accompanied by an observation of GPS longitude and latitude, which shall be allocated the same unique value.	C

6.6.23 Speed

Table 27 summarizes the attributes of the speed numeric object.

Table 27 —Speed numeric object attributes

Attribute name	**Extended configuration**	
	Value	**Qual**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_SPEED	M
Supplemental-Types	The value that denotes the location or appliance to which the sensor is attached. This location is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_LOCATION or MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for wheelchair, and the lower 5 bits could be the value for wheelchair 3 (1 being the first wheelchair). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_M_PER_SEC, MDC_DIM_KILO_M_PER_HR MDC_DIM_MPH	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C

Attribute name	Extended configuration	
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.6.24 Heading

Table 28 summarizes the attributes of the heading numeric object.

Table 28 —GPS latitude numeric object attributes

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	{MDC_PART_PHD_AI, MDC_AI_TYPE_SENSOR_HEADING	M
Supplemental-Types	The value that denotes the location or appliance to which the sensor is attached. This location is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_LOCATION or MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for wheelchair, and the lower 5 bits could be the value for wheelchair 3 (1 being the first wheelchair). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated | mss-acc-manager-initiated | mss-acc-manager-initiated-immediate	M
Unit-Code	MDC_DIM_ANG_DEG	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Simple-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Basic-Nu-Observed-Value	See IEEE Std 11073-20601.	C
Nu-Observed-Value	See IEEE Std 11073-20601.	C

6.7 Real-time sample array objects

Real-time sample array objects are not required by this standard.

6.8 Enumeration objects

6.8.1 General

The independent living activity hub requires one activity data object for each supported sensor instance. Objects instantiated from the enumeration class are used to model the activity data. The nomenclature code to identify the enumeration class is MDC_MOC_VMO_METRIC_ENUM. The attributes of an activity data object are shown in Table 29. The attribute structure shown in Table 29 is common to all sensor event types. Later clauses define the precise definitions for each sensor event type.

Table 29—Activity data enumeration object attributes

Attribute name	Extended configuration	
	Value	Qual.
Handle	See IEEE Std 11073-20601.	M
Type	Any of the MDC_PART_PHD_AI | MDC_AI_TYPE_* ID's. See Annex C.	M
Supplemental-Types	The value that denotes the location of the sensor, or the appliance to which the sensor is attached. This location or appliance is a 16-bit value constructed, where the high 11 bits are one of the MDC_AI_LOCATION or MDC_AI_APPLIANCE family of constants as denoted in Annex C, and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom, and the lower 5 bits could be the value for bedroom 3 (1 being the first bedroom). All of these values are from the Partition ID: MDC_PART_PHD_AI.	C
Metric-Spec-Small	mss-avail-intermittent | mss-avail-stored-data | mss-upd-aperiodic | mss-msmt-aperiodic | mss-acc-agent-initiated.	M
Metric-Structure-Small	See IEEE Std 11073-20601.	NR
Measurement-Status	See IEEE Std 11073-20601.	NR
Metric-Id	See IEEE Std 11073-20601.	NR
Metric-Id-List	See IEEE Std 11073-20601.	NR
Metric-Id-Partition	See IEEE Std 11073-20601.	NR
Unit-Code	See IEEE Std 11073-20601.	NR
Attribute-Value-Map	See IEEE Std 11073-20601.	C
Source-Handle-Reference	The value references the handle of the identity object to which the sensor is associated if applicable.	C
Label-String	See IEEE Std 11073-20601.	O
Unit-LabelString	See IEEE Std 11073-20601.	NR
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
Relative-Time-Stamp	See IEEE Std 11073-20601.	NR
HiRes-Time-Stamp	See IEEE Std 11073-20601.	C
Measure-Active-Period	See IEEE Std 11073-20601.	NR
Enum-Observed-Value-Simple-OID	See IEEE Std 11073-20601.	NR

Attribute name	**Extended configuration**	
	Value	**Qual.**
Enum-Observed-Value-Simple-Bit-Str	This attribute contains the data for this sensor measurement. This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical “or” of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* The interpretation of this is specific to the instance type and is enumerated in the following specific sensor clauses. *Generic sensor health properties flags:* See Annex B for ASN.1 definitions. (multiple of the following flags may be set) auto-presence-received(16) (For sensors that have “heartbeat” operational status: indicates that the “heartbeat” has been seen and is ok. This flag shall be reset if Auto-Presence-Failed is set.) auto-presence-failed(17) (For sensors that have “heartbeat” operational status: indicates that the “heartbeat” has not been seen as expected. This flag shall be reset if Auto-Presence-Received is set.) low-battery(18) (Indicates the sensor is in the low battery condition. This determination is unique to the sensor.) fault(19) (Indicates that the sensor is in a fault condition and needs attention. This determination is unique to the sensor.) end-of-life(20) (Indicates that the sensor has reached end of life and needs replacement. This indication is unique to the sensor.) tamper-detected(21) (Indicates that the sensor has been tampered. This indication is unique to the sensor.) This attribute is of type BITS-32.	C
Capability-Mask-Simple	This attribute indicates whether the corresponding bit in the Enum-Observed-Value-Simple-Bit-Str attribute is supported by the Agent. The value of this attribute is implementation specific.	M
State-Flag-Simple	This attribute indicates whether the corresponding bit in the Enum-Observed-Value-Simple-Bit-Str attribute is a state (1) or event (0). The value of this attribute is implementation specific.	M
Enum-Observed-Value-Basic-Bit-Str	See IEEE Std 11073-20601.	NR
Capability-Mask-Basic	See IEEE Std 11073-20601.	NR
State-Flag-Basic	See IEEE Std 11073-20601.	NR
Enum-Observed-Value-Simple-Str	See IEEE Std 11073-20601.	C
Enum-Observed-Value	See IEEE Std 11073-20601.	NR

Attribute name	Extended configuration	
	Value	Qual.
Enum-Observed-Value-Partition	See IEEE Std 11073-20601.	NR
NOTE—See IEEE Std 11073-20601 for information on whether an attribute is static, dynamic, or observational.		

NOTE—See 6.3 for a description of the qualifiers.

6.8.2 to 6.8.16 describe the possible uses of the activity data enumeration object. Each use is an instance of the activity data enumeration class with a particular type value. The interpretation of associated values is dependent on the type value. The description of each activity data enumeration object defines all the possible states, and where appropriate, its behavior. The respective tables define the activity data sensor events generated by the agent in response to a change in sensor state.

Unless otherwise defined, a sensor shall always generate a sensor event whenever a sensor changes from the no condition detected (non-activated) state to another (activated) state. A sensor shall always generate a sensor event should the sensor state subsequently change to another (activated) state. A sensor may generate a sensor event whenever the sensor returns to the no condition detected (non-activated) state. The no condition event is a generic sensor event that has no implication as to any sensor state. This is merely a value that can be used to delimit the ending of a previous event and/or be used as the specific sensor event value when desiring to issue a generic event value.

The activity data enumeration object does not support any methods, events, or other services.

See IEEE Std 11073-20601 for descriptive explanations on the individual attributes as well as information on attribute ID and attribute type.

The event indicated by a change of the sensor event generic health properties flags of an activity data object may be represented by the term code and Reference Id of Table 30 in protocols that require an enumeration code.

Table 30 —Activity data enumeration object generic health event codes

Flag	Event	Part::code	Reference Id
auto-presence-received(16)	0->1	130::55000	MDC_AI_EVT_PRESENCE_REGAINED
auto-presence-failed(17)	0->1	130::55001	MDC_AI_EVT_PRESENCE_LOST
low-battery(18)	0->1	130::55002	MDC_AI_EVT_LOW_BATTERY
low-battery(18)	1->0	130::55003	MDC_AI_EVT_BATTERY_REPLACED
fault(19)	0->1	130::55004	MDC_AI_EVT_FAULT
fault(19)	1->0	130::55005	MDC_AI_EVT_FAULT_CLEARED
end-of-life(20)	0->1	130::55006	MDC_AI_EVT_END_OF_LIFE
tamper-detected(21)	0->1	130::55007	MDC_AI_EVT_TAMPER_DETECTED

6.8.2 Fall sensor

Table 31 is the sensor event specification of the fall sensor (see 5.3.2). It is fundamentally activated or non-activated. A fall detected sensor event shall be sent whenever a sensor determines a fall has occurred. A condition ended detected sensor event may be sent if the sensor can determine such a status. As the fall sensor is mobile, there is no requirement that its location be updated or altered.

This object is an instance of the enumeration class.

Table 31 —Activity data enumeration object—fall sensor

Attribute name	Extended configuration	
	Value	Qual.
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_FALL.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C

Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical "or" of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions of the FallSensorFlags. (Exactly one flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) fall-detected(0) stumble-detected(1) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a fall sensor data object may be represented by the term code and Reference Id of Table 32 in protocols that require an enumeration code.

Table 32—Fall sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
stumble-detected(0)	0->1	130::55020	MDC_AI_EVT_STUMBLE_DETECTED,
fall-detected(0)	0->1	130::55021	MDC_AI_EVT_FALL_DETECTED,
fall-detected(0)	1->0	130::55022	MDC_AI_EVT_FALL_RECOVERY_DETECTED

6.8.3 PERS sensor

Table 33 is the sensor event specification of the PERS sensor (see 5.3.3). A button-activated sensor event shall be sent whenever the button is pressed. A condition ended detected sensor event may be sent when the button is released. A button-activated sensor event is always treated and acted on as an urgent event regardless of whether a condition ended detected sensor event is subsequently generated. Should the PERS sensor be mobile, there is no requirement that its location be updated or altered.

This object is an instance of the enumeration class.

NOTE—The above discussion of a button usage to trigger the event is an example only. The agent implementation may use any mechanism to accomplish this function. If the switch is in a fixed location, this may may be reported.

Table 33—Activity data enumeration object—PERS sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_PERS.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	O
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
Enum-Observed-Value-	This attribute consists of two parts:	M

Simple-Bit-Str	1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical "or" of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) pers-activated(0) *Generic sensor health properties flags:* See Table 29 for definition.	

The event indicated by a change of the sensor event properties flags of a PERS sensor data object may be represented by the term code and Reference Id of Table 34 in protocols that require an enumeration code.

Table 34—PERS sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
pers-activated(0)	0->1	130::55030	MDC_AI_EVT_PERS_ACTIVATED
pers-activated(0)	1->0	130::55031	MDC_AI_EVT_PERS_RESET

6.8.4 Environmental alert sensor

Table 35 is the sensor event specification of the smoke sensor, carbon monoxide sensor, water sensor, and gas sensor (see 5.3.4). It is fundamentally activated or non-activated. A condition detected event shall be sent whenever a sensor determines the condition has occurred. A condition ended detected sensor event may be sent if the sensor can determine such a status.

This object is an instance of the enumeration class.

Table 35—Activity data enumeration object—environmental alert sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_SMOKE or MDC_AI_TYPE_SENSOR_CO or MDC_AI_TYPE_SENSOR_WATER or MDC_AI_TYPE_SENSOR_GAS.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C

Attribute name	Extended configuration	
	Value	**Qual.**
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical “or” of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one flag at most shall be set. In the case of no condition detected then all specific sensor event flags shall be reset.) condition-detected(0) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of an environmental alert sensor data object may be represented by the term code and Reference Id of Table 36 in protocols that require an enumeration code.

Table 36—Environmental alert sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
condition-detected(0)	0->1	130::55040	MDC_AI_EVT_CONDITION_DETECTED
condition-detected(0)	1->0	130::55041	MDC_AI_EVT_CONDITION_CLEARED

6.8.5 Motion sensor

Table 37 is the sensor event specification of the motion sensor (see 5.3.5). It is fundamentally activated immediate, activated delayed, non-activated, or tampered. A motion detected event shall be sent whenever a sensor determines that motion has occurred. A motion detected, delayed, and condition ended detected sensor event may be sent if the sensor can determine such a status.

This object is an instance of the enumeration class.

Table 37—Activity data enumeration object—motion sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_MOTION.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31).	M

	The field is the logical “or” of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one motion flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) motion-detected(0) motion-detected-delayed(1) *Generic sensor health properties flags:* See Table 29 for definition.	

The event indicated by a change of the sensor event properties flags of a motion sensor data object may be represented by the term code and Reference Id of Table 38 in protocols that require an enumeration code.

Table 38 —Motion sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
motion-detected(0)	0->1	130::55050	MDC_AI_EVT_MOTION_DETECTED
motion-detected-delayed(1)	0->1	130::55051	MDC_AI_EVT_MOTION_DETECTED_DELAYED
motion-detected(0)	1->0	130::55052	MDC_AI_EVT_MOTION_ENDED

6.8.6 Property exit sensor

Table 39 is the sensor event specification of the property exit sensor (see 5.3.6). It is fundamentally activated or non-activated and may detect if the door of the premises has been left open. An occupant exit detected event shall be sent whenever a sensor determines an occupant exiting event has occurred. An exit door left open and condition ended detected sensor event may be sent if the sensor can determine such a status. If the sensor is place within a building or defined area, the more general boundary exit/enter condition may be used.

An example use of this sensor would be in situations where a dwelling had occupants that suffered from Alzheimer’s. It could notify when an occupant had strayed from the premises and whether the exit door had been left open.

This object is an instance of the enumeration class.

Table 39—Activity data enumeration object—property exit sensor

Attribute name	Extended configuration	
	Value	Qual.
Handle	See IEEE Std 11073-20601	M
Type	MDC_AI_TYPE_SENSOR_PROPEXIT.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601	C

Attribute name	Extended configuration	
	Value	Qual.
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical "or" of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) occupant-exit-property(0) (one or more people have departed via exit door) exit-door-left-open(1) (exit door remains open) exit-boundary(2) (exit beyond defined perimeter) occupant-enter-property(3) (one or more people have entered via entrance door) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a property exit sensor data object may be represented by the term code and Reference Id of Table 40 in protocols that require an enumeration code.

Table 40 — Property exit sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
occupant-exit-property(0)	0->1	130::55060	MDC_AI_EVT_OCCUPANT_EXIT_PROPERTY
occupant-enter-property(3)	0->1	130::55061	MDC_AI_EVT_OCCUPANT_ENTER_PROPERTY
exit-door-left-open(1)	0->1	130::55062	MDC_AI_EVT_EXIT_DOOR_OPEN
exit-door-left-open(1)	1->0	130::55063	MDC_AI_EVT_EXIT_DOOR_CLOSED
exit-boundary(2)	0->1	130::55064	MDC_AI_EVT_EXIT_BOUNDARY
exit-boundary(2)	1->0	130::55065	MDC_AI_EVT_ENTER_BOUNDARY

6.8.7 Enuresis sensor

Table 41 is the sensor event specification of the bed-wetting (enuresis) sensor (see 5.3.7). It is fundamentally activated or non-activated. An enuresis detected event shall be sent whenever a sensor determines the condition has occurred. A condition ended detected sensor event may be sent if the sensor can determine such a status.

