ZigBee Document 075360r15

ZigBee Health Care™ Profile Specification

ZigBee Profile: 0x0108

Revision 15

Version 1.0

March 2010

Sponsored by:
ZigBee Alliance

Accepted for release by:
ZigBee Alliance Board of Directors.

Abstract:
This profile defines device descriptions and standard practices for Health Care applications. The application domains covered are Disease Monitoring, Personal Wellness Monitoring and Personal Fitness monitoring. The environments addressed are residential environments, retirement communities, medical care facilities (low acuity aspects only) and fitness centers.

Keywords:
ZigBee, Profile, Health, ZHC, PHHC, HC, Medical, Application Framework.

ZigBee Alliance, Inc.
2400 Camino Ramon, Suite 375
San Ramon, CA 94583, USA

Participants

The following is a list of those who were members of the ZigBee Architecture Review Committee (ZARC) leadership when this document was released:

Skip Ashton: *Chair*

Phil Jamieson: *Vice-Chair*

When the document was released, the ZigBee Health Care Working Group leadership was composed of the following members:

Matt Perkins: *Chair*

Jon Adams: *Vice Chair*

Phil Rudland: *Technical Editor*

Table of Contents

1 Introduction 9
1.1 Scope 9
1.2 Purpose 9

2 References 10
2.1 ZigBee Alliance documents 10
2.2 ISO / IEEE Standards Documents 10

3 Definitions 12
3.1 Conformance levels 12
3.2 ZigBee Definitions 12
3.3 Definitions specific to this profile 13

4 Acronyms and abbreviations 14

5 Profile Description 15
5.1 Stack profile 15
5.1.1 Additional restrictions 15
5.2 Device descriptions 17
5.3 ZigBee Cluster Library (ZCL) 18
5.4 Clusters used in this profile 19

6 Constants 21

7 Device Descriptions 24
7.1 Introduction 24
7.1.1 Device Terminology 24
7.1.2 Device Descriptions and 11073 Device Specializations 24
7.1.3 Cluster Usage 24
7.1.4 Bulk data transfer 27
7.2 Disease Management Devices 28
7.3 Health and Fitness Devices 29
7.4 Aging Independently Devices 30
7.5 Multifunction Devices 31
7.5.1 Generic Multifunction Healthcare Device 31
7.6 Data Management Devices 32
7.6.1 Introduction 32
7.6.2 Data Collection Unit (DCU) 32

8 Commissioning 34
8.1 Deployment scenarios 34
8.1.1 Common initial features for all scenarios 34
8.2 Service Provider scenario 35
8.2.1 Actions before delivery 35
8.2.2 Commissioning phase 36
8.2.3 Joining the network 36
8.2.4 Application Pairing of devices 37
8.3 In-house Commissioning scenario 38
8.3.1 Actions before delivery 38
8.3.2 Commissioning phase 38
8.3.3 Joining the network 39

8.3.4 Application Pairing of devices 39
8.4 Consumer scenario 40
8.4.1 Actions before delivery 40
8.4.2 Commissioning phase 40
8.4.3 Joining the network 40
8.4.4 Application Pairing of devices 41
8.5 Device indications 41
8.6 Application Layer Security 41
8.6.1 Introduction 41
8.6.2 Trust Server role 42
8.6.3 Using the ASKE cluster 42
8.6.4 Using the ASAC cluster 44

9 Candidate ZCL Material for use with this Profile 47

A.1 11073 Protocol Tunnel cluster 47
A.1.1 Overview 47
A.1.2 Server 48
A.1.2.1 Dependencies 48
A.1.2.2 Attributes 48
A.1.2.2.1 DeviceIDList attribute 49
A.1.2.2.2 Manager target attribute 50
A.1.2.2.3 Manager endpoint attribute 50
A.1.2.2.4 Connected attribute 50
A.1.2.2.5 Preemptible attribute 50
A.1.2.2.6 Idle timeout attribute 50
A.1.2.3 Commands Received 51
A.1.2.3.1 Transfer APDU Command 51
A.1.2.3.2 Connect Request Command 52
A.1.2.3.3 Disconnect Request Command 53
A.1.2.3.4 Connect Status Notification Command 54
A.1.2.4 Commands Generated 55
A.1.3 Client 55
A.1.3.1 Dependencies 55
A.1.3.2 Attributes 55
A.1.3.3 Commands Received 55
A.1.3.4 Commands Generated 55

List of Figures

Figure 2 – Transfer APDU payload....51
Figure 3 – Connect Request command payload....52
Figure 4 – Connect control field format....52
Figure 5 – Disconnect Request command payload....53
Figure 6 – Connect Status Notification command payload....54

List of Tables

Table 1 – Document revision change history 8
Table 2 – Device Descriptions specified in the HC profile 17
Table 3 – Clusters used in the HC profile 19
Table 4 – Constants Specific to the HC Profile 21
Table 5 – Clusters common to all device descriptions 24
Table 6 – Disease Management Devices currently defined in the HC profile 28
Table 7 – Health and Fitness Devices currently defined in the HC profile 29
Table 8 – Aging Independently Devices currently defined in the HC profile 30
Table 9 – Additional Clusters Supported by Medical Data Management devices 32
Table 10 – Startup Attribute Values – common set 35
Table 11 – Startup Attribute Values –Service Provider scenario. 36
Table 12 – Startup Attribute Values (In-house Commissioning scenario) – as delivered 38
Table 13 – Startup Attribute Values (In-house Commissioning scenario) – after commissioning 39
Table 14 – Startup Attribute Values – Consumer scenario. 40
Table 15 – HC-specific LDC property types 46
Table 16 – Provisional Clusters ID allocation for Candidate clusters 47
Table 17 – Attributes of the 11073 Protocol Tunnel server cluster 49
Table 18 – Command IDs for the 11073 protocol tunnel cluster 51
Table 19 – Connect status values 54

Change history

Table 1 shows the change history for this specification.

Table 1 – Document revision change history

Revision	Version	Description
15	1.0	Accepted by the ZigBee Alliance Board of Directors

1 Introduction

1.1 Scope

This profile defines device descriptions and standard practices for ZigBee Health Care™ (ZHC) applications implemented on a ZigBee compliant platform.

The principal application domains and use cases catered for are

- Disease Management (DM)
  - Non-critical patient monitoring (episodic)
  - Non-critical patient monitoring (continuous)
  - Patient alarm monitoring (low acuity)
  - Drug administration (e.g. insulin pumps)
- Personal Fitness Monitoring (PFM)
  - Monitoring / tracking fitness level
  - Personalized fitness schedule
- Personal Wellness Monitoring (PWM)
  - Activity monitoring
  - Safety Monitoring
  - Living independently

See the ZHC MRD [R6] for details of these use cases. This document covers only the ZigBee related aspects of these use cases.

The environments addressed by this profile are

- Residential environments, with size ranging from a single room up to a floor area of 20,000 square feet, as specified in the Home Automation Application Profile [R10].
- Retirement communities, nursing homes
- Medical care facilities e.g. hospital, physicians office, outpatient surgery center (low acuity aspects only)
- Fitness centers

Note – 'acuity' is a medical term used in triage, meaning 'urgency'. 'Low acuity' above denotes low-urgency monitoring scenarios.

1.2 Purpose

This specification provides standard interfaces and device descriptions to allow inter-operability among ZigBee devices produced by various manufacturers of health care products.

2 References

The following standards and specifications contain provisions, which through reference in this document constitute provisions of this specification. All the standards and specifications listed are normative references. At the time of publication, the editions indicated were valid. All standards and specifications are subject to revision, and parties to agreements based on this specification are encouraged to investigate the possibility of applying the most recent editions of the standards and specifications indicated below.

2.1 ZigBee Alliance documents

[R1] ZigBee document 053474, ZigBee Specification

[R2] ZigBee document 08006, ZigBee-2007 Layer PICS and Stack Profiles

[R3] ZigBee document 075123, ZigBee Cluster Library Specification: Foundation Specification chapter

[R4] ZigBee document 075123, ZigBee Cluster Library Specification: General Specification chapter

[R5] ZigBee document 075123, ZigBee Cluster Library Specification: Protocol Interfaces chapter

[R6] ZigBee document 074828, Personal, Home and Hospital Care - Market Requirements

[R7] ZigBee document 075111, Personal, Home and Hospital Care - Technical Requirements

[R8] ZigBee document 095203, Health Care Profile - Part2, Security Clusters

[R9] ZigBee document 095167, Device List for the ZigBee Health Care Application Profile

[R10] ZigBee document 053520, Home Automation Profile Specification

[R11] ZigBee document 075307, Telecom Applications Profile Specification

[R12] ZigBee document 064309, Commissioning Framework

[R13] ZigBee document 075329, IP Gateway Specification

[R14] ZigBee document 105567, ASKE and ASAC best practice and usage guidelines

2.2 ISO / IEEE Standards Documents

[R15] Institute of Electrical and Electronics Engineers, Inc., IEEE Std. 802.15.4 2003, IEEE Standard for Information Technology Telecommunications and Information Exchange between Systems – Local and Metropolitan Area Networks – Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs). New York: IEEE Press. 2003

[R16] ISO/IEEE 11073-20601: Health Informatics - Personal Health Device Communication - Application Profile - Optimized Exchange Protocol - version 1.0 or later.

[R17] ISO/IEEE P11073-10404, Health informatics – Personal health device communication – Device specialization – Pulse oximeter.

[R18] ISO/IEEE P11073-10407, Health informatics – Personal health device communication – Device specialization – Blood pressure monitor.

[R19] ISO/IEEE P11073-10408, Health informatics – Personal health device communication – Device specialization – Thermometer.

[R20] ISO/IEEE P11073-10415, Health informatics – Personal health device communication – Device specialization – Weighing scale.

[R21] ISO/IEEE P11073-10417, Health informatics – Personal health device communication – Device specialization – Glucose meter.

[R22] ISO/IEEE P11073-10419, Health informatics – Personal health device communication – Device specialization – Insulin Pump

[R23] ISO/IEEE P11073-10421, Health informatics – Personal health device communication – Device specialization – Peak Expiratory Flow Monitor

[R24] ISO/IEEE P11073-10441, Health informatics – Personal health device communication – Device specialization – Cardiovascular Fitness and Activity Monitor.

[R25] ISO/IEEE P11073-10442, Health informatics – Personal health device communication – Device specialization – Strength Fitness Equipment.

[R26] ISO/IEEE P11073-10471, Health informatics – Personal health device communication – Device specialization – Independent living activity hub.

[R27] ISO/IEEE P11073-10472, Health informatics – Personal health device communication – Device specialization – Medication Monitor.

3 Definitions

3.1 Conformance levels

Expected: A key word used to describe the behavior of the hardware or software in the design models *assumed* by this profile. Other hardware and software design models may also be implemented.