This object is an instance of the enumeration class.

Table 41 — Activity data enumeration object—enuresis sensor

Attribute name	Extended configuration	
	Value	Qual.
Handle	See IEEE Std 11073-20601	M
Type	MDC_AI_TYPE_SENSOR_ENURESIS.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical “or” of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) enuresis-detected(0) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of an enuresis sensor data object may be represented by the term code and Reference Id of Table 42 in protocols that require an enumeration code.

Table 42 — Enuresis sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
enuresis-detected(0)	0->1	130::55070	MDC_AI_EVT_ENURESIS_DETECTED
enuresis-detected(0)	1->0	130::55071	MDC_AI_EVT_ENURESIS_CLEARED

6.8.8 Contact closure sensor

Table 43 is the sensor events of the contact closure sensor (see 5.3.8). It is fundamentally activated or non-activated. A closure closed event and closure opened event shall be sent whenever a sensor determines the condition has occurred.

This object is an instance of the enumeration class.

Table 43 — Activity data enumeration object—contact closure sensor

Attribute name	Extended configuration	
	Value	Qual.
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_CONTACTCLOSURE.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C

Attribute name	**Extended configuration**	
	Value	**Qual.**
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical “or” of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) contact-opened(0) contact-closed(1) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a contact closure sensor data object may be represented by the term code and Reference Id of Table 44 in protocols that require an enumeration code.

Table 44—Contact closure sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
contact-opened(0)	0->1	130::55080	MDC_AI_EVT_CONTACT_OPENED
contact-closed(1)	0->1	130::55081	MDC_AI_EVT_CONTACT_CLOSED

6.8.9 Usage sensor

Table 45 is the sensor event specification of the usage sensor (see 5.3.9). An example of this sensor would be a bed/chair usage sensor. It is fundamentally activated or non-activated. It may generate further events based on violation of timing constraints based on usage or absence. A usage started event and usage ended event shall be sent whenever a sensor determines the condition has occurred. An expected use start violation, expected use stop violation, absence of use violation, and condition ended detected sensor event may be sent if the sensor can determine such a status.

This object is an instance of the enumeration class.

Table 45 —Activity data enumeration object—usage sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_USAGE.	M
Supplemental-Types	The unique value that denotes the applianceto which the sensor is connected. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical "or" of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) usage-started(0) (bed/chair in) usage-ended(1) (bed/chair out) expected-use-start-violation(2) (expected usage not started) expected-use-stop-violation(3) (usage continued beyond expected usage end) absence-violation(4) (absent for too long a period during expected usage) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a usage sensor data object may be represented by the term code and Reference Id of Table 46 in protocols that require an enumeration code.

Table 46 —Usage sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
usage-started(0)	0->1	130::55090	MDC_AI_EVT_USAGE_STARTED
usage-ended(1)	0->1	130::55091	MDC_AI_EVT_USAGE_ENDED
expected-use-start-violation(2)	0->1	130::55092	MDC_AI_EVT_USE_START_VIOLATION
expected-use-stop-violation(3)	0->1	130::55093	MDC_AI_EVT_USE_STOP_VIOLATION
absence-violation(4)	0->1	130::55094	MDC_AI_EVT_ABSENCE_VIOLATION

6.8.10 Switch sensor

Table 47 is the sensor event specification of the switch use sensor (see 5.3.10). It is fundamentally activated or non-activated. A switch ON and switch OFF event shall be sent whenever a sensor determines the condition has occurred.

This object is an instance of the enumeration class.

Table 47 —Activity data enumeration object—switch sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_SWITCH.	M
Supplemental-Types	The unique value that denotes the appliance to which the sensor is connected. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical "or" of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) switch-on(0) switch-off(1) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a switch sensor data object may be represented by the term code and Reference Id of Table 48 in protocols that require an enumeration code.

Table 48 —Switch sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
switch-on(0)	0->1	130::55100	MDC_AI_EVT_SWITCH_ON
switch-off(1)	0->1	130::55101	MDC_AI_EVT_SWITCH_OFF

6.8.11 Medication dosage sensor

Table 49 is the sensor event specification of the medication dispenser (see 5.3.11). A dosage taken event shall be sent whenever a sensor determines the condition has occurred. A dosage missed and empty dispenser condition event may be sent if the sensor can determine such a status.

This object is an instance of the enumeration class.

Table 49 —Activity data enumeration object—medication dosage sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_DOSAGE.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	O
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C

Attribute name	Extended configuration	
	Value	**Qual.**
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical "or" of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one flag at most shall be set. In the case of no condition detected then all specific sensor event flags shall be reset.) dosage-taken(0) dosage-missed(1) dispenser-empty(2) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a medication dosage sensor data object may be represented by the term code and Reference Id of Table 50 in protocols that require an enumeration code.

Table 50 —Medication dosage sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
dosage-taken(0)	0->1	130::55110	MDC_AI_EVT_DOSAGE_TAKEN
dosage-missed(1)	0->1	130::55111	MDC_AI_EVT_DOSAGE_MISSED
dispenser-empty(2)	0->1	130::55112	MDC_AI_EVT_DOSAGE_EMPTY

6.8.12 Temperature alert sensor

Table 51 is the sensor event specification of the temperature alert sensor (see 5.3.12). A high temperature detected and low temperature detected event shall be sent whenever a sensor determines the condition has occurred. A rate of change too fast and no condition detected sensor event may be sent if the sensor can determine such a status.

This object is an instance of the enumeration class.

Table 51 —Activity data enumeration object—temperature alert sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_TEMP.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C

Attribute name	Extended configuration	
	Value	**Qual.**
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical "or" of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one temperature flag or no condition detected flag at most shall be set; the rate_of_change_too_fast flag may be set at any time. In the case of no condition detected, then all specific sensor event flags shall be reset.) high-temperature-detected(0) low-temperature-detected(1) rate-of-change-too-fast(2) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a temperature alert sensor data object may be represented by the term code and Reference Id of Table 52 in protocols that require an enumeration code.

Table 52—Temperature alert sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
high-temperature-detected(0)	0->1	130::55120	MDC_AI_EVT_HIGH_TEMP_DETECTED
low-temperature-detected(1)	0->1	130::55121	MDC_AI_EVT_LOW_TEMP_DETECTED
high-temperature-detected(0) or low-temperature-detected(1)	1->0	130::55122	MDC_AI_EVT_NORMAL_TEMP_DETECTED
rate-of-change-too-fast(2)	0->1	130::55123	MDC_AI_EVT_TEMP_CHANGE_TOO_FAST

6.8.13 Humidity alert sensor

Table 53 is the sensor event specification of the humidity alert sensor (see 5.3.13). A high humidity detected and humidity returned to normal event shall be sent whenever a sensor determines the condition has occurred.

This object is an instance of the enumeration class.

Table 53—Activity data enumeration object—humidity alert sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_HUMIDITY.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C

Attribute name	Extended configuration	
	Value	**Qual.**
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical “or” of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one humidity flag or no condition detected flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) high-humidity-detected(0) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a humidity alert sensor data object may be represented by the term code and Reference Id of Table 54 in protocols that require an enumeration code.

Table 54—Humidity alert sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
high-humidity-detected(0)	0->1	130::55130	MDC_AI_EVT_HIGH_HUMIDITY_DETECTED,
high-humidity-detected(0)	1->0	130::55131	MDC_AI_EVT_NORMAL_HUMIDITY_DETECTED

6.8.14 Location

Table 55 is the specification of the location sensor (see 5.3.25). Detection of a sensor being within range and moving out of range of a specified location shall be sent whenever a sensor determines the condition has occurred.

This object is an instance of the enumeration class.

Table 55—Activity data enumeration object—location sensor

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_LOCATION.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	M
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C

Attribute name	Extended configuration	
	Value	**Qual.**
Enum-Observed-Value-Simple-Bit-Str	This attribute consists of two parts: 1) The specific sensor event properties flags are represented by bit(0) through bit(15). 2) The generic sensor health flags are represented by bit(16) through bit(31). The field is the logical "or" of these two values. Any valid setting of the specific sensor event flags can be combined with any valid setting of the generic sensor health flags. *Specific sensor event properties flags:* See Annex B for ASN.1 definitions. (Exactly one location flag or no condition detected flag at most shall be set. In the case of no condition detected, then all specific sensor event flags shall be reset.) location-detected(0) location-not-detected(1) *Generic sensor health properties flags:* See Table 29 for definition.	M

The event indicated by a change of the sensor event properties flags of a location sensor data object may be represented by the term code and Reference Id of Table 54 in protocols that require an enumeration code.

Table 56—Location sensor enumeration object event codes

Flag	Event	Part::code	Reference Id
location-detected(0)	0->1	130::55140	MDC_AI_EVT_AT_LOCATION
location-not-detected(1)	0->1	130::55141	MDC_AI_EVT_NOT_AT_LOCATION

6.8.15 Base coordinate location

Table 57 is the specification of the base coordinate location (see 5.3.25). Location can be reported as one of the location enumeration codes in the Supplemental-Types attribute, and/or a textual description as the Enum-Observed-Value-Simple-Str attribute.

One of Supplemental-Types or Enum-Observed-Value-Simple-Str must be reported.

This object is an instance of the enumeration class.

Table 57—Activity data enumeration object—base coordinate location

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_SENSOR_BASE_COORD_LOCATION.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	C
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
Enum-Observed-Value-Simple-Str	See IEEE Std 11073-20601.	C

6.8.16 Identity

Table 58 is the specification of the identity sensor (see 5.3.29). A locally unique opaque string shall be reported in the Enum-Observed-Value-Simple-Str attribute as the identity. If the identity object is present, then those sensors associated with a person or object shall have the Source-Handle-Reference set the value of the handle of the identity object.

Table 58 —Activity data enumeration object—identity

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_PERSON_IDENTITY.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	C
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
Enum-Observed-Value-Simple-Str	A locally unique opaque string	M

6.8.17 Encounter

Table 59 is the specification of the encounter sensor (see 5.3.30). A locally unique opaque string shall be reported in the Enum-Observed-Value-Simple-Str attribute as the identity of the person or object that has been encountered.

Table 59 —Activity data enumeration object—encounter

Attribute name	Extended configuration	
	Value	**Qual.**
Handle	See IEEE Std 11073-20601.	M
Type	MDC_AI_TYPE_PERSON_ENCOUNTER.	M
Supplemental-Types	The unique value that denotes the location. See Table 29 for definition.	O
Absolute-Time-Stamp	See IEEE Std 11073-20601.	C
Base-Offset-Time-Stamp	See IEEE Std 11073-20601.	C
Enum-Observed-Value-Simple-Str	A locally unique opaque string	M

6.9 PM-store objects

PM-store objects are not required by this standard.

6.10 Scanner objects

Scanner objects are not required by this standard.

6.11 Class extension objects

In this standard, no class extension objects are defined with respect to IEEE Std 11073-20601.

6.12 Independent living activity hub information model extensibility rules

The independent living activity hub domain information model of this standard may be extended by including vendor-specific metrics and attributes as required. For example, a vendor might include a location attribute value that is not included in the standard location attribute value table. Any object or attribute extensions implemented should follow the guidelines of this standard as closely as possible.

All configurations are required to be extended configurations defined by the vendor, since this standard does not define any standard configurations.

Vendor nomenclature extensions can be made in the preserved area for extension. In the aging independently partition (MDC_PART_PHD_AI), this is the space 0xF000 to 0xFFFF.

7. Independent living activity hub service model

7.1 General

The service model defines the conceptual mechanisms for data-exchange services. These services are mapped to messages that are exchanged between the agent and the manager. Protocol messages within the ISO/IEEE 11073 series of standards are defined in ASN.1. See IEEE Std 11073-20601 for a detailed description of the personal health device service model. The following sub-clauses define the specifics of object access and event reporting services for an independent living activity hub agent according to this standard.

7.2 Object access services

The object access services of IEEE Std 11073-20601 are used to access the objects defined in the domain information model of the independent living activity hub.

The following generic object access services are supported by an independent living activity hub according to this standard:

- GET service: used by the manager to retrieve the implemented attribute values of the agent MDS object. The list of independent living activity hub MDS object attributes is given in 6.5.1.
- SET service: used by the manager to set the values of the agent object attributes. There are no settable attributes defined for an independent living activity hub agent according to this standard.
- EVENT REPORT service: used by the agent to send configuration reports and measurement data to the manger. The list of event reports for the independent living activity hub agent standard is given in 6.5.3.
- ACTION service: used by the manager to invoke actions (or methods) supported by the agent. An example is Set-Time action, which is used to set a real-time clock with the absolute time at the agent.

Table 60 summarizes the object access services described in this standard.

Table 60 —Independent living activity hub object access services

Service	Subservice type name	Mode	Subservice type	Parameters	Result	Remarks
GET	<na>	<implied Confirmed>	<na>	GetArgumentSimple =(obj-handle=0), attribute-id-list<optional>	GetResultSimple=(obj-handle=0), attribute-list	Allows the manager to retrieve the value of an attribute of an object in the agent.
EVENT REPORT	MDS-Configuration-Event	Confirmed	MDC_NOTI_CONFIG	ConfigReport	ConfigReportRsp	Configuration Report to inform manager of the configuration of the agent.
	MDS-Scan-Report-Var	Confirmed	MDC_NOTI_SCAN_REPORT_VAR	ScanReportInfoVar	—	Data Report to provide dynamic data to manager for some or all of the agent’s objects in variable format.
	MDS-Scan-Report-Fixed	Confirmed	MDC_NOTI_SCAN_REPORT_FIXED	ScanReportInfoFixed	—	Data Report to provide dynamic data to manager for some or all of the agent’s objects in fixed format.

Service	Subservice type name	Mode	Subservice type	Parameters	Result	Remarks
	MDS-Scan-Report-MP-Var	Confirmed	MDC_NOTI_SCAN_REPORT_MP_VAR	ScanReportInfoMPVar	—	This is the same as MDS-Dynamic-Data-Update-Var but allows inclusion of data from multiple people.
	MDS-Scan-Report-MP-Fixed	Confirmed	MDC_NOTI_SCAN_REPORT_MP_FIXED	ScanReportInfoMPFixed	—	This is the same as MDS-Dynamic-Data-Update-Fixed but allows inclusion of data from multiple people.
ACTION	Set-Time	Confirmed	MDC_ACT_SET_TIME	SetTimeInvoke	—	Manager method to invoke the agent to set time to requested value.
	Set-Base-Offset-Time	Confirmed	MDC_ACT_SET_BO_TIME	SetBOTimeInvoke	—	Manager method to invoke the agent to set time to requested value.
	MDS-Data-Request	Confirmed	MDC_ACT_ DATA_REQUEST	DataRequest	DataResponse	Allows the manager to request measurement data from the agent.