May: A key word indicating a course of action permissible within the limits of the standard (may equals is permitted).

Shall: A key word indicating mandatory requirements to be strictly followed in order to conform to the standard; deviations from shall are prohibited (shall equals is required to).

Should: A key word indicating that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others; that a certain course of action is preferred but not necessarily required; or, that (in the negative form) a certain course of action is deprecated but not prohibited (should equals is recommended that).

3.2 ZigBee Definitions

Attribute: A data entity which represents a physical quantity or state. This data is communicated to other devices using commands.

Cluster: A collection of related attributes and commands, which together define a communications interface between two devices. The devices implement server and client sides of the interface respectively.

Cluster identifier: A 16-bit number unique within the scope of an application profile which identifies a specific cluster. Note however that clusters used in standard application profiles are given unique numbers across all such profiles (see [R3]).

Device: A device consists of one or more ZigBee device descriptions (e.g. thermometer and pulse oximeter) and their corresponding application profile(s), each on a separate endpoint, that share a single 802.15.4 radio (see [R15]). Each device has a unique 64-bit IEEE address.

Device Description: A collection of clusters and associated functionality implemented on a ZigBee endpoint. Device descriptions are defined in the scope of an application profile. Each device description has a unique identifier that is exchanged as part of the discovery process.

Node: Same as a device.

Product: A product is a unit that is intended to be marketed. It may implement a combination of private, published, and standard application profiles.

Trust Center: The device trusted by devices within a ZigBee network to distribute keys for the purpose of network and end-to-end application configuration management (see [R1]).

ZigBee Coordinator: An IEEE 802.15.4-2003 PAN coordinator (see [R15]).

ZigBee End Device: An IEEE 802.15.4-2003 RFD (Reduced Function Device) or FFD (Full Function Device) (see [R15]) participating in a ZigBee network, which is neither the ZigBee coordinator nor a ZigBee router.

ZigBee Router: An IEEE 802.15.4-2003 FFD (Full Function Device) participating in a ZigBee network, which is not the ZigBee coordinator but may act as an IEEE 802.15.4-2003 coordinator within its personal operating space, that is capable of routing messages between devices and supporting associations.

3.3 Definitions specific to this profile

Security Domain: collection of all ZigBee devices entitled to establish pairwise Application Link Keys by the possession of alpha-secure keying material.

Trust Server: the entity trusted by the devices within a Security Domain to initially distribute keying material shares and related security parameters – such as certificates, access control roles and access control policies - for the purpose of end-to-end application link key establishment and access control, according to the ASKE and/or ASAC cluster (see [R8]).

4 Acronyms and abbreviations

AIB	Application support layer Information Base
APDU	Application Protocol Data Unit
APS	Application Support Sub-layer
ASAC	Alpha-Secure Access Control
ASDU	APS Service Data Unit
ASKE	Alpha-Secure Key Establishment
DCU	Data Collection Unit
EPID	Extended PAN Identifier
GAP	Gateway/Access Point
HC	Health Care
ILAH	Independent Living Activity Hub
LK	Link Key
MK	Master Key
MT	Mobile Terminal
PAN	Personal Area Network
PERS	Personal Emergency Response System
PHHC	Personal, Home and Hospital Care (the earlier domain title of this profile)
PICS	Protocol Implementation Conformance Statement
SAS	Startup Attribute Set
TC	Trust Center
TC-LK	Trust Center Link Key
TS	Trust Server
TS-LK	Trust Server Link Key
VOZ	Voice Over ZigBee
ZCL	ZigBee Cluster Library

5 Profile Description

5.1 Stack profile

End devices that conform to this specification may use either the ZigBee Feature Set Stack Profile or the ZigBee PRO Feature Set stack profile, as defined in [R2]. All devices that conform to this specification and are capable of routing shall use the ZigBee PRO stack profile.

In addition to the requirements specified in [R2], the following requirements are specified by this application profile.

5.1.1 Additional restrictions

The various parameters referenced in this section are defined in [R1] or [R2].

Network Join Application Settings:-

- Config_NWK_Scan_Attempts is recommended to be set to 2.
- Config_NWK_Time_btwn_Scans is recommended to be set to 10ms.
- Config_Permit_Join_Duration is recommended to be set to 20 seconds, to allow the consumer commissioning scenario described in 8.4. This duration is chosen to give the installing person time to read and carry out the instructions re button pushing. In the other commissioning scenarios this parameter is not used.

Source Bind Application Settings:-

- Config_Max_Bind (the size of the source binding table) should be set per physical device (node) to a minimum value of one per client cluster, plus any value needed for all unsolicited commands that may be sent (such as attribute reports), summed over all endpoints.
  If a device is required to support additional profiles, this number should be incremented by the amount required by the additional profiles.

End Device Bind Application Settings:-

- Config_EndDev_Bind_Timeout is recommended to be set to 20 seconds.

Fragmentation Settings:-

- See 7.1.4 for details on when APS fragmentation shall be used.
  Where APS fragmentation is implemented, the following parameters shall be employed:-
  apscMaxWindowSize = 1
  apscInterFrameDelay parameter is not used for this value of apscMaxWindowSize.

Security Settings:-

- Trust Center Security Mode (actually the security mode of the whole network) – Standard Security shall be supported, with preconfigured link keys.
- TC Link Key Delivery in the Clear is not permitted.
- Network Key Delivery in the Clear is not permitted.