7.3 Object access event report services

The event report service (see Table 60) is used by the agent to report its information (e.g., measurements). Event reports in this standard are a property of the MDS object only. The event reports used in this standard are defined in IEEE Std 11073-20601.

The following conditions apply for an independent living activity hub agent according to this standard:

- Event reports shall be used in confirmed mode.
- Agent-initiated mode shall be supported for measurement data transmission if an agent contains such objects.
- Manager-initiated mode shall be supported for the retrieval of measurement data if an agent contains such objects.

An independent living activity hub agent, which is designed to operate in an environment where data may be collected from multiple people, may use one of the multiple-person event report styles to transmit all the data from each person in a single event report. If this functionality is not required, the agent may use the single-person event report styles, which have reduced overhead.

A manager shall support both single-person and multiple-person event reports. An independent living activity hub agent may support either one or both single-person and multiple-person event reports. The formats for single- and multiple-person event reports are described in IEEE Std 11073-20601.

8. Independent living activity hub communication model

8.1 Overview

The following sub-clauses describe the general communication model and procedures of the independent living activity hub agent as defined in IEEE Std 11073-20601. Therefore, the respective parts of IEEE Std 11073-20601 are not reproduced; rather the specific choices and restrictions with respect to optional elements (e.g., objects, attributes, and actions) and specific extensions (e.g., nomenclature terms) are specified.

For an illustrative overview of the various message transactions during a typical measurement session, see the sequence diagram for the example use case in Annex D and the corresponding protocol data unit (PDU) examples in Annex E.

8.2 Communication characteristics

In this section, limits on the size of an application protocol data unit (APDU) to be transmitted or received by an independent living activity hub agent are defined. Small limits allow for simple implementations in terms of low cost and complexity.

For an independent living activity hub agent implementing no other device specialization except this standard, the maximum size of an APDU sent shall be not larger than N_{tx}. For this standard, it is N_{tx} = 5120 octets. This allows for 100 sensors using the objects in this standard. An agent according to this definition shall be capable of receiving an APDU up to the size of at least N_{rx}. For this standard, it is N_{rx} = 224 octets.

For an independent living activity hub agent implementing multiple functions according to multiple device specializations, the maximum size of an APDU sent shall not be larger than $N_{tx,i}$ if this agent implements another device specialization *i* with $N_{tx,i} > N_{tx}$. Otherwise, the maximum size of an APDU sent shall not be larger than N_{tx}. An agent according to this definition shall be capable of receiving an APDU up to the size of at least $N_{rx,i}$ octets if this agent implements another device specialization *i* with $N_{rx,i} > N_{tx}$. Otherwise, the agent shall be capable of receiving an APDU up to the size of at least N_{rx}.

In case the APDU size limit does not allow for the inclusion of a certain amount of multiple pending measurements at the agent, they shall be sent using multiple event reports. See 8.5.3 for the maximum number of measurements allowed for inclusion in a single event report.

8.3 Association procedure

8.3.1 General

Unless otherwise stated, the association procedure for an independent living activity hub agent and manager according to this standard shall be pursued as specified in IEEE Std 11073-20601.

Table 61—Valid combinations of protocol and nomenclature version

Protocol version	Bit value	Corresponding nomenclature version	Bit value
1	0x80000000	1	0x80000000
2	0x40000000	1	0x80000000
3	0x20000000	2	0x40000000
4	0x10000000	3	0x20000000

Table 61 lists the valid combinations of protocol version and nomenclature version. In the association procedure, an agent indicating support for a specific protocol version shall indicate support for the corresponding nomenclature version. In the association procedure, a manager selecting a specific protocol version shall select the corresponding nomenclature version.

Support for multiple protocol versions is indicated by selecting the respective bit values for protocol and nomenclature. For example, if the agent supports protocol-version2, protocol-version3 and protocol-version4, it shall use protocol version bits 0x70000000 and nomenclature-version bits 0xE0000000.

8.3.2 Agent procedure—association request

The association request sent by the agent to the manager contains the following data fields:

- The version of the association procedure used by the agent shall be set to assoc-version1 (i.e., assoc-version = 0x80000000).
- The DataProtoList structure element of the data protocol identifier shall be set to data-proto-id-20601 (i.e., data-proto-id = 0x5079).

- The data-proto-info field shall contain a PhdAssociationInformation structure that shall contain the following parameter values:
  1) The version of the data exchange protocol shall be set to protocol-version4 (i.e., protocol-version = 0x10000000). Support for any other version may be indicated by setting additional bits. When protocols lower than protocol-version4 are used, the agent shall use only features in that protocol.
  2) At least the MDER shall be supported (i.e., encoding-rules = 0x8000).
  3) The protocol version bits and nomenclature version bits shall consist of valid combinations of bits as defined in Table 61.
  4) The field functional-units may have the test association bits set but shall not have any other bits set.
  5) The field system-type shall be set to sys-type-agent (i.e., system-type = 0x00800000).
  6) The system-id field shall be set to the value of the System-Id attribute of the MDS object of the agent. The manager may use this field to determine the identity of the independent living activity hub with which it is associating and, optionally, to implement a simple access restriction policy.
  7) The dev-config-id field shall be set to the value of the Dev-Configuration-Id attribute of the MDS object of the agent. The independent living activity hub agent does not support standard configurations. Only extended configuration IDs shall be specified here. The extended configurations range from 0x4000–0x7FFF.
  8) If the agent supports only the independent living activity hub specialization, then the field indicating the data request modes (data-req-mode-capab) supported by the agent shall be set based on the capabilities of the agent.
  9) If the agent supports agent-initiated measurement transfer, then data-req-mode-capab shall have the data-req-supp-init-agent bit set, and data-req-init-agent-count shall be set to 1.
  10) If the agent supports manager-initiated measurement transfer, then data-req-mode-capab shall have the data-req-supp-stop, data-req-supp-scope-handle, data-req-supp-mode-single-rsp, and data-req-supp-mode-time-no-limit bits set appropriately. The data-req-init-manager-count shall be set to the maximum number of concurrent manager-initiated flows supported by the agent.

8.3.3 Manager procedure—association response

The association response message sent by the manager contains following data fields:

- The *result* field shall be set to an appropriate response from those defined in IEEE Std 11073-20601. For example, if all other conditions of the association protocol are satisfied, accepted is returned when the manager recognizes the *dev-config-id* of the agent and accepted-unknown-config otherwise.
- In the DataProtoList structure element, the data protocol identifier shall be set to data-proto-id-20601 (i.e., *data-proto-id* = 0x5079).
- The *data-proto-info* field shall be filled in with a PhdAssociationInformation structure that shall contain the following parameter values:
  1) The manager following this specialization shall support protocol-version4. The manager may support additional protocol versions and select them if the agent offers them. When protocols lower than protocol-version4 are used, the manager shall use only features in that protocol.
  2) The manager shall respond with a single selected encoding rule that is supported by both agent and manager. The manager shall support at least the MDER.
  3) The manager shall select a valid combination of protocol version and nomenclature version as defined in Table 61.
  4) The field functional-units shall have all bits reset except for those relating to a test association.
  5) The field system-type shall be set to sys-type-manager (i.e., system-type = 0x80000000).
  6) The system-id field shall contain the unique system ID of the manager device, which shall be a valid EUI-64 type identifier.
  7) The field dev-config-id shall be manager-config-response (0).
  8) The field data-req-mode-capab shall be 0.

9) The fields data-req-init-*-count shall be 0.

8.4 Configuring procedure

8.4.1 General

The agent enters the Configuring state if it receives an association response of accepted-unknown-config. In this case, the configuration procedure as specified in IEEE Std 11073-20601 shall be followed. Subclause 8.4.2 specifies the configuration notification and response messages for an independent living activity hub agent with an example extended configuration ID 0x4000. Normally, a manager would not already know the extended configuration unless it had previously associated with this agent. For the purposes of this example, it does not.

8.4.2 Independent living activity hub—extended configuration

8.4.2.1 Agent procedure

The agent performs the configuration procedure using a “Remote Operation Invoke | Confirmed Event Report” message with an MDC_NOTI_CONFIG event to send its configuration to the manager (see IEEE Std 11073-20601). The ConfigReport structure is used for the *event-info* field (see Table 3). An example configuration notification message is shown in E.3.2.2 for an independent living activity hub agent with an extended configuration ID 0x4000.

8.4.2.2 Manager procedure

The manager shall respond to a configuration notification message using a “Remote Operation Response | Confirmed Event Report” data message with an MDC_NOTI_CONFIG event using the ConfigReportRsp structure for the event-info field (see Table 3). An example configuration notification response message corresponding to the configuration notification request message in 8.4.2.1 can be seen in E.3.2.3.

8.5 Operating procedure

8.5.1 General

Measurement data and status information are communicated from the independent living activity hub agent during the Operating state. If not stated otherwise, the operating procedure for an independent living activity hub agent of this standard shall be as specified in IEEE Std 11073-20601.

8.5.2 GET an independent living activity hub’s MDS attributes

See Table 4 for a summary of the GET service.

If the manager leaves the attribute-id-list field in the roiv-cmip-get service message empty, the independent living activity hub agent shall respond with a rors-cmip-get service message, in which the attribute-list contains a list with the values of all implemented attributes of the MDS object.

If the manager requests specific MDS object attributes, indicated by the elements in attribute-id-list, and the agent supports this capability, then the independent living activity hub agent shall respond with a rors-cmip-get service message in which the attribute-list contains a list of the values of the requested attributes of the MDS object that are implemented. It is not required for an independent living activity hub agent to support this capability. If this capability is not implemented, the independent living activity hub agent shall respond with a “Remote Operation Error Result” service message (see IEEE Std 11073-20601) with the error-value field set to no-such-action (9).

8.5.3 Measurement data transmission

See Table 3 for a summary of the event report services available for agent-initiated measurement data transfer of activity data sensor events, and Table 2 for manager-initiated retrieval of data.

Measurement data transfer for an independent living activity hub agent of this standard may be initiated by either the agent or the manager (see agent- and manager-initiated measurement data transmission in IEEE Std 11073-20601).

Agents compliant to the independent living activity hub specialization would be expected to be constantly or frequently associated with a manager so that data is sent immediately. However, communication issues may prevent transmission for a period of time. Upon reconnection, temporarily stored measurements should be sent to the manager using agent-initiated transfer to maintain the flow of measurements. Some types of sensors may collect a large number of measurements during periods when disconnected. It is permissible for agents in this specialization to store more than the 25 pending measurements as defined in IEEE 11073-20601.

To limit the amount of data being transported within an APDU, the independent living activity hub agent shall not include more than 25 temporarily stored measurements in a single event report. If more than 25 pending measurements are available for transmission, they shall be sent using multiple event reports. If multiple measurements are available, up to 25 measurements should be transmitted within a single event report. Alternatively, they may be transmitted using a single event report for each measurement. However, the former strategy is recommended to reduce overall message size and power consumption.

An independent living activity hub agent always has an extended configuration and may use either fixed or variable format data update messages for transmitting measurement data.

8.6 Time synchronization

Time synchronization between an independent living activity hub agent and a manager may be used to coordinate the clocks used when reporting physiological events. Note that the mechanism for synchronizing an agent to a manager is outside the scope of this standard. If time synchronization is used, then this shall be reported in the Mds-Time-Info attribute of the MDS object.

9. Test associations

An independent living activity hub may implement a wide range of behaviors in a test association that enable a manufacturer to test features of a product in a comprehensive manner. It is also possible for an independent living activity hub not to support test associations at all. This clause defines a simple behavior that simulates the generation of a measurement in the context of an extended device configuration.

9.1 Behavior with standard configuration

As the independent living activity hub does not support a standard configuration, this standard does not define a test association that uses a standard configuration.

9.2 Behavior with extended configurations

To facilitate automated standardized test processes, an independent living activity hub that presents the extended configuration ID of 0x7FFF and enters into a test association should be able to simulate the arrival of measurement data from the device sensors. It should not be necessary for an operator to stimulate the sensors for the measurement data to be generated.

After the agent enters the Operating state, it simulates the reception of a sensor event from the sensor representing a temperature alert sensor event of a high temperature detected. To the extent possible, this measurement is seen only by those components of the agent that understand the test association. When the event is propagated into an activity data enumeration object, the test-data bit of the measurement-status attribute shall be set if the measurement-status attribute is supported. An agent is not required to use the measurement-status attribute if it would not normally do so outside of a test association.

The agent should send the events reports for all simulated measures within 30 s of entering the operating state. The test association is terminated in a manner consistent with the agent’s normal behavior for terminating an association.

10. Conformance

10.1 Applicability

This standard shall be used in conjunction with IEEE Std 11073-20601.

An implementation or a system can conform to the following elements of this standard:

- Domain information model class hierarchy and object definitions (object attributes, notifications, methods, and data type definitions)
- Nomenclature code values
- Protocol and service models
- Communication service model (association and configuration)

10.2 Conformance specification

This standard offers levels of conformance with respect to strict adherence to the standard device and the use of extensions for:

- Information model of a specific device
- Use of attributes, value ranges, and access methods

A vendor shall specify the level of conformance for an implementation based on this standard and provide details of the way in which the definitions of this standard and any extensions are applied.

Specifications shall be provided in the form of a set of implementation conformance statements (ICSs) as detailed in 10.4.

This standard is used in conjunction with IEEE Std 11073-20601; the ICSs should be created for this standard first. The ICS created for IEEE Std 11073-20601 may refer to the ICS for this standard where applicable.

10.3 Levels of conformance

10.3.1 General

The following subclauses define levels of conformnce

10.3.2 Conformance level 1: Base conformance

The application uses elements of the information, service, communication models (object hierarchy, actions, event reports, and data type definitions), and the nomenclature scheme defined in IEEE Std 11073-20601 and IEEE 11073-104zz documents. All mandatory features defined in the object definition tables and in the ICS tables are implemented. Furthermore, any conditional, recommended, or optional features that are implemented shall follow the requirements in IEEE Std 11073-20601 and IEEE 11073-104zz documents.

10.3.3 Conformance level 2: Extended nomenclature (ASN.1 and/or IEEE 11073-10101)

Conformance level 2 meets conformance level 1 but also uses or adds extensions in at least one of the information, service, communication, or nomenclature models. These extensions shall conform to the structures of ASN.1 and/or nomenclature codes from within the IEEE 11073-10101 framework (0xF000 – 0xFFFF). These extensions should be defined in ICS tables pointing toward their reference.