- When needed (see scenarios and requirements discussed in section 7.2 of [R7]), the communication between the devices that support the HC profile may also be secured by employing an alpha-secure key distribution scheme (see [R8]). However, the profile does not make this feature mandatory as some devices might not have enough resources (e.g. memory, processing power, etc.) to implement this feature.
- When management of access control identities, roles and policies is needed (see section 7.1.2 of [R7] for example use cases), the access to the data produced by the devices that support the HC profile may be securely controlled by employing the alpha-secured access control cluster (see [R8]). However, the profile does not make this feature mandatory as some devices might not have enough resources (e.g. memory, processing power) to implement this feature.

PICS parameters (general):-

- Radio Frequency Operation - to ensure global interoperability, the frequency shall be 2.4 GHz.

PICS parameters (Network Layer):-

- Many to one route discovery (PICS parameter NLF112):-
  All router capable devices shall support use of many to one route discovery, and the TC shall be designated as a concentrator. HC data management devices (see 7.5) should also be designated as concentrators. The radius parameter for many to one route discovery should be set to zero, which results in the appropriate default radius value being automatically chosen. Polling rate for end devices (PICS parameter NLF17):-
  This parameter is not specified for HC, except/unless where specified in individual device descriptions. However, to avoid a low battery life, it is recommended that the period should not be less than 7.5 seconds. Devices with a long polling period will have to be activated 'out of band', e.g. by a button press, to enable timely communication with a data management device.
- Number of child end devices (PICS parameter NLF27):-
  This parameter applies only to data management devices (see 7.5), and should be equal to 16.

PICS parameters (Security):-

- Network key change policies (PICS parameter NLS5):-
  Subject to application requirements, it is recommended that the network key is not changed on a periodic basis, but only if it is suspected that the network has been compromised.

  To avoid delays in getting the new key from the TC (e.g. when a medical sensor that is normally asleep is switched on to communicate an urgent reading and finds the key has changed), it is recommended that devices should check periodically, with a period appropriate to application requirements, whether they have the correct network key. If they do not, they must obtain a new one using their TC link key.
- TC link keys (PICS parameter ALS7):-
  TC link keys are mandatory for all devices.

PICS parameters (Application Layer):-

- Auxiliary APS security header (PICS parameters ADF4/ACF500/ACF501):-
  This profile makes use of application-level security and thus this header is required.
- Fragmentation (PICS parameters ADF5/ADF6):-
  APS fragmentation/reassembly is required under the conditions detailed in 7.1.4.

5.2 Device descriptions

Device descriptions specified in this profile are listed in Table 2 along with their respective Device IDs. The device descriptions are organized according to the end application areas they address. For device descriptions other than Data Management and Multifunction, the Device ID shall equal the MDC_DEV_SPEC_PROFILE_* value or MDC_DEC_SUB_SPEC_PROFILE_* value of the corresponding IEEE specialization or IEEE device sub-specialization, respectively.

Table 2 – Device Descriptions specified in the HC profile

	Device Description	Device ID
Data Management	Data Collection Unit (DCU)	0x0000
	Reserved	0x0001 – 0x0eff
Multifunction	Generic Multifunction Healthcare Device	0x0f00
	Reserved	0x0f01 – 0x0fff
Disease Management	Pulse Oximeter	0x1004
	ECG	0x1006
	Blood Pressure Monitor	0x1007
	Thermometer	0x1008
	Weight Scale	0x100f
	Glucose Meter	0x1011
	International normalized ratio (INR)	0x1012
	Insulin Pump	0x1013
	Peak Flow Monitor	0x1015
Health and Fitness	Cardiovascular fitness and activity monitor	0x1029
	Strength fitness equipment	0x102a
	Physical Activity Monitor	0x102b
	Step counter	0x1068
Aging Independently	Independent Living Activity Hub (ILAH)	0x1047
	Adherence Monitor	0x1048
	Fall Sensor	0x1075
	PERS Sensor	0x1076
	Smoke Sensor	0x1077
	CO Sensor	0x1078
	Water Sensor	0x1079

	Gas Sensor	0x107a
	Motion Sensor	0x107b
	Property Exit Sensor	0x107c
	Enuresis Sensor	0x107d
	Contact Closure Sensor	0x107e
	Usage Sensor	0x107f
	Switch Use Sensor	0x1080
	Dosage Sensor	0x1081
	Temperature Sensor	0x1082
	Reserved	All other values in the range 0x1000-0xffff

At the time of publication of this version of the profile, the devices shown in Table 2 were defined. However, further devices are expected to be added in future versions as new IEEE 11073 device specializations are developed. The current full list of devices is contained in [R9].

A product that conforms to this specification shall implement at least one of these device descriptions. It may implement more than one - for example, it may implement both the thermometer and blood pressure monitor device descriptions, in order to provide both a thermometer application and a blood pressure monitor application.

Note that manufacturer specific device descriptions may not be added to any public profile (see [R3]), so any such device descriptions shall reside on a separate endpoint and use a private profile ID.

5.3 ZigBee Cluster Library (ZCL)

This profile utilizes clusters specified in the ZigBee Cluster Library (ZCL), see [R4]. The Cluster IDs used for the clusters are those given in the ZCL.

The implementation details for each cluster are given in the ZCL specifications. Further specification and clarification is given in this profile where necessary.

This profile also utilizes the clusters specified in Annex A of this document. This annex specifies candidate clusters for incorporation into the ZCL.

The ZCL provides a mechanism for clusters to report attribute value changes. It also provides commands to configure the reporting parameters. The attributes that a particular cluster is capable of reporting are listed in the ZCL specification for each cluster. Products shall support the reporting mechanism for all attributes specified in the ZCL that the product implements within a given cluster.