10.4 Implementation conformance statements (ICSs)

10.4.1 General format

The ICSs are provided as an overall conformance statement document that comprises a set of tables in the form given by the templates in the following clauses.

Each ICS table has the following columns:

Index	Feature	Reference	Req/Status	Support	Comment

The table column headings have the following meaning:

- **Index:** an identifier (e.g., a tag) of a specific feature.
- **Feature:** briefly describes the characteristic for which a conformance statement is being made.
- **Reference:** to the clause/paragraph within this document or an external source for the definition of the feature (may be empty).
- **Req/Status:** specifies the conformance requirement (e.g., mandatory or recommended)—in some cases, this standard does not specify conformance requirements but requests the status of a particular feature be provided.
- **Support:** specifies the presence or absence of a feature and any description of the characteristics of the feature in the implementation. This column is to be filled out by the implementer.
- **Comment:** contains any additional information on the feature. This column is to be filled out by the implementer.

Subclauses 10.4.2 to 10.4.6 specify the format of the specific ICS tables.

10.4.2 General ICS

The general ICS specifies the versions/revisions that are supported by the implementation and high-level system behavior.

Table 62 shows general ICSs.

Table 62 —IEEE 11073-10471 general ICSs table

Index[a]	Feature	Reference	Req./Status	Support	Comment
GEN 11073-10471-1	Implementation Description	—	Identification of the device/application. Description of functionality.		
GEN 11073-10471-2	Standards followed and their revisions	(standard documents)	(set of existing revisions)	(set of supported revision)	
GEN 11073-10471-3	Nomenclature document used and revision	(standard documents)	(set of existing revisions)	(set of supported revisions)	
GEN 11073-10471-4	Conformance Adherence - Level 1 -	See 10.3.2	Base conformance declaration that device meets the following IEEE 11073-10471 conformance requirements: a) All mandatory requirements shall be implemented. b) If implemented, conditional, recommended, and	Yes/No (No is not expected as No implies that the implementation is non-conformant)	

Index[a]	Feature	Reference	Req./Status	Support	Comment
			optional requirements shall conform to standard.		
GEN 11073-10471-5	Conformance Adherence - Level 2 -	See 10.3.3	In addition to GEN 11073-10471-3, if the device implements extensions and/or additions, they shall conform to nomenclature codes from ASN.1 and/or 10101 framework. These extensions should also be defined in ICS tables pointing toward their reference.	Yes/No	
GEN 11073-10471-6	Object Containment Tree	See 6.3	Provide Object Containment Diagram showing relations between object instances used by the application. A conforming implementation uses only object relations as defined in the DIM.		
GEN 11073-10471-7	Nomenclature document used and revision	(standard documents)	(set of existing revisions)	(set of supported revision)	
GEN 11073-10471-8	Data Structure Encoding	—	—	description of encoding method(s) for ASN.1 data structures	
GEN 11073-10471-9	Use of Private Objects	—	Does the implementation use objects that are not defined in the DIM?	Yes/No [If yes: explain in Table 63]	
GEN 11073-10471-10	Use of Private Nomenclature Extensions	—	Does the implementation use private extensions to the nomenclature (i.e., 0xF000-0xFFFF codes from IEEE 11073-10101)? Private nomenclature extensions are *only* allowed if the standard nomenclature does not include the specific terms required by the application.	Yes/No [If yes: explain in the Table 66]	
GEN 11073-10471-11	11073-20601 Conformance		Provide the conformance report required by IEEE Std 11073-20601.		

[a]The prefix GEN11073-10471- is used for the index in the general ICSs table.

10.4.3 DIM MOC implementation conformance statement

The DIM managed object class (MOC) ICS defines which objects are implemented. Information on each object shall be provided as a separate row in the template of Table 63.

Table 63—Template for DIM MOC ICS table

Index	Feature	Reference	Req./Status	Support	Comment
MOC-n	Object description	Reference to the clause in the standard or other location where the object is defined.	Implemented	Specify restrictions (e.g., maximum number of supported instances)	

The n in the Index column should be the object handle for implementations that have predefined objects. Otherwise, the Index column shall simply be a unique number (1..m).

All private objects shall be specified and include either a reference to the definition for the object, or, where no publicly available reference is available, the definition of the object should be appended to the conformance statement.

The Support column shall specify whether the attribute is implemented; for extension attributes, whether the attribute value is static or dynamic; any value ranges; restrictions on attribute access or availability; and any other information.

An object containment diagram (class instance diagram) should be provided as part of the DIM MOC ICS.

10.4.4 MOC attribute ICS

For each supported object as defined in the DIM MOC ICS, a MOC attribute ICS has to be provided that defines which attributes are used/supported by the implementation, including any inherited attributes. Table 64 is a template only.

Table 64—Template for MOC attribute ICS table

Index	Feature	Reference	Req./Status	Support	Comment
ATTR-n-x	Attribute Name. Extended attributes shall include the attribute ID also.	Fill in the reference to the ASN.1 structure if the attribute is not defined in this standard.	M = Mandatory / C = Conditional / R = Recommended / O = Optional (as per definition in Attribute Definition Tables)	Implemented? Yes/No Static/Dynamic Specify restrictions (e.g., value ranges). Describe how attribute is accessed (e.g., Get, Set, sent in config event report, or sent in a data event report). Describe any specific restrictions.	

All private attributes shall be specified and include reference to the definition for the attribute. Where no publicly available reference is available, the definition of the attribute should be appended to the conformance statement.

The Support column shall specify whether the attribute is implemented; for extension attributes, whether the attribute value is static or dynamic; any value ranges; restrictions on attribute access or availability; and any other information.

The n in the Index column refers to the ID of the managed object for which the table is supplied (i.e., the index of the managed object as specified in the MOC ICS). There is one separate table for each supported managed object.

The x in the Index column is a unique serial number (1..m).

NOTE—The attribute definition tables in the standard define a minimum mandatory set of attributes for each object.

10.4.5 MOC notification ICS

The MOC notification ICS specifies all implemented notifications (typically in the form of the event report service) that are emitted by the agent. Table 65 provides a template for use. One table has to be provided for each object that supports special object notifications. One row of the table shall be used for each notification.

Table 65 —Template for MOC notification ICS table

Index	Feature	Reference	Req./Status	Support	Comment
NOTI-n-x	Notification Name and Notification ID	Reference to the clause in the standard or other location where the event is defined.		The Support column shall specify how the notification is sent and any restrictions.	

The n in the Index column refers to the ID of the managed object for which the table is supplied (i.e., the index of the managed object as specified in the POC ICS). There is one separate table for each managed object that supports specific object notifications (i.e., events).

The x in the Index column is a unique serial number (1..m).

All private notifications shall be specified and include reference to the definition for the notification. Where no publicly available reference is available, the definition of the notification should be appended to the conformance statement.

10.4.6 MOC ICS

The MOC nomenclature ICS specifies all nonstandard nomenclature codes that are utilized by the agent. Table 66 provides a template for use. One row of the table is to be used for each nomenclature element.

Table 66 —Template for MOC nomenclature ICS table

Index	Feature	Reference	Req./Status	Support	Comment
NOME-n	Nomenclature Name and Nomenclature value	Reference to the clause in the standard or other location where the nomenclature is defined or used.		Describe how the nomenclature is used. Describe any specific restrictions.	

The n in the Index column is a unique serial number (1..m).

Annex A

(informative)

Bibliography

Bibliographical references are resources that provide additional or helpful material but do not need to be understood or used to implement this standard. Reference to these resources is made for informational use only. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

[B1] IEC 60601-1:2005, Ed. 3, Medical electrical equipment-Part 1: General requirements for basic safety and essential performance.

[B2] IEC 60601-2, Medical electrical equipment-Part 2: Particular requirements for the basic safety and essential performance for specific device. (See the entire series of standards, Part 2-1 through Part 2-51.)

[B3] IEC 62304, Medical device software-Software life-cycle processes.

[B4] IEC 80001-1:2010, Application of risk management for IT-networks incorporating medical devices-Part 1: Roles, responsibilities, and activities.

[B5] IEEE Std 11073-10201, Health informatics—Point-of-care medical device communication—Part 10201: Domain information model.

[B6] ISO 9919 (2005), Medical electrical equipment – Particular requirements for the basic safety and essential performance of pulse oximeter equipment for medical use.

[B7] ISO 14971:2007, Medical devices-Application of risk management to medical devices.

[B8] ISO/IEEE 11073:10101-2020, Health informatics—Point-of-care medical device communication—Part 10101: Nomenclature.[14]

[B9] ISO/IEEE 11073:20101, Health informatics—Point-of-care medical device communication—Part 20101: Application profile—Base standard.

[B10] ITU-T Rec. X.680-2002, Information technology—Abstract Syntax Notation One (ASN.1): Specification of basic notation.[15]

[14] The IEEE standards or products referred to in this clause are trademarks of the Institute of Electrical and Electronics Engineers, Inc.

[15] ISO/IEEE publications are available from the ISO Central Secretariat, (http://www.iso.ch/). ISO/IEEE publications are also available in the United States from the Institute of Electrical and Electronics Engineers (http://standards.ieee.org/).

Annex B

(normative)

Any additional ASN.1 definitions

```
-- All unassigned "SensorHealthFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Zero or more flags may be set.
SensorHealthFlags ::= BITS-32 {   -- this field is used in the activity data events
                                  -- to report sensor health
             auto-presence-received(16),    -- For sensors that have “heartbeat” operational status:
                                            -- indicates that the “heartbeat” has been seen and is ok.
                                            -- This flag shall not be set if auto-presence-failed is set.
             auto-presence-failed(17),   -- For sensors that have “heartbeat” operational status:
                                         -- indicates that the “heartbeat” has not been seen as expected.
                                         -- This flag shall not be set if auto-presence-received is set.
             low-battery(18),     -- Indicates the sensor is in the low battery condition.
                                  -- This determination is unique to the sensor.
             fault(19),           -- Indicates that the sensor is in a fault condition and needs attention.
                                  -- This determination is unique to the sensor.
             end-of-life(20)      -- Indicates that the sensor has reached end of life.
                                  -- This indication is unique to the sensor.
             tamper-detected(21) -- indicates that a tamper event has been detected.
}

-- All unassigned "FallSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
FallSensorFlags::= BITS-32 {      -- this field is used in the fall sensor activity data events
             fall-detected(0      -- indicates that a fall has been detected
}

-- All unassigned "PersSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
PersSensorFlags::= BITS-32 {      -- this field is used in the PERS sensor activity data events
             pers-activated(0) -- indicates that a PERS event has been detected
}

-- All unassigned "EnvironmentalSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.

EnvironmentalSensorFlags::= BITS-32 {-- this field is used in the environmental alert sensor activity data events
             condition-detected(0) -- indicates that an environmental event has been detected
}

-- All unassigned "MotionSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one motion flag can be set at a time.
-- The tamper flag may be additionally set at any time.
MotionSensorFlags::= BITS-32 {  -- this field is used in the motion sensor activity data events
             motion-detected(0),            -- indicates that a motion event has been detected
             motion-detected-delayed(1)     -- indicates that a motion with delay event has been
                                            -- detected
}
-- All unassigned "PropertyExitSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
PropertyExitSensorFlags::= BITS-32 {-- this field is used in the property exit sensor activity data events
                    occupant-exit-property(0), -- indicates that an occupant exit event has been detected
                    exit-door-left-open(1), -- indicates that an exit door left open event has been detected
                    exit-boundary(2), -- indicates that there has been exit beyond defined perimeter
                    occupant-enter-property(3) -- indicates that an occupant entrance event has been detected
}

-- All unassigned "EnuresisSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
EnuresisSensorFlags::= BITS-32 {-- this field is used in the enuresis sensor activity data events
                    enuresis-detected(0) -- indicates that an enuresis event has been detected
}

-- All unassigned "ContactClosureSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
ContactClosureSensorFlags::= BITS-32 {-- this field is used in the door use sensor activity data events
                    contact-opened(0),    -- indicates that a door open event has been detected
                    contact-closed(1)     -- indicates that a door close event has been detected
}

-- All unassigned "UsageSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
UsageSensorFlags::= BITS-32 {-- this field is used in the usage sensor (bed/chair) activity data events
                    usage-started(0),              -- indicates that a usage started event has been detected
                    usage-ended(1),                -- indicates that a usage ended event has been detected
                    expected-use-start-violation(2), -- indicates that an expected use start violation event
                                                   -- has been detected
                    expected-use-stop-violation(3), -- indicates that an expected use stop violation event
                                                   -- has been detected
                    absence-violation(4)           -- indicates that an absence violation event has been detected
}

-- All unassigned "SwitchSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
SwitchSensorFlags::= BITS-32 {   -- this field is used in the switch sensor activity data events
                    switch-on(0),      -- indicates that a switch on event has been detected
                    switch-off(1)      -- indicates that a switch off event has been detected
}

-- All unassigned "MedDosageSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
MedDosageSensorFlags::= BITS-32 {-- this field is used in the dosage sensor activity data events
                    dosage-taken(0),      -- indicates the med dosage was taken
                    dosage-missed(1)      -- indicates the med dosage was not taken when expected
                    dispenser-empty(2) ) -- indicates the med dispenser is empty
}

-- All unassigned "TemperatureSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag of high-temperature-detected and low-temperature-detected can be set at a time.
```

```
TemperatureSensorFlags::= BITS-32 {-- this field is used in the temperature alert sensor activity data events
                        high-temperature-detected(0), -- indicates that a high temperature event has been detected
                        low-temperature-detected(1),  -- indicates that a low temperature event has been detected
                        rate-of-change-too-fast(2)    -- indicates that a rate of change too fast event
                                                      -- has been detected
}

-- All unassigned "HumiditySensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
HumiditySensorFlags::= BITS-32 {-- this field is used in the humidity alert sensor activity data events
                        high-humidity-detected(0),     -- indicates that a high humidity event has been detected
}

-- All unassigned "LocationSensorFlags" bit values are reserved for future expansion and
-- shall be reset.
-- Only one flag can be set at a time.
LocationSensorFlags::= BITS-32 {-- this field is used in the location sensor activity data events
                        location-detected(0),         -- indicates that a location proximityevent has been detected
                        location-not-detected(1),     -- indicates that a location proximityevent has been detected
}
```

Annex C

(normative)

Allocation of identifiers

This annex contains the nomenclature codes used in this document and not found in IEEE Std 11073-20601. For those not contained in this annex, the normative definition is found in IEEE Std 11073-20601.