Where reporting is enabled for a cluster, the minimum reporting interval shall be set to a value greater than or equal to one second. It is recommended that the minimum reporting interval be set to a higher value whenever the application can tolerate it. The maximum reporting interval should be set to zero (disables periodic reporting) by default, and if it is set to a non-zero value it shall be set to a value greater than or equal to one minute and greater than the value of the minimum reporting interval. It is recommended that the maximum reporting interval be set to as great a value as the application can tolerate to avoid unnecessary traffic.

5.4 Clusters used in this profile

The clusters used in this profile, are listed in Table 3. The clusters are listed according to the functional domain they belong to in the ZCL. The corresponding cluster identifiers can be found in the ZCL Foundation specification [R3].

The functionality made available by all supported clusters shall be that given in their ZCL specifications except where a device description in this profile includes further specification, clarification or restriction as needed for a particular device.

Most clusters include optional attributes. The application designer must be aware that optional attributes may be not implemented on another device. It is the responsibility of a device's application to discover and deal with unsupported attributes on other devices.

It is expected that clusters will continue to be developed in the ZCL that will be useful in this profile. In many cases, new clusters will be organized into new device descriptions that are separate from those currently defined. There may also be situations where it makes sense to add clusters as new optional elements of existing device descriptions.

Adding a new mandatory cluster to a device description should preferably be done by creating a new device ID and accompanying device description, as adding it to an existing device description may cause backward compatibility issues. If clusters (either server or client) are added that are designated as mandatory for future devices, the updated specification shall note that legacy devices may not support them, and shall specify how devices built to the new specification shall interoperate with legacy devices that do not support them.

Table 3 – Clusters used in the HC profile

Functional Domain	Cluster Name	Cluster ID
General	Basic	0x0000
General	Power Configuration	0x0001
General	Identify	0x0003
General	Alarms	0x0009
General	Time	0x000a
General	RSSI Location	0x000b
General	Commissioning	0x0015
General	Partition	0x0016 (Note 1)
General	Alpha-Secure Key Establishment	0x0017 (Note 2)
General	Alpha-Secure Access Control	0x0018 (Note 3)
Protocol Interfaces	Generic Tunnel	0x0600
Protocol Interfaces	11073 Protocol Tunnel	0x0614 (Note 4)
Telecommunications	Voice over ZigBee	0x0904 (Note 1)

Note 1:- This cluster is currently not in the ZigBee Cluster Library. It is specified in [R11], and its ID has been allocated in the Cluster ID Database maintained by the ZigBee Cluster Library Development Board (CLDB).

Note 2:- This cluster is currently not in the ZigBee Cluster Library. It is specified in [R8], and its ID has been allocated in the Cluster ID Database maintained by the CLDB.

Note 3:- This cluster is currently not in the ZigBee Cluster Library. It is specified in [R8], and its ID has been allocated in the Cluster ID Database maintained by the CLDB.

Note 4:- This cluster is currently not in the ZigBee Cluster Library. It is specified in A.1, and its ID has been allocated in the Cluster ID Database maintained by the CLDB.

6 Constants

Profile-specific constants are shown in Table 4. The *PhysicalEnvironment* attribute of the Basic cluster may be used to indicate the location of a device. If more detailed location information is required, the *LocationDescription* attribute of the Basic cluster could be used to satisfy the requirement.

Table 4 – Constants Specific to the HC Profile

Constant	Description	Value
Application Profile Identifier	Value of the Application Profile Identifier for use in the Simple Descriptor of any endpoint supporting this profile.	0x0108
minHCGroups	Minimum number of groups that shall be supported per node that implements the Groups cluster.	1
Values of the *PhysicalEnvironment* attribute of the Basic cluster for use with this profile.	Values specified by Home Automation [R10] Atrium Bar Courtyard Bathroom Bedroom Billiard Room Utility Room Cellar Closet Theater Office Deck Den Dining Room Electrical Room Elevator Entry Family Room Main Floor Upstairs Downstairs Basement/Lower Level Gallery Game Room Garage Gym Hallway House Kitchen Laundry Room Library Master Bedroom Mud Room (small room for coats and boots) Nursery Pantry Office Outside Pool Porch Sewing Room Sitting Room Stairway	 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x0a 0x0b 0x0c 0x0d 0x0e 0x0f 0x10 0x11 0x12 0x13 0x14 0x15 0x16 0x17 0x18 0x19 0x1a 0x1b 0x1c 0x1d 0x1e 0x1f 0x20 0x21 0x22 0x23 0x24 0x25 0x26 0x27 0x28 0x29 0x2a