The format used here follows that of IEEE 11073-10101.

```
#define MDC_PART_OBJ                          1      /* Object Infrastr.             */
#define MDC_PART_SCADA                        2      /* SCADA (Physio IDs)           */
#define MDC_PART_EVT                          3      /* Event                        */
#define MDC_PART_DIM                          4      /* Dimension                    */
#define MDC_PART_SITES                        7      /* [Body] Site                  */
#define MDC_PART_INFRA                        8      /* Infrastructure               */
#define MDC_PART_PHD_DM                       128    /* PHD Disease Management       */
#define MDC_PART_PHD_HF                       129    /* PHD Health Fitness           */
#define MDC_PART_PHD_AI                       130    /* PHD Aging                    */

/*****************************************************************************
* All of the following are from NomPartition (MDC_PART_PHD_ INFRA)          *
*****************************************************************************/
#define MDC_DEV_SPEC_PROFILE_AI_ACTIVITY_HUB                        4167  /*       */

#define MDC_DEV_SUB_SPEC_PROFILE_FALL_SENSOR                        4213  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_PERS_SENSOR                        4214  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_SMOKE_SENSOR                       4215  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_CO_SENSOR                          4216  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_WATER_SENSOR                       4217  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_GAS_SENSOR                         4218  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_MOTION_SENSOR                      4219  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_PROPEXIT_SENSOR                    4220  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_ENURESIS_SENSOR                    4221  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_CONTACTCLOSURE_SENSOR              4222  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_USAGE_SENSOR                       4223  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_SWITCH_SENSOR                      4224  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_DOSAGE_SENSOR                      4225  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_TEMP_SENSOR                        4226  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_HUMIDITY_SENSOR                    4227  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_LOCATION_SENSOR                    4228  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_AMBIENT_TEMP_SENSOR                4229  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_AMBIENT_HUMIDITY_SENSOR            4230  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_UTILITY_USAGE_SENSOR               4231  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_INSTANTANEOUS_UTILITY_USAGE_SENSOR        4232
#define MDC_DEV_SUB_SPEC_PROFILE_GPS_LOCATION_SENSOR                4233  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_SPEED_SENSOR                       4234  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_ENCOUNTER                          4235  /*       */
#define MDC_DEV_SUB_SPEC_PROFILE_WIND_SPEED                         4236  /*       */

/*****************************************************************************
* All of the following are from NomPartition (MDC_PART_PHD_AI)              *
*****************************************************************************/
#define MDC_AI_TYPE_SENSOR_FALL                                     1     /*       */
#define MDC_AI_TYPE_SENSOR_PERS                                     2     /*       */
#define MDC_AI_TYPE_SENSOR_SMOKE                                    3     /*       */
#define MDC_AI_TYPE_SENSOR_CO                                       4     /*       */
```

```
#define MDC_AI_TYPE_SENSOR_WATER                              5       /*          */
#define MDC_AI_TYPE_SENSOR_GAS                                6       /*          */
#define MDC_AI_TYPE_SENSOR_MOTION                             7       /*          */
#define MDC_AI_TYPE_SENSOR_PROPEXIT                           8       /*          */
#define MDC_AI_TYPE_SENSOR_ENURESIS                           9       /*          */
#define MDC_AI_TYPE_SENSOR_CONTACTCLOSURE                     10      /*          */
#define MDC_AI_TYPE_SENSOR_USAGE                              11      /*          */
#define MDC_AI_TYPE_SENSOR_SWITCH                             12      /*          */
#define MDC_AI_TYPE_SENSOR_DOSAGE                             13      /*          */
#define MDC_AI_TYPE_SENSOR_TEMP                               14      /*          */
#define MDC_AI_TYPE_SENSOR_HUMIDITY                           15      /*          */
#define MDC_AI_TYPE_ELECTRICITY_ACCUMULATED_USAGE             24      /*          */
#define MDC_AI_TYPE_GAS_ACCUMULATED_USAGE                     28      /*          */
#define MDC_AI_TYPE_WATER_ACCUMULATED_USAGE                   32      /*          */
#define MDC_AI_TYPE_SOLAR_ACCUMULATED                         36      /*          */
#define MDC_AI_TYPE_ELECTRICITY_INSTANTANEOUS_USAGE           40      /*          */
#define MDC_AI_TYPE_GAS_INSTANTANEOUS_USAGE                   44      /*          */
#define MDC_AI_TYPE_WATER_INSTANTANEOUS_USAGE                 48      /*          */
#define MDC_AI_TYPE_SOLAR_INSTANTANEOUS                       52      /*          */
#define MDC_AI_TYPE_PERSON_IDENTITY                           56      /*          */
#define MDC_AI_TYPE_SENSOR_LOCATION                           60      /*          */
#define MDC_AI_TYPE_BASE_COORD                                64      /*          */
#define MDC_AI_TYPE_BASE_COORD_X                              68      /*          */
#define MDC_AI_TYPE_BASE_COORD_Y                              72      /*          */
#define MDC_AI_TYPE_BASE_COORD_Z                              76      /*          */
#define MDC_AI_TYPE_BASE_COORD_ANGLE                          80      /*          */
#define MDC_AI_TYPE_BASE_LOCATION                             84      /*          */
#define MDC_AI_TYPE_SENSOR_REL_COORD                          88      /*          */
#define MDC_AI_TYPE_SENSOR_REL_COORD_X                        92      /*          */
#define MDC_AI_TYPE_SENSOR_REL_COORD_Y                        96      /*          */
#define MDC_AI_TYPE_SENSOR_REL_COORD_Z                        100     /*          */
#define MDC_AI_TYPE_SENSOR_GPS_LOCATION                       104     /*          */
#define MDC_AI_TYPE_SENSOR_GPS_LATITUDE                       108     /*          */
#define MDC_AI_TYPE_SENSOR_GPS_LONGITUDE                      112     /*          */
#define MDC_AI_TYPE_SENSOR_GPS_ALTITUDE                       116     /*          */
#define MDC_AI_TYPE_SENSOR_SPEED                              120     /*          */
#define MDC_AI_TYPE_SENSOR_HEADING                            124     /*          */
#define MDC_AI_TYPE_ENCOUNTER                                 128     /*          */
/* The range 128-1023 is reserved for future expansion                            */

/*****************************************************************************
* All of the following are from NomPartition (MDC_PART_SCADA)                    *
*****************************************************************************/
#define MDC_TEMP_ROOM                              57436    /*                    */
#define MDC_PRESS_AIR_AMBIENT                      21764    /*                    */
#define MDC_REL_HUMIDITY_AMBIENT                   57732    /*                    */
#define MDC_LEVEL_POLLEN                           57736    /*                    */
#define MDC_LEVEL_DUST                             57740    /*                    */
#define MDC_WIND_VELOCITY                          57744    /*                    */
#define MDC_WIND_DIRECTION                         57740    /*                    */
```

```
/*******************************************************************************
* All of the following are from NomPartition – Partition: MDC_PART_PHD_AI               *
*******************************************************************************/
/*******************************************************************************
* Locations are encoded within 16 bits as…..                                             *
* upper 11 bits are the location type (for example bedroom - MDC_AI_LOCATION_BEDROOM)  *
* lower 5 bits are the location type instance identifier (for example bedroom 0, bedroom 1, etc.   *
* therefore, location viewed as a 16 bit entity would have the types assigned in blocks of 32     *
* The location type instance identifier allows for both a single dwelling with multiple rooms of that  *
* type, as well as a multiple family dwelling that may contain that type of room in each subunit     *
*******************************************************************************/
/* General                                                                        */
#define MDC_AI_LOCATION_UNKNOWN                1024    /*                         */
#define MDC_AI_LOCATION_UNSPECIFIED            1088    /*                         */
#define MDC_AI_LOCATION_RESIDENT               1152    /*                         */
#define MDC_AI_LOCATION_LOCALUNIT              1216    /*                         */
#define MDC_AI_LOCATION_BUILDING               1248    /* Buildings 0-31          */
#define MDC_AI_LOCATION_BUILDING_MID           1280    /* Buildings 32-63         */
#define MDC_AI_LOCATION_BUILDING_HIGH          1312    /* Buildings 64-95         */
#define MDC_AI_LOCATION_FLOOR                  1344    /* Floors 0-31             */
#define MDC_AI_LOCATION_FLOOR_MID              1376    /* Floors 32-63            */
#define MDC_AI_LOCATION_FLOOR_HIGH             1408    /* Floors 64-95            */
#define MDC_AI_LOCATION_BASEMENT               1440    /* Lower floors 0 - 31     */
#define MDC_AI_LOCATION_ZONE                   1472    /* Zone 0-31               */
#define MDC_AI_LOCATION_ZONE_MID               1504    /* Zone 32-63              */
#define MDC_AI_LOCATION_ZONE_HIGH              1536    /* Zone 64-95              */
#define MDC_AI_LOCATION_BEACON                 1568    /* e.g., wireless beacon   */
/* The range 1600-3071 is reserved for future expansion                           */

/* Rooms                                                                          */
#define MDC_AI_LOCATION_BEDROOM                3072    /*                         */
#define MDC_AI_LOCATION_BEDROOMMASTER          3136    /*                         */
#define MDC_AI_LOCATION_TOILET                 3200    /*                         */
#define MDC_AI_LOCATION_TOILETMAIN             3264    /*                         */
#define MDC_AI_LOCATION_OUTSIDETOILET          3328    /*                         */
#define MDC_AI_LOCATION_SHOWERROOM             3392    /*                         */
#define MDC_AI_LOCATION_KITCHEN                3456    /*                         */
#define MDC_AI_LOCATION_KITCHENMAIN            3520    /*                         */
#define MDC_AI_LOCATION_LIVINGAREA             3584    /*                         */
#define MDC_AI_LOCATION_LIVINGROOM             3648    /*                         */
#define MDC_AI_LOCATION_DININGROOM             3712    /*                         */
#define MDC_AI_LOCATION_STUDY                  3776    /*                         */
#define MDC_AI_LOCATION_HALL                   3840    /*                         */
#define MDC_AI_LOCATION_LANDING                3904    /*                         */
#define MDC_AI_LOCATION_STAIRS                 3968    /*                         */
#define MDC_AI_LOCATION_HALLLANDINGSTAIRS      4032    /*                         */
#define MDC_AI_LOCATION_GARAGE                 4096    /*                         */
#define MDC_AI_LOCATION_GARDENGARAGE           4160    /*                         */
#define MDC_AI_LOCATION_GARDENGARAGEAREA       4224    /*                         */
#define MDC_AI_LOCATION_FRONTGARDEN            4288    /*                         */
#define MDC_AI_LOCATION_BACKGARDEN             4352    /*                         */
#define MDC_AI_LOCATION_SHED                   4416    /*                         */
#define MDC_AI_LOCATION_CONSERVATORY           4480    /*                         */
/* The range 4544-7167 is reserved for future expansion                           */

/* Appliances – used for contact sensor and utility usage                         */
#define MDC_AI_APPLIANCE_KETTLE                7168    /*Note: Self heating       */
#define MDC_AI_APPLIANCE_TELEVISION            7232    /*                         */
#define MDC_AI_APPLIANCE_STOVE                 7296    /*                         */
```

```
#define MDC_AI_APPLIANCE_MICROWAVE         7360    /*               */
#define MDC_AI_APPLIANCE_TOASTER           7424    /*               */
#define MDC_AI_APPLIANCE_VACUUM            7488    /*               */
#define MDC_AI_APPLIANCE_APPLIANCE         7552    /*               */
#define MDC_AI_APPLIANCE_FAUCET            7616    /*               */
#define MDC_AI_APPLIANCE_OVEN              7648    /*               */
#define MDC_AI_APPLIANCE_FRIDGE            7680    /*               */
#define MDC_AI_APPLIANCE_COFFEEMAKER       7712    /*               */
#define MDC_AI_APPLIANCE_DISHWASHER        7744    /*               */
#define MDC_AI_APPLIANCE_CANOPENER         7776    /*               */
#define MDC_AI_APPLIANCE_FOODPROCESSOR     7808    /*               */
#define MDC_AI_APPLIANCE_MIXER             7840    /*               */
#define MDC_AI_APPLIANCE_EXTRACTORFAN      7872    /*               */
#define MDC_AI_APPLIANCE_HEATER            7904    /*               */
#define MDC_AI_APPLIANCE_BOILER            7936    /*               */
#define MDC_AI_APPLIANCE_FAN               7968    /*               */
#define MDC_AI_APPLIANCE_AIRCON            8000    /*               */
#define MDC_AI_APPLIANCE_LIGHT             8032    /*               */
#define MDC_AI_APPLIANCE_LIGHTSWITCH       8064    /*               */
#define MDC_AI_APPLIANCE_LAMP              8096    /*               */
#define MDC_AI_APPLIANCE_COMPUTER          8128    /*               */
#define MDC_AI_APPLIANCE_MONITOR           8160    /*               */
#define MDC_AI_APPLIANCE_PRINTER           8192    /*               */
#define MDC_AI_APPLIANCE_WASHINGMACHINE    8224    /*               */
#define MDC_AI_APPLIANCE_HOUSE             8256    /*               */
#define MDC_AI_APPLIANCE_GATEWAY           8288    /*               */
#define MDC_AI_APPLIANCE_SHOWER            8320    /*               */
#define MDC_AI_APPLIANCE_BATHROOMTAP       8352    /*               */
#define MDC_AI_APPLIANCE_KITCHENTAP        8384    /*               */
#define MDC_AI_APPLIANCE_TELEPHONE         8416    /*               */
#define MDC_AI_APPLIANCE_ENTRYPHONE        8448    /*               */
#define MDC_AI_APPLIANCE_MODEM             8480    /*               */
#define MDC_AI_APPLIANCE_TABLETPC          8512    /*               */
#define MDC_AI_APPLIANCE_CELLPHONE         8544    /*               */
#define MDC_AI_APPLIANCE_BLENDER           8576    /*               */
#define MDC_AI_APPLIANCE_JUICER            8608    /*               */
#define MDC_AI_APPLIANCE_OUTLET            8640    /*               */
#define MDC_AI_APPLIANCE_RADIATOR          8672    /*               */
/* The range 8304-9215 is reserved for future expansion            */

/* Doors                                                            */
#define MDC_AI_LOCATION_FRONTDOOR          9216    /*               */
#define MDC_AI_LOCATION_BACKDOOR           9280    /*               */
#define MDC_AI_LOCATION_FRIDGEDOOR         9344    /*               */
#define MDC_AI_LOCATION_MEDCABDOOR         9408    /*               */
#define MDC_AI_LOCATION_WARDROBEDOOR       9472    /*               */
#define MDC_AI_LOCATION_FRONTCUPBOARDDOOR  9536    /*               */
#define MDC_AI_LOCATION_OTHERDOOR          9600    /*               */
#define MDC_AI_LOCATION_SIDEDOOR           9632    /*               */
#define MDC_AI_LOCATION_WINDOW             9664    /* Windows 0-31  */
#define MDC_AI_LOCATION_WINDOW_MID         9696    /* Windows 32-63 */
#define MDC_AI_LOCATION_WINDOW_HIGH        9728    /* Windows 64-95 */
/* The range 9760-11263 is reserved for future expansion           */

/* Usage and location sensors                                       */
#define MDC_AI_LOCATION_BED                11264   /*               */
#define MDC_AI_LOCATION_CHAIR              11328   /*               */
#define MDC_AI_LOCATION_SOFA               11392   /*               */
#define MDC_AI_LOCATION_TOILET_SEAT        11456   /*               */
```

```
#define MDC_AI_LOCATION_STOOL                    11520    /*              */
#define MDC_AI_LOCATION_ARMCHAIR                 11552    /*              */
#define MDC_AI_LOCATION_DESK                     11584    /*              */
#define MDC_AI_LOCATION_SWING                    11616    /*              */
#define MDC_AI_LOCATION_TABLE                    11648    /*              */
#define MDC_AI_LOCATION_CUPBOARD                 11680    /*              */
#define MDC_AI_LOCATION_WHEELCHAIR               11712    /*              */
/* The range 11744-19999 is reserved for future expansion                 */

/* The following are specified in ZigBee Home Automation Profile
 – locations already defined in this standard are not repeated and the
 – existing code is given as a comment =>
 – the number may be determined as ((HA-1)*32+20000)
 – the ZigBee HA code is given as comment)                                */
/* Rooms                                                                  */
#define MDC_AI_LOCATION_ATRIUM                   20000    /* 0x01         */
#define MDC_AI_LOCATION_BAR                      20032    /* 0x02         */
#define MDC_AI_LOCATION_COURTYARD                20064    /* 0x03         */
#define MDC_AI_LOCATION_BATHROOM                 20096    /* 0x04         */
/* Bedroom => MDC_AI_LOCATION_BEDROOM_0          3072     0x05            */
#define MDC_AI_LOCATION_BILLIARDROOM             20160    /* 0x06         */
#define MDC_AI_LOCATION_UTILITYROOM              20192    /* 0x07         */
#define MDC_AI_LOCATION_CELLAR                   20224    /* 0x08         */
#define MDC_AI_LOCATION_CLOSET                   20256    /* 0x09         */
#define MDC_AI_LOCATION_THEATER                  20288    /* 0x0A         */
#define MDC_AI_LOCATION_OFFICE                   20320    /* 0x0B         */
#define MDC_AI_LOCATION_DECK                     20352    /* 0x0C         */
#define MDC_AI_LOCATION_DEN                      20384    /* 0x0D         */
/* Dining room => MDC_AI_LOCATION_DININGROOM_0   3712     0x0E            */
#define MDC_AI_LOCATION_ELECTRICALROOM           20448    /* 0x0F         */
#define MDC_AI_LOCATION_ELEVATOR                 20480    /* 0x10         */
#define MDC_AI_LOCATION_ENTRY                    20512    /* 0x11         */
#define MDC_AI_LOCATION_FAMILYROOM               20544    /* 0x12         */
#define MDC_AI_LOCATION_MAINFLOOR                20576    /* 0x13         */
/* Upstairs => MDC_AI_LOCATION_FLOOR_1           1345     0x14            */
/* Downstairs => MDC_AI_LOCATION_FLOOR_0         1344     0x15            */
/* Basement => MDC_AI_LOCATION_BASEMENT_0        1440     0x16            */
#define MDC_AI_LOCATION_GALLERY                  20704    /* 0x17         */
#define MDC_AI_LOCATION_GAMEROOM                 20736    /* 0x18         */
/* Garage => MDC_AI_LOCATION_GARAGE_0            4096     0x19            */
#define MDC_AI_LOCATION_GYM                      20800    /* 0x1A         */
/* Hallway => MDC_AI_LOCATION_HALL_0             3840     0x1B            */
#define MDC_AI_LOCATION_HOUSE                    20864    /* 0x1C         */
/* Kitchen => MDC_AI_LOCATION_KITCHEN_0          3456     0x1D            */
#define MDC_AI_LOCATION_LAUNDRYROOM              20928    /* 0x1E         */
#define MDC_AI_LOCATION_LIBRARY                  20960    /* 0x1F         */
/* Master bedroom => MDC_AI_LOCATION_BEDROOMMASTER 3136 0x20              */
#define MDC_AI_LOCATION_MUDROOM                  21024    /* 0x21         */
#define MDC_AI_LOCATION_NURSERY                  21056    /* 0x22         */
#define MDC_AI_LOCATION_PANTRY                   21088    /* 0x23         */
/* MDC_AI_LOCATION_OFFICE                        20320    0x24            */
#define MDC_AI_LOCATION_OUTSIDE                  21152    /* 0x25         */
#define MDC_AI_LOCATION_POOL                     21184    /* 0x26         */
#define MDC_AI_LOCATION_PORCH                    21216    /* 0x27         */
#define MDC_AI_LOCATION_SEWINGROOM               21248    /* 0x28         */
#define MDC_AI_LOCATION_SITTINGROOM              21280    /* 0x29         */
/* stairway => MDC_AI_LOCATION_STAIRS_0          3968     0x2A            */
#define MDC_AI_LOCATION_YARD                     21344    /* 0x2B         */
#define MDC_AI_LOCATION_ATTIC                    21376    /* 0x2C         */
```

```
#define MDC_AI_LOCATION_HOTTUB                      21408   /* 0x2D              */
/* Living room => MDC_AI_LOCATION_LIVINGROOM_0      3648    0x2E                 */
#define MDC_AI_LOCATION_SAUNA                       21472   /* 0x2F              */
#define MDC_AI_LOCATION_WORKSHOP                    21504   /* 0x30              */
#define MDC_AI_LOCATION_GUESTBEDROOM                21536   /* 0x31              */
#define MDC_AI_LOCATION_GUESTBATHROOM               21568   /* 0x32              */
#define MDC_AI_LOCATION_POWDERROOM                  21600   /* 0x33              */
#define MDC_AI_LOCATION_BACKYARD                    21632   /* 0x34              */
#define MDC_AI_LOCATION_FRONTYARD                   21664   /* 0x35              */
#define MDC_AI_LOCATION_PATIO                       21696   /* 0x36              */
#define MDC_AI_LOCATION_DRIVEWAY                    21728   /* 0x37              */
#define MDC_AI_LOCATION_SUNROOM                     21760   /* 0x38              */
/* Living room => MDC_AI_LOCATION_LIVINGROOM        3648    0x39                 */
#define MDC_AI_LOCATION_SPA                         21824   /* 0x3A              */
#define MDC_AI_LOCATION_WHIRLPOOL                   21856   /* 0x3B              */
/* Shed => MDC_AI_LOCATION_SHED_0                   4416    0x3C                 */
#define MDC_AI_LOCATION_EQUIPMENTSTORAGE            21920   /* 0x3D              */
#define MDC_AI_LOCATION_HOBBYROOM                   21952   /* 0x3E              */
#define MDC_AI_LOCATION_FOUNTAIN                    21984   /* 0x3F              */
#define MDC_AI_LOCATION_POND                        22016   /* 0x40              */
#define MDC_AI_LOCATION_RECEPTIONROOM               22048   /* 0x41              */
#define MDC_AI_LOCATION_BREAKFASTROOM               22080   /* 0x42              */
#define MDC_AI_LOCATION_NOOK                        22112   /* 0x43              */
#define MDC_AI_LOCATION_GARDEN                      22144   /* 0x44              */
#define MDC_AI_LOCATION_BALCONY                     22176   /* 0x45              */
#define MDC_AI_LOCATION_PANICROOM                   22208   /* 0x46              */
#define MDC_AI_LOCATION_TERRACE                     22240   /* 0x47              */
#define MDC_AI_LOCATION_ROOF                        22272   /* 0x48              */
#define MDC_AI_LOCATION_PLAYROOM                    22304   /* 0x49              */
/* The range 22728-24999 is reserved for future expansion                        */

/* The following are for medical facility rooms from ZigBee Healthcare Profile (ZHCP)  */
/* The number may be determined as ((ZHCP-0x64)*32+35000)                        */
#define MDC_AI_LOCATION_WAITINGROOM                 35000   /* 0x64              */
#define MDC_AI_LOCATION_TRIAGEROOM                  35032   /* 0x65              */
#define MDC_AI_LOCATION_DOCTORSOFFICE               35064   /* 0x66              */
#define MDC_AI_LOCATION_PATIENTSPRIVATEROOM         35096   /* 0x67              */
#define MDC_AI_LOCATION_CONSULTATIONROOM            35128   /* 0x68              */
#define MDC_AI_LOCATION_NURSESTATION                35160   /* 0x69              */
#define MDC_AI_LOCATION_WARD                        35192   /* 0x6A              */
#define MDC_AI_LOCATION_CORRIDOR                    35224   /* 0x6B              */
#define MDC_AI_LOCATION_OPERATINGTHEATRE            35256   /* 0x6C              */
#define MDC_AI_LOCATION_DENTALSURGERYROOM           35288   /* 0x6D              */
#define MDC_AI_LOCATION_MEDICALIMAGINGROOM          35320   /* 0x6E              */
#define MDC_AI_LOCATION_DECONTAMINATIONROOM         35352   /* 0x6F              */
#define MDC_AI_LOCATION_ICU                         35384   /*                   */
#define MDC_AI_LOCATION_CCU                         35416   /*                   */
#define MDC_AI_LOCATION_ER                          35448   /*                   */
#define MDC_AI_LOCATION_OBSERVATIONWARD             35480   /*                   */
/* The range 35512-54999 is reserved for future expansion                        */

/*******************************************************************************
* All of the following are from NomPartition (MDC_PART_PHD_AI)                   *
********************************************************************************/
/* Events                                                                        */
#define MDC_AI_EVT_PRESENCE_REGAINED                55000   /*                   */
#define MDC_AI_EVT_PRESENCE_LOST                    55001   /*                   */
#define MDC_AI_EVT_LOW_BATTERY                      55002   /*                   */
#define MDC_AI_EVT_BATTERY_REPLACED                 55003   /*                   */
```

```
#define MDC_AI_EVT_FAULT                        55004    /*                */
#define MDC_AI_EVT_FAULT_CLEARED                55005    /*                */
#define MDC_AI_EVT_END_OF_LIFE                  55006    /*                */
#define MDC_AI_EVT_TAMPER_DETECTED              55007    /*                */
#define MDC_AI_EVT_STUMBLE_DETECTED             55020    /*                */
#define MDC_AI_EVT_FALL_DETECTED                55021    /*                */
#define MDC_AI_EVT_FALL_RECOVERY_DETECTED       55022    /*                */
#define MDC_AI_EVT_PERS_ACTIVATED               55030    /*                */
#define MDC_AI_EVT_PERS_RESET                   55031    /*                */
#define MDC_AI_EVT_CONDITION_DETECTED           55040    /*                */
#define MDC_AI_EVT_CONDITION_CLEARED            55041    /*                */
#define MDC_AI_EVT_MOTION_DETECTED              55050    /*                */
#define MDC_AI_EVT_MOTION_DETECTED_DELAYED      55051    /*                */
#define MDC_AI_EVT_MOTION_ENDED                 55052    /*                */
#define MDC_AI_EVT_OCCUPANT_EXIT_PROPERTY       55060    /*                */
#define MDC_AI_EVT_OCCUPANT_ENTER_PROPERTY      55061    /*                */
#define MDC_AI_EVT_EXIT_DOOR_OPEN               55062    /*                */
#define MDC_AI_EVT_EXIT_DOOR_CLOSED             55063    /*                */
#define MDC_AI_EVT_EXIT_BOUNDARY                55064    /*                */
#define MDC_AI_EVT_ENTER_BOUNDARY               55065    /*                */
#define MDC_AI_EVT_ENURESIS_DETECTED            55070    /*                */
#define MDC_AI_EVT_ENURESIS_CLEARED             55071    /*                */
#define MDC_AI_EVT_CONTACT_OPENED               55080    /*                */
#define MDC_AI_EVT_CONTACT_CLOSED               55081    /*                */
#define MDC_AI_EVT_USAGE_STARTED                55090    /*                */
#define MDC_AI_EVT_USAGE_ENDED                  55091    /*                */
#define MDC_AI_EVT_USE_START_VIOLATION          55092    /*                */
#define MDC_AI_EVT_USE_STOP_VIOLATION           55093    /*                */
#define MDC_AI_EVT_ABSENCE_VIOLATION            55094    /*                */
#define MDC_AI_EVT_SWITCH_ON                    55100    /*                */
#define MDC_AI_EVT_SWITCH_OFF                   55101    /*                */
#define MDC_AI_EVT_DOSAGE_TAKEN                 55110    /*                */
#define MDC_AI_EVT_DOSAGE_MISSED                55111    /*                */
#define MDC_AI_EVT_DOSAGE_EMPTY                 55112    /*                */
#define MDC_AI_EVT_HIGH_TEMP_DETECTED           55120    /*                */
#define MDC_AI_EVT_LOW_TEMP_DETECTED            55121    /*                */
#define MDC_AI_EVT_NORMAL_TEMP_DETECTED         55122    /*                */
#define MDC_AI_EVT_TEMP_CHANGE_TOO_FAST         55123    /*                */
#define MDC_AI_EVT_HIGH_HUMIDITY_DETECTED       55130    /*                */
#define MDC_AI_EVT_NORMAL_HUMIDITY_DETECTED     55131    /*                */
#define MDC_AI_EVT_AT_LOCATION                  55140    /*                */
#define MDC_AI_EVT_NOT_AT_LOCATION              55141    /*                */
/* The range 55112-61439 (0x3400-0xEFFF) is reserved for future expansion  */

/**************************************************************************
* From Units Partition (MDC_PART_DIM)                                     *
**************************************************************************/
#define MDC_DIM_PERCENT                         544      /*                */
#define MDC_DIM_DEGC                            6048     /*                */
#define MDC_DIM_FAHR                            4416     /*                */
#define MDC_DIM_CUBIC_M                         1568     /*                */
#define MDC_DIM_WATT                            4032     /*                */
#define MDC_DIM_CUBIC_M_PER_HR                  2976     /*                */
#define MDC_DIM_L_PER_MIN                       3072     /*                */
#define MDC_DIM_ANG_DEG                         736      /*                */
#define MDC_DIM_M                               1280     /*                */
#define MDC_DIM_M_PER_SEC                       2816     /*                */
#define MDC_DIM_KILO_M_PER_HR                   12003    /*                */
#define MDC_DIM_MPH                             12032    /*                */
#define MDC_DIM_PER_DECI_L                      1712     /*                */
```

```
#define MDC_DIM_MILLI_G_PER_M_CUBE        2002     /*                  */
#define MDC_DIM_MICRO_G_PER_M_CUBE        2003     /*                  */
#define MDC_DIM_KILO_PASCAL               3843     /*                  */
#define MDC_DIM_BAR                       3936     /*                  */
#define MDC_DIM_MILLI_BAR                 3954     /*                  */
#define MDC_DIM_MMHG                      3872     /*                  */
#define MDC_DIM_INHG                      11968    /*                  */
#define MDC_DIM_PSI                       6592     /*                  */
#define MDC_DIM_KILO_WATT_HR              11779    /*                  */
```