	Yard	0x2b
	Attic	0x2c
	Hot Tub	0x2d
	Living Room	0x2e
	Sauna	0x2f
	Shop/Workshop	0x30
	Guest Bedroom	0x31
	Guest Bath	0x32
	Powder Room (1/2 bath)	0x33
	Back Yard	0x34
	Front Yard	0x35
	Patio	0x36
	Driveway	0x37
	Sun Room	0x38
	Living Room	0x39
	Spa	0x3a
	Whirlpool	0x3b
	Shed	0x3c
	Equipment Storage	0x3d
	Hobby/Craft Room	0x3e
	Fountain	0x3f
	Pond	0x40
	Reception Room	0x41
	Breakfast Room	0x42
	Nook	0x43
	Garden	0x44
	Balcony	0x45
	Panic Room	0x46
	Terrace	0x47
	Roof	0x48
	Values specified by the Independent Living Activity Hub [R24]	
	Toilet	0x49
	Toilet Main	0x4a
	Outside Toilet	0x4b
	Shower room	0x4c
	Study	0x4d
	Front Garden	0x4e
	Back Garden	0x4f
	Kettle	0x50
	Television	0x51
	Stove	0x52
	Microwave	0x53
	Toaster	0x54
	Vacuum	0x55
	Appliance	0x56
	Front Door	0x57
	Back Door	0x58
	Fridge Door	0x59
	Medication Cabinet Door	0x60
	Wardrobe Door	0x61
	Front Cupboard Door	0x62
	Other Door	0x63
	Medical Facility Rooms	
	Waiting Room	0x64
	Triage Room	0x65
	Doctor's Office	0x66
	Patient's Private Room	0x67
	Consultation Room	0x68
	Nurse Station	0x69
	Ward	0x6a

	Corridor Operating Theatre Dental Surgery Room Medical Imaging Room Decontamination Room Reserved for future additions Available for use by vendors	0x6b 0x6c 0x6d 0x6e 0x6f 0x70-0x77 0x78-0x7f

1

7 Device Descriptions

7.1 Introduction

7.1.1 Device Terminology

Except where otherwise indicated, the term 'device' is used in this section to indicate the functionality described by a Device Description, as opposed to the physical node on which this functionality is implemented (see 3.2).

Note that device manufacturers may combine the functionalities of two or more Device Descriptions, each on a separate endpoint, into a single physical device. The functionality of such a “combination” (physical) device is the sum of the functions of its constituent devices.

7.1.2 Device Descriptions and 11073 Device Specializations

Each ZigBee Health Care device (apart from multifunction devices) is identified with an 11073 Device Specialization. For ‘combination’ devices (see 7.1.1) each endpoint should be handled as an independent communications channel at the 11073 level - all 11073 communication required to support its device specialization, such as association, should take place with that endpoint and not shared with other endpoints.

The IEEE 11073 protocol also allows for multiple device specializations to be provided via a single 11073 association. A special device type is provided for this possibility (see 7.5).

7.1.3 Cluster Usage

7.1.3.1 Mandatory and Optional Clusters

Each device description supports a number of mandatory clusters. It also supports some optional clusters. The optional clusters, if implemented, have a known (and described if not obvious) role in device operation and interaction with the other clusters present in the device, and are considered part of the device from the point of view of this profile.

7.1.3.2 Common Clusters

Support for certain clusters is common to all the device descriptions in this profile. The clusters shown in Table 5 shall be supported by all devices descriptions in this profile as mandatory or optional according to the designation given here. Individual device descriptions specify further mandatory and optional clusters per device, and may place further restrictions on support of the optional clusters shown in Table 5.

Table 5 – Clusters common to all device descriptions

Server side	Client side
Mandatory	
Basic	-
Identify	-
Generic Tunnel	
11073 Protocol Tunnel	11073 Protocol Tunnel
Optional	

Server side	Client side
Power Configuration	-
Alarms	-
Time	-
RSSI Location	RSSI Location
ASKE (see [R8])	ASKE (see [R8])
ASAC (see [R8])	ASAC (see [R8])
Partition (see 7.1.4)	Partition (see 7.1.4)
Commissioning cluster	-
-	Generic Tunnel
Manufacturer specific (See 7.1.3.3 for details)	Manufacturer specific (See 7.1.3.3 for details)

7.1.3.2.1 Basic Cluster

The mandatory Basic server cluster holds read-only information about the device (such as model identifier and manufacturer name), allows user device information such as location to be set, and provides commands for enabling or disabling a device and resetting it to factory defaults.

7.1.3.2.2 Identify Cluster

The mandatory Identify server cluster allows the device to be put into an Identification mode, in which the device uses a physical indication (e.g. a light or sound) to indicate to an observer (e.g. an installer) which of several devices it is.

7.1.3.2.3 Generic Tunnel Cluster

The Generic Tunnel cluster provides a basic common set of attributes and commands necessary to tunnel any protocol. The server cluster is mandatory on all devices, as data is tunneled in both directions.

7.1.3.2.4 11073 Protocol Tunnel Cluster

The mandatory 11073 Protocol Tunnel cluster (see A.1) provides specific commands for tunneling data conforming to the IEEE 11073 protocol. Both server and client clusters are mandatory, as data is tunneled in both directions.

Some optional commands and attributes are additionally mandated based on the device type, as follows:

- If one of the Data Management Devices (see 7.6) is implemented, then the device shall implement transmission of the connect request command, and reception of the connect status notification command.
- If any other device type is implemented, then the device shall implement the *manager target*, *manager endpoint*, *connected*, *preemptible* and *idle timeout* attributes, and shall implement reception of the connect request and disconnect commands, and transmission of the connect status notification command.

This cluster requires the presence of the Generic Tunnel server cluster on the same endpoint.

7.1.3.2.5 Power Configuration Cluster

The optional Power Configuration server cluster has attributes for determining information about a device's power source(s), and for configuring under/over voltage alarms (e.g. that a battery voltage has dropped below a threshold). To send an alarm, it requires the presence of the Alarms cluster on the same endpoint.

7.1.3.2.6 Alarms Cluster

The optional Alarms server cluster provides the means for other clusters to send alarm notifications.

7.1.3.2.7 Time Cluster

The optional Time server cluster provides the means for all devices in a network to synchronize their real-time clocks. Synchronized clocks are important e.g. for 11073 message time stamps.

Note that if a device changes the time on the agent device by means of a ZigBee command, the IEEE 11073 implementation running on the agent might need to take proper action to account for the clock change.

7.1.3.2.8 RSSI Location Cluster

This optional cluster provides a means for exchanging Received Signal Strength Indication (RSSI) information between devices, thus allowing estimates of location of the devices to be made. Examples of the use of location information include

- for a Fall Sensor (see 7.4), knowing where the patient fell.
- Per a PERS device (see 7.4), determining that an elopement event is happening because the location of the patient is known.

7.1.3.2.9 ASKE and ASAC Clusters

Motivation for the use of the ASKE or ASAC cluster is described in 8.6.

7.1.3.2.10 Partition Cluster

The Partition cluster is used for transfer of bulk data which is longer than can be transferred using ZigBee fragmentation. Details of its use in this profile are given in 7.1.4.

Where used in this profile, the partition cluster handshake phase shall consist of the transmission of the WriteHandshakeParam command only. The ReadHandshakeParam and ReadHandshakeParamResponse commands shall not be used.

7.1.3.2.11 Commissioning Cluster

The optional Commissioning server cluster provides a standard facility for commissioning and management of ZigBee health care devices.

7.1.3.3 Manufacturer specific clusters

The ZCL provides a range of cluster IDs that are reserved for manufacturer specific clusters. Manufacturer specific clusters may be added to any device description specified in this profile, as described in the ZCL Foundation specification [R3].

7.1.3.4 Use of other ZCL clusters

As well as the specified mandatory and optional clusters for a device, any other ZCL cluster(s) may be included on the same endpoint. However, such clusters are not formally considered part of the device from the point of view of this profile.

7.1.4 Bulk data transfer

Devices described in this profile may need to transfer application layer commands which would result in physical layer frames that are longer than the maximum size of a single ZigBee packet (aMaxPHYPacketSize [R15]). To accomplish this, the following approach shall be used (but see 7.1.4.1).

- If the maximum size ASDU (see [R1]) required by any cluster supported will fit in a single ZigBee packet, then APS fragmentation need not be implemented.

- If the maximum size ASDU required by any cluster supported will not fit in a single ZigBee packet but is less than or equal to 240 bytes, then APS Fragmentation shall be implemented, supporting up to at least the maximum size required by any cluster supported.

- If the maximum size ASDU required by any cluster supported is larger than 240 bytes then the Partition cluster (see [R11]) shall be implemented, supporting up to at least the maximum size required by any cluster supported, and additionally APS Fragmentation shall be implemented, supporting up to 240 bytes.

When calculating the maximum frame size required, the following points apply

- For the 11073 Protocol Tunnel cluster (see A.1) the theoretical maximum supported is 64kbytes, but devices are not required to support this for most device specializations. Rather they need only support up to the limits given in the relevant device specialization document. In each case that document quotes maximum sizes for incoming and outgoing transfers. Devices shall support transport of messages up to (at least) the specified maximum incoming size.

- Data Management devices shall support transport of messages up to 64kBytes (the above indicates that they need to support the partition cluster up to 64kBytes and APS Fragmentation up to 240 bytes).

- The ASKE and ASAC clusters (see [R8]) do not require use of the Partition cluster, but require APS fragmentation up to 240 bytes. The ASKE and ASAC configuration commands are highly modular, and can be used with frame sizes of less than 240 bytes.

The settings for the APS fragmentation mechanism are specified in 5.1.1.

7.1.4.1 Streaming data

As an exception to the above, this profile also addresses the use case when episodic streaming of medical data is required. Such a use case tolerates data loss but requires minimizing the latency. If the packets to be transmitted are large and it is known that both ends support the partition cluster and that the transmission path is one hop only, streaming can be achieved by employing the partition cluster with its NumberOfACKFrame attribute set to 0x00.

7.2 Disease Management Devices

This category of devices includes sensing, measurement and drug administration devices primarily for use in disease management.

All disease management devices shall support the clusters specified in 7.1.3.2.

To provide role-based access control to the generated data, as well as distributed key establishment of application link keys, disease management devices should implement the alpha-secure access control cluster. Alternatively, to enable efficient, distributed key agreement for the establishment of application link keys only, disease management devices should implement the alpha-secure key establishment cluster.

For the current definitive list of devices, see [R9]. At the time of publication of this version of the profile, those devices which were defined in this profile are listed in Table 6. For each device, the information exchanged via the 11073 Protocol Tunnel cluster shall conform to the given Device Specialization Document.

Table 6 – Disease Management Devices currently defined in the HC profile

Device	Device ID	Device Specialization Document	Device function
Pulse Oximeter	0x1004	[R17]	Indirectly measures the amount of oxygen in a patient's blood. Expected frequency of measurement is typically once a day.
ECG	0x1006	See note [1]	Measures electrical activity of the heart over time to provide 1 to 3 channel electrocardiographic waveforms or derived heart rate.
Blood Pressure Meter	0x1007	[R18]	Measures a patient's blood pressure. Expected frequency of measurement is several times a day.
Thermometer	0x1008	[R19]	Measures the body temperature of a patient. The expected frequency of measurement is several times a day.
Weight Scale	0x100f	[R20]	Measures the weight of a patient. Expected frequency of measurement is from once per week to several times a day.