C.1 Systematic derivations of terms and codes for independent living monitoring measurements

C.1.1 Base concepts

The target category is *device*. The different devices applied within the scenarios forming the basis of the DIM have different functionality. Accordingly, the following base concepts have been identified:

- **Sensor** (devices [or the subsystems of more complex devices] that acquires and reports data or an event.

C.1.2 First set of differentiating criteria, sensor

The first semantic link is based on the concept *performs* (typically afferent functions, particularly measurement, but also efferent functions such as regulation). The devices are, therefore, classified into a number of possible categories based on the functionality they perform.

C.1.2.1 Semantic link "*has measured property:*"

Applicable descriptors include the following:

- **Fall**
- **Pers**
- **Smoke**
- **CO**
- **Water**
- **Gas**
- **Motion**
- **Property exit**
- **Enuresis**
- **Contact closure**
- **Usage**
- **Switch**
- **Dosage**
- **Temperature**
- **Humidity**
- **Current ambient temperature**
- **Current ambient relative humidity**
- **Accumulated electrity usage**
- **Accumulated gas usage**

- **Accumulated water usage**
- **Accumulate solar generated**
- **Instantaneous electricity usage**
- **Instantaneous gas usage**
- **Instantaneous water usage**
- **Instantaneous solar generated**
- **Identity of person**
- **Location of sensor**
- **Base X coordinate**
- **Base Y coordinate**
- **Base Z coordinate**
- **Base coordinate angle**
- **Base location**
- **Relative X coordinate**
- **Relative Y coordinate**
- **Relative Z coordinate**
- **GPS latitude**
- **GPS longitude**
- **GPS altitude**
- **Speed**
- **Heading**

Table 67 —Nomenclature and codes for Infrastructure, device specialization

Systematic name	Common term	Acronym	Description/definition	Reference ID	Part::Code
Profile | Device | Aging Independently | Activity Hub	Aging Independently Activity Hub			MDC_DEV_SPEC_PROFILE_AI_ACTIVITY_HUB	8::4167

Table 68 —Nomenclature and codes for Infrastructure, Sub-specialization

Systematic name	Common term	Acronym	Description/definition	Reference ID	Part::Code
Profile | Device | Sub-specialization | Aging Independently | Sensor | Fall	Fall sensor			MDC_DEV_SUB_SPEC_PROFILE_FALL_SENSOR	8::4213
Profile | Device | Sub-specialization | Aging Independently | Sensor | PERS	PERS sensor	PERS		MDC_DEV_SUB_SPEC_PROFILE_PERS_SENSOR	8::4214
Profile | Device | Sub-specialization | Aging Independently | Sensor | Smoke	Smoke sensor			MDC_DEV_SUB_SPEC_PROFILE_SMOKE_SENSOR	8::4215
Profile | Device | Sub-specialization | Aging Independently | Sensor | CO	CO sensor			MDC_DEV_SUB_SPEC_PROFILE_CO_SENSOR	8::4216
Profile | Device | Sub-specialization | Aging Independently | Sensor | Water	Water sensor			MDC_DEV_SUB_SPEC_PROFILE_WATER_SENSOR	8::4217
Profile | Device | Sub-specialization | Aging Independently | Sensor | Gas	Gas sensor			MDC_DEV_SUB_SPEC_PROFILE_GAS_SENSOR	8::4218
Profile | Device | Sub-specialization | Aging Independently | Sensor | Motion	Motion sensor			MDC_DEV_SUB_SPEC_PROFILE_MOTION_SENSOR	8::4219
Profile | Device | Sub-specialization | Aging Independently | Sensor | Property Exit	Property exit sensor			MDC_DEV_SUB_SPEC_PROFILE_PROPEXIT_SENSOR	8::4220
Profile | Device | Sub-specialization | Aging Independently | Sensor | Enuresis	Enuresis sensor			MDC_DEV_SUB_SPEC_PROFILE_ENURESIS_SENSOR	8::4221
Profile | Device | Sub-specialization | Aging Independently | Sensor | Contact Closure	Contact closure sensor			MDC_DEV_SUB_SPEC_PROFILE_CONTACTCLOSURE_SENSOR	8::4222
Profile | Device | Sub-specialization | Aging Independently | Sensor | Usage	Usage sensor			MDC_DEV_SUB_SPEC_PROFILE_USAGE_SENSOR	8::4223
Profile | Device | Sub-specialization | Aging Independently | Sensor | Switch	Switch sensor			MDC_DEV_SUB_SPEC_PROFILE_SWITCH_SENSOR	8::4224
Profile | Device | Sub-specialization | Aging Independently | Sensor | Dosage	Dosage sensor			MDC_DEV_SUB_SPEC_PROFILE_DOSAGE_SENSOR	8::4225
Profile | Device | Sub-specialization | Aging Independently | Sensor | Temperature	Temperature sensor			MDC_DEV_SUB_SPEC_PROFILE_TEMP_SENSOR	8::4226
Profile | Device | Sub-specialization | Aging Independently | Sensor | Humidity	Humidity sensor			MDC_DEV_SUB_SPEC_PROFILE_HUMIDITY_SENSOR	8::4227

Systematic name	Common term	Acronym	Description/definition	Reference ID	Part::Code
Profile | Device | Sub-specialization | Aging Independently | Sensor | Location	Location sensor			MDC_DEV_SUB_SPEC_PROFILE_LOCATION_SENSOR	8::4228
Profile | Device | Sub-specialization | Aging Independently | Sensor | Temperature, Ambient	Ambiennt temperature sensor			MDC_DEV_SUB_SPEC_PROFILE_AMBIENT_TEMP_SENSOR	8::4229
Profile | Device | Sub-specialization | Aging Independently | Sensor | Humidity, Ambient	Ambient humidity sensor			MDC_DEV_SUB_SPEC_PROFILE_AMBIENT_HUMIDITY_SENSOR	8::4230
Profile | Device | Sub-specialization | Aging Independently | Sensor | Usage, Utility	Utility usage sensor			MDC_DEV_SUB_SPEC_PROFILE_UTILITY_USAGE_SENSOR	8::4231
Profile | Device | Sub-specialization | Aging Independently | Sensor | Usage, Utility, Instantaneous	Instantaneous utility usage sensor			MDC_DEV_SUB_SPEC_PROFILE_INSTANTANEOUS_UTILITY_USAGE_SENSOR	8::4232
Profile | Device | Sub-specialization | Aging Independently | Sensor | Location, GPS	GPS location sensor			MDC_DEV_SUB_SPEC_PROFILE_GPS_LOCATION_SENSOR	8::4233
Profile | Device | Sub-specialization | Aging Independently | Sensor | Speed	Speed sensor			MDC_DEV_SUB_SPEC_PROFILE_SPEED_SENSOR	8::4234
Profile | Device | Sub-specialization | Aging Independently | Sensor | Encounter	Encounter sensor			MDC_DEV_SUB_SPEC_PROFILE_ENCOUNTER	8::4235
Profile | Device | Sub-specialization | Aging Independently | Sensor | Wind Speed	Wind speed sensor			MDC_DEV_SUB_SPEC_PROFILE_WIND_SPEED	8::4236

Table 69 —Nomenclature and codes for sensor type

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Sensor | Fall	Fall sensor	Sensor to report when fall detected	MDC_AI_TYPE_SENSOR_FALL	130::1
Sensor | PERS	PERS sensor	Sensor to report when person indicates an event	MDC_AI_TYPE_SENSOR_PERS	130::2
Sensor | smoke	Smoke sensor	Sensor to report when smoke is detected	MDC_AI_TYPE_SENSOR_SMOKE	130::3
Sensor | CO	CO sensor	Sensor to report when CO is detected	MDC_AI_TYPE_SENSOR_CO	130::4
Sensor | water	Water sensor	Sensor to report when water is detected	MDC_AI_TYPE_SENSOR_WATER	130::5
Sensor | gas	Gas sensor	Sensor to report when gas is detected	MDC_AI_TYPE_SENSOR_GAS	130::6
Sensor | motion	Motion sensor	Sensor to report motion in a vicinity is detected	MDC_AI_TYPE_SENSOR_MOTION	130::7

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Sensor | property exit	Property exit sensor	Sensor to report a person exits or enters a property, and when door is left	MDC_AI_TYPE_SENSOR_PROPEXIT	130::8
Sensor | enuresis	Enuresis sensor	Sensor to report when enuresis is detected	MDC_AI_TYPE_SENSOR_ENURESIS	130::9
Sensor | contact closure	Contact closure sensor	Sensor to report a contact is closed or opened	MDC_AI_TYPE_SENSOR_CONTACTCLOSURE	130::10
Sensor | usage	Usage sensor	Sensor to report an appliance is in use	MDC_AI_TYPE_SENSOR_USAGE	130::11
Sensor | switch	Switch sensor	Sensor to report when a switch is closed or opened	MDC_AI_TYPE_SENSOR_SWITCH	130::12
Sensor | dosage	Dosage sensor	Sensor to report when medication is taken or forgotten	MDC_AI_TYPE_SENSOR_DOSAGE	130::13
Sensor | temperature	Temperature sensor	Sensor to report high or low temperature	MDC_AI_TYPE_SENSOR_TEMP	130::14
Sensor | humidity	Humidity sensor	Sensor to report over high humidity	MDC_AI_TYPE_SENSOR_HUMIDITY	130::15
Sensor | usage | accumulated | electricity	Accumulated electricity usage sensor	Accumulated usage of electricity	MDC_AI_TYPE_ELECTRICITY_ACCUMULATED_USAGE	130::24
Sensor | usage | accumulated | gas	Accumulated gas usage sensor	Accumulated usage of gas	MDC_AI_TYPE_GAS_ACCUMULATED_USAGE	130::28
Sensor | usage | accumulated | water	Accumulated water usage sensor	Accumulated usage of water	MDC_AI_TYPE_WATER_ACCUMULATED_USAGE	130::32
Sensor | generated | accumulated | solar	Accumulated generated solar electricity	Accumulated generation of solar electricity	MDC_AI_TYPE_SOLAR_ACCUMULATED	130::36
Sensor | usage | instantaneous | electricity	Instantaneous electricity usage sensor	Instantaneous usage of electricity	MDC_AI_TYPE_ELECTRICITY_INSTANTANEOUS_USAGE	130::40
Sensor | usage | instantaneous | gas	Instantaneous gas usage sensor	Instantaneous usage of gas	MDC_AI_TYPE_GAS_INSTANTANEOUS_USAGE	130::44
Sensor | usage | instantaneous | water	Instantaneous water usage sensor	Instantaneous usage of water	MDC_AI_TYPE_WATER_INSTANTANEOUS_USAGE	130::48
Sensor | generated | instantaneous | solar	Instantaneous generated solar electricity	Instantaneous generation of solar electricity	MDC_AI_TYPE_SOLAR_INSTANTANEOUS	130::52
Sensor | identity | person	Person identity	Iidentity of a person	MDC_AI_TYPE_PERSON_IDENTITY	130::56
Sensor | location, current	Sensor location	Sensor to report the current location	MDC_AI_TYPE_SENSOR_LOCATION	130::60

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Sensor | location, base	Base {X,Y,Z} coordinates	{X (latitude), Y(latitude), Z(altitude) } coordinates for the datum base coordinate of the defined space	MDC_AI_TYPE_BASE_COORD	130::64
Sensor | location, base, X	Base X coordinate	X coordinate (latitude) for the datum base coordinate of the defined space	MDC_AI_TYPE_BASE_COORD_X	130::68
Sensor | location, base, Y	Base Y coordinate	Y coordinate (longitude) for the datum base coordinate of the defined space	MDC_AI_TYPE_BASE_COORD_Y	130::72
Sensor | location, base, Z	Base Z coordinate	Z coordinate (altitude) for the datum base coordinate of the defined space	MDC_AI_TYPE_BASE_COORD_Z	130::76
Sensor | base, angle	Base coordinate angle	Angle of the X axis of the local coordinate space relative to geographic coordinates	MDC_AI_TYPE_BASE_COORD_ANGLE	130::80
Sensor | base, location	Base location	Datum base coordinate as a well-defined location within the defined space	MDC_AI_TYPE_BASE_LOCATION	130::84
Sensor | location, relative	Relative coordinates	{X,Y,Z} coordinates of the relative location within a defined space	MDC_AI_TYPE_SENSOR_REL_COORD	130::88
Sensor | location, relative, X	Relative X coordinate	X coordinate of the relative location within a defined space	MDC_AI_TYPE_SENSOR_REL_COORD_X	130::92
Sensor | location, relative, Y	Relative Y coordinate	Y coordinate of the relative location within a defined space	MDC_AI_TYPE_SENSOR_REL_COORD_Y	130::96
Sensor | location, relative, Z	Relative Z coordinate	Z coordinate of the relative location within a defined space	MDC_AI_TYPE_SENSOR_REL_COORD_Z	130::100
Sensor | location, GPS	GPS location	GPS location {latitude, longitude, altitude}	MDC_AI_TYPE_SENSOR_GPS_LOCATION	130::104
Sensor | location, GPS, latitude	GPS latitude	GPS latitude	MDC_AI_TYPE_SENSOR_GPS_LATITUDE	130::108
Sensor | location, GPS, longitude	GPS longitude	GPS longitude	MDC_AI_TYPE_SENSOR_GPS_LONGITUDE	130::112
Sensor | location, GPS, altitude	GPS altitude	GPS altitude	MDC_AI_TYPE_SENSOR_GPS_ALTITUDE	130::116
Sensor | speed	Speed	Sensor to report speed	MDC_AI_TYPE_SENSOR_SPEED	130::120
Sensor | heading	Heading	Sensor to report heading	MDC_AI_TYPE_SENSOR_HEADING	130::124

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Sensor | encounter	Encounter	Sensor to report encounter between agent and another object or person	MDC_AI_TYPE_ENCOUNTER	130::128

Table 70 —Nomenclature and codes for sensor type

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Temperature | | Room | Ambient	Room temperature	Room temperature	MDC_TEMP_ROOM	2::57436
Barometric Pressure | | Ambient, immediate patient environment	Ambient Pressure	Ambient (immediate patient environment) air pressure.	MDC_PRESS_AIR_AMBIENT	2::21764
Humidity, Relative | | Ambient, immediate patient environment	Ambient humidity	Ambient (immediate patient environment) relative humidty.	MDC_REL_HUMIDITY_AMBIENT	2::57732
Level | | Pollen	Pollen level	Pollen level.	MDC_LEVEL_POLLEN	2::57736
Level | | Dust	Dust level	Dust level.	MDC_LEVEL_DUST	2::57740
Velocity | | Wind	Wind velocity	Wind velocity	MDC_WIND_VELOCITY	2::57744
Direction | | Wind	Wind direction	Wind direction	MDC_WIND_DIRECTION	2::57748

C.1.3 Second set of differentiating criteria, location, general

One semantic link is applied for this set of differentiating criteria.

C.1.3.1 Semantic link "*locations:*"

Applicable descriptors include the following:

— **Location, general**

C.1.3.2 Discriminator Sets

This value denotes the location of a sensor. This location is a 16-bit value constructed, where the high 11 bits are the MDC_AI_LOCATION and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for building and the lower 5 bits are the value for the instance of the building. If only one instance of a location is defined, then the value of the unique instance shall be 0 and the base Reference ID shall be used (e.g., MDC_AI_LOCATION_BUILDING). If more than one instance of a location is defined, then locations shall be numbered from 01, and the value 0 shall not be used. The Reference ID shall denote the unique instance as the number of that instance as a suffix comprising the instance number preceeded by underscore (_) (e.g., MDC_AI_LOCATION_BUILDING_03).

Systematic derivations of terms and codes for general locations are outlined in Table 71.

Table 71—Nomenclature for terms and codes for locations, general

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Location | Unknown	Unknown location	An unknown location	MDC_AI_LOCATION_UNKNOWN	130::1024
Location | Unspecified	Unspecified location	The location is not specified	MDC_AI_LOCATION_UNSPECIFIED	130::1088
Location | Resident	Resident	A sensor connected to a resident	MDC_AI_LOCATION_RESIDENT	130::1152
Location | LocalUnit	Local unit	A local control unit such as emergency phone	MDC_AI_LOCATION_LOCALUNIT	130::1216
Location | Building, number, low	Building	Building (0) or a numbered building (1-31) within a site	MDC_AI_LOCATION_BUILDING	130::1248
Location | Building, number, mid	Building	A numbered building (32-63) within a site	MDC_AI_LOCATION_BUILDING_MID	130::1280
Location | Building, number, high	Building	A numbered building (64-95) within a site	MDC_AI_LOCATION_BUILDING_HIGH	130::1312
Location | Floor, level, low	Floor	Floor (0) or a numbered floor (1-31) within a building	MDC_AI_LOCATION_FLOOR	130::1344
Location | Floor, level, mid	Floor	A numbered floor (32-63) within a building	MDC_AI_LOCATION_FLOOR_MID	130::1376
Location | Floor, level, high	Floor	A numbered floor (64-95) within a building	MDC_AI_LOCATION_FLOOR_HIGH	130::1408
Location | Basement	Basement/Lower floor	Basement (0) or a numbered lower floor (-1- -31) within a building	MDC_AI_LOCATION_BASEMENT	130::1440
Location | Zone, low	Zone	A numbered zone (0-31) within a site	MDC_AI_LOCATION_ZONE	130::1472
Location | Zone, mid	Zone	A numbered zone (32-63) within a site	MDC_AI_LOCATION_ZONE_MID	130::1504
Location | Zone, high	Zone	A numbered zone (64-95) within a site	MDC_AI_LOCATION_ZONE_HIGH	130::1536
Location | Beacon	Beacon	A radio beacon	MDC_AI_LOCATION_BEACON	130::1568

C.1.4 Second set of differentiating criteria, location, room

One semantic link is applied for this set of differentiating criteria.

C.1.4.1 Semantic link “*locations:*”

Applicable descriptors include the following:

— **Location, room**

C.1.4.2 Discriminator Sets

This value denotes the location of a sensor. This location is a 16-bit value constructed, where the high 11 bits are the MDC_AI_LOCATION and the lower 5 bits are the unique instance of that location. For example, the high 11 bits could be the value for bedroom and the lower 5 bits are the value for that instance of the bedroom. If only one instance of a location is defined, then the value of the unique instance shall be 0 and the base Reference ID shall be used (e.g., MDC_AI_LOCATION_BEDROOM). The Reference ID shall denote the unique instance as the number of that instance as a suffixcomprising the instance number preceeded by underscore (_) (e.g., MDC_AI_LOCATION_BEDROOM_03).

Systematic derivations of terms and codes are outlined in Table 72.

Table 72 —Nomenclature for terms and codes for locations, rooms

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Location | Room | Bedroom	Bedroom	A bedroom with a building (0-31)	MDC_AI_LOCATION_BEDROOM	130::3072
Location | Room | Bedroom, master	Master bedroom	The master bedroom in a building	MDC_AI_LOCATION_BEDROOMMASTER	130::3136
Location | Room | Toilet	Toilet	A toilet within a building (0-31)	MDC_AI_LOCATION_TOILET	130::3200
Location | Room | Toilet, main	Main toilet	The main toilet in a building	MDC_AI_LOCATION_TOILETMAIN	130::3264
Location | Room | Toilet, outside	Outside toilet	A toilet in a structure away from the main building	MDC_AI_LOCATION_OUTSIDETOILET	130::3328
Location | Room | Showerroom	Shower room	A shower room (0-31)	MDC_AI_LOCATION_SHOWERROOM	130::3392
Location | Room | Kitchen	Kitchen	A kitchen (0-31)	MDC_AI_LOCATION_KITCHEN	130::3456
Location | Room | Kitchen, main	Main kitchen	The main kitchen in a building	MDC_AI_LOCATION_KITCHENMAIN	130::3520
Location | Room | Living area	Living area	A living area (0-31)	MDC_AI_LOCATION_LIVINGAREA	130::3584
Location | Room | Living room	Living room	A living room (0-31)	MDC_AI_LOCATION_LIVINGROOM	130::3648
Location | Room | Dining room	Dining room	A dining room (0-31)	MDC_AI_LOCATION_DININGROOM	130::3712
Location | Room | Study	Study	A study (0-31)	MDC_AI_LOCATION_STUDY	130::3776
Location | Room | Hall	Hall	A hallway (0-31)	MDC_AI_LOCATION_HALL	130::3840
Location | Room | Landing	Landing	A landing on the stairs (0-31)	MDC_AI_LOCATION_LANDING	130::3904
Location | Room | Stairs	Stairs	A set of stairs (0-31)	MDC_AI_LOCATION_STAIRS	130::3968
Location | Room | Hall landing	Hall landing	The hallway landing	MDC_AI_LOCATION_HALLLANDINGSTAIRS	130::4032
Location | Room | Garage	Garage	A garage (0-31)	MDC_AI_LOCATION_GARAGE	130::4096
Location | Room | Garage, garden	Garden garage	A garage separate from the main building (0-31)	MDC_AI_LOCATION_GARDENGARAGE	130::4160
Location | Room | Garage, garden, area	Garden garage area	The area adjacent to a garage	MDC_AI_LOCATION_GARDENGARAGEAREA	130::4224
Location | Room | Garden, front	Front garden	The front garden	MDC_AI_LOCATION_FRONTGARDEN	130::4288
Location | Room | Garden, back	Back garden	The back garden	MDC_AI_LOCATION_BACKGARDEN	130::4352
Location | Room | Shed	Shed	A shed (0-31)	MDC_AI_LOCATION_SHED	130::4416
Location | Room | Conservatory	Conservatory	A conservatory (0-31)	MDC_AI_LOCATION_CONSERVATORY	130::4480
Location | Room | Atrium	Atrium	An atrium (0-31)	MDC_AI_LOCATION_ATRIUM	130::20000

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Location | Room | Bar	Bar	A bar (0-31)	MDC_AI_LOCATION_BAR	130::20032
Location | Room | Courtyard	Courtyard	A courtyard (0-31)	MDC_AI_LOCATION_COURTYARD	130::20064
Location | Room | Bathroom	Bathroom	A bathroom (0-31)	MDC_AI_LOCATION_BATHROOM	130::20096
Location | Room | Billiard room	Billiard room	A billiard room holding table for such game (0-31)	MDC_AI_LOCATION_BILLIARDROOM	130::20160
Location | Room | Utility room	Utility room	A utility room (0-31)	MDC_AI_LOCATION_UTILITYROOM	130::20192
Location | Room | Cellar	Cellar	A cellar (0-31)	MDC_AI_LOCATION_CELLAR	130::20224
Location | Room | Closet	Closet	A closet (0-31)	MDC_AI_LOCATION_CLOSET	130::20256
Location | Room | Theater	Theater	A theater housing entertainment system (0-31)	MDC_AI_LOCATION_THEATER	130::20288
Location | Room | Office	Office	A room designated as anoffice (0-31)	MDC_AI_LOCATION_OFFICE	130::20320
Location | Room | Deck	Deck	A deck	MDC_AI_LOCATION_DECK	130::20352
Location | Room | Den	Den	A den (0-31)	MDC_AI_LOCATION_DEN	130::20384
Location | Room | Electrical room	Electrical room	A room containing electrical utility equipment	MDC_AI_LOCATION_ELECTRICALROOM	130::20448
Location | Room | Elevator	Elevator	An elevator (0-31)	MDC_AI_LOCATION_ELEVATOR	130::20480
Location | Room | Entry	Entry	An entry way to a building (0-31)	MDC_AI_LOCATION_ENTRY	130::20512
Location | Room | Family room	Family room	A family room (0-31)	MDC_AI_LOCATION_FAMILYROOM	130::20544
Location | Room | Main floor	Main floor	The main (ground) floor of a building	MDC_AI_LOCATION_MAINFLOOR	130::20576
Location | Room | Gallery	Gallery	A gallery (0-31)	MDC_AI_LOCATION_GALLERY	130::20704
Location | Room | Game room	Game room	A game room (0-31)	MDC_AI_LOCATION_GAMEROOM	130::20736
Location | Room | Gym	Gym	A gym (0-31)	MDC_AI_LOCATION_GYM	130::20800
Location | Room | House	House	A house	MDC_AI_LOCATION_HOUSE	130::20864
Location | Room |Laundry room	Laundry room	A laundry room (0-31)	MDC_AI_LOCATION_LAUNDRYROOM	130::20928
Location | Room | Library	Library	A library (0-31)	MDC_AI_LOCATION_LIBRARY	130::20960
Location | Room | Mud room	Mud room	A mud room (0-31)	MDC_AI_LOCATION_MUDROOM	130::21024
Location | Room | Nursery	Nursery	A nursery (0-31)	MDC_AI_LOCATION_NURSERY	130::21056
Location | Room | Pantry	Pantry	A pantry (0-31)	MDC_AI_LOCATION_PANTRY	130::21088
Location | Room | Ourside	Outside	The general area outside of the building	MDC_AI_LOCATION_OUTSIDE	130::21152
Location | Room | Pool	Pool	A pool	MDC_AI_LOCATION_POOL	130::21184

Systematic name	Common term	Description/definition	Reference ID	Part::Code
Location | Room | Porch	Porch	A porch	MDC_AI_LOCATION_PORCH	130::21216
Location | Room | Sewing room	Sewing room	A sewing room	MDC_AI_LOCATION_SEWINGROOM	130::21248
Location | Room | Sitting room	Sitting room	A sitting room (0-31)	MDC_AI_LOCATION_SITTINGROOM	130::21280
Location | Room | Yard	Yard	A yard	MDC_AI_LOCATION_YARD	130::21344
Location | Room | Attic	Attic	An attic	MDC_AI_LOCATION_ATTIC	130::21376
Location | Room | Hot tub	Hot tub	A hot tub	MDC_AI_LOCATION_HOTTUB	130::21408
Location | Room | Sauna	Sauna	A sauna	MDC_AI_LOCATION_SAUNA	130::21472
Location | Room | Workshop	Workshop	A workshop (0-31)	MDC_AI_LOCATION_WORKSHOP	130::21504
Location | Room | Bedroom, guest	Guest bedroom	A guest bedroom (0-31)	MDC_AI_LOCATION_GUESTBEDROOM	130::21536
Location | Room | Bathroom, guest	Guest bathroom	A guest bathroom (01-31)	MDC_AI_LOCATION_GUESTBATHROOM	130::21568
Location | Room | Powder room	Powder room	A powder room (1/2 bathroom)	MDC_AI_LOCATION_POWDERROOM	130::21600
Location | Room | Back yard	Back yard	A back yard	MDC_AI_LOCATION_BACKYARD	130::21632
Location | Room | Front yard	Front yard	A front yard	MDC_AI_LOCATION_FRONTYARD	130::21664
Location | Room | Patio	Patio	A patio	MDC_AI_LOCATION_PATIO	130::21696
Location | Room | Driveway	Driveway	A driveway	MDC_AI_LOCATION_DRIVEWAY	130::21728
Location | Room | Sun room	Sun room	A sun room	MDC_AI_LOCATION_SUNROOM	130::21760
Location | Room | Spa	Spa	A spa	MDC_AI_LOCATION_SPA	130::21824
Location | Room | Whirlpool	Whirlpool	A whirlpool	MDC_AI_LOCATION_WHIRLPOOL	130::21856
Location | Room | Equipment storage room	Equipment storage room	An equipment storage room	MDC_AI_LOCATION_EQUIPMENTSTORAGE	130::21920
Location | Room | Hobby room	Hobby room	A hobby room	MDC_AI_LOCATION_HOBBYROOM	130::21952
Location | Room | Fountain	Fountain	A fountain (0-31)	MDC_AI_LOCATION_FOUNTAIN	130::21984
Location | Room | Pond	Pond	A pond (0-31)	MDC_AI_LOCATION_POND	130::22016
Location | Room | Reception room	Reception room	A reception room	MDC_AI_LOCATION_RECEPTIONROOM	130::22048
Location | Room | Breakfast room	Breakfast room	A breakfast room	MDC_AI_LOCATION_BREAKFASTROOM	130::22080
Location | Room | Nook	Nook	A nook	MDC_AI_LOCATION_NOOK	130::22112
Location | Room | Garden	Garden	A garden	MDC_AI_LOCATION_GARDEN	130::22144
Location | Room | Balcony	Balcony	A balcony	MDC_AI_LOCATION_BALCONY	130::22176
Location | Room | Panic room	Panic room	A panic room	MDC_AI_LOCATION_PANICROOM	130::22208
Location | Room | Terrace	Terrace	A terrace (0-31)	MDC_AI_LOCATION_TERRACE	130::22240
Location | Room | Roof	Roof	A roof	MDC_AI_LOCATION_ROOF	130::22272