[1] The corresponding IEEE device specialization has not been published yet at time of finalization of this document. See [R9] for the current full list of device descriptions

Device	Device ID	Device Specialization Document	Device function
Glucose Meter	0x1011	[R19]	Measures the approximate concentration of glucose in a patient's blood. It is used by disease (e.g. diabetes) management applications. For current generation devices, the expected frequency of measurement is several times a day. Future generation devices are expected to function in continuous mode and as such, the frequency of measurement is application dependent.
International normalized ratio (INR)	0x1012	See note [1]	Determines the normalized ratio for the coagulation time when a thromboplastin is added to a sample of venous or capillary blood.
Insulin Pump	0x1013	[R22]	Administers insulin to a patient, via subcutaneous cannula (tube), in the treatment of diabetes.
Peak Flow Meter	0x1015	[R23]	Measures a patient's maximum speed of expiration (peak expiratory flow rate).

7.3 Health and Fitness Devices

This category of devices includes sensing and measurement devices primarily for use managing health and fitness.

All health and fitness devices shall support the clusters specified in 7.1.3.2.

To provide role-based access control to the generated data, as well as distributed key establishment of application link keys, health and fitness devices should implement the alpha-secure access control cluster. Alternatively, to enable efficient, distributed key agreement for the establishment of application link keys only, health and fitness devices should implement the alpha-secure key establishment cluster.

For the current definitive list of devices, see [R9]. At the time of publication of this version of the profile, those devices which were defined in this profile are listed in Table 7. For each device, the information exchanged via the 11073 Protocol Tunnel cluster shall conform to the given Device Specialization Document.

Table 7 – Health and Fitness Devices currently defined in the HC profile

Device	Device ID	Device Specialization Document	Device function
Cardiovascular fitness and activity monitor	0x1029	[R24]	Measures physical actions and the body's various physiological responses to that activity. Cardiovascular fitness and activity-monitor devices include e.g. treadmills, exercise bikes, heart rate monitors, bike computers, pedometers, and overall activity/lifestyle monitors.

Device	Device ID	Device Specialization Document	Device function
Strength fitness equipment	0x102a	[R25]	Measures musculo-skeletal strength-conditioning activities, e.g. the extent to which a person can perform a certain motion with a given resistance.
Physical activity monitor	0x102b	See note [2]	Measures kinematic data (typically acceleration, speed, position, and orientation) of a point in space.
Step counter	0x1068	See note [2]	Measures walking distance.

7.4 Aging Independently Devices

Aging Independently devices detect events that may have consequences for the health of an individual (typically an aging individual living alone), for example a fall, a gas leak, or medicine not being taken. Those devices which are currently defined in this profile are listed in Table 8.

For each device, the information exchanged via the 11073 Protocol Tunnel cluster shall conform to the given Device Specialization Document, which is [R26] for Aging Independently devices. Note that individual devices need only implement that subset of functionality defined by [R26] that is necessary for that particular device.

Table 8 – Aging Independently Devices currently defined in the HC profile

Device	Device ID	Device Specialization Document	Device function
ILAH	0x1047	[R26]	Aggregates activity data sensor events from multiple sensor data sources, used in the support of the independent living of one or more individuals.
Adherence Monitor	0x1048	[R27]	Provides a record (dosage, time of ingestion, user feedback) of the person's usage of medication for monitoring the person's adherence to a medication regime.
Fall Sensor	0x1075	[R26]	Detects when a person wearing it falls.
PERS Sensor	0x1076	[R26]	Used by an individual to raise an alarm, e.g. by pressing a button on the device.
Smoke Sensor	0x1077	[R26]	Detects smoke in the atmosphere.
CO Sensor	0x1078	[R26]	Detects carbon monoxide in the atmosphere.
Water Sensor	0x1079	[R26]	Detects unexpected presence of water.
Gas Sensor	0x107a	[R26]	Detects levels of gas in the atmosphere above safe limits.
Motion Sensor	0x107b	[R26]	Detects movement within a certain area around the sensor.

[2] The corresponding IEEE device specialization has not been published yet at time of finalization of this document. See [R9] for the current full list of device descriptions

Device	Device ID	Device Specialization Document	Device function
Property Exit Sensor	0x107c	[R26]	Detects exit of an occupant from the premises.
Enuresis Sensor	0x107d	[R26]	Detects occurrences of involuntary urination or bedwetting
Contact Closure Sensor	0x107e	[R26]	Detects opening or closing of a contact – used for e.g. doors, windows, pressure mats.
Usage Sensor	0x107f	[R26]	Detects usage, and abnormal usage patterns, of e.g. a bed or chair.
Switch Use Sensor	0x1080	[R26]	Reports change of state (on/off) of a switch used to control electrical apparatus.
Dosage Sensor	0x1081	[R26]	Detects whether an individual has taken or missed a predetermined medication dose from a dispenser within a specified time.
Temperature Sensor	0x1082	[R26]	Monitors whether the temperature in an environment is within preset limits.

7.4.1.1 Supported clusters

In addition to those clusters specified in 7.1.3.2, Aging Independently devices may also support the Voice over ZigBee (VOZ) cluster (see [R11]).

This optional cluster enables voice communication with a data management device such as the Gateway / Access Point (see 7.5). Subject to resource constraints, this cluster may be employed by any Personal Health device, such as the PERS or Fall Detector, in order to enable voice communication in an emergency.

The client and server VOZ clusters are responsible for sending and receiving speech, respectively. Thus, if two way speech communication is required, both server and client clusters are needed.

To maximize available bandwidth, Voice over ZigBee communication may be inactivated during sending of 11073 Tunnel commands.

7.5 Multifunction Devices

A Multifunction Device offers the functionality of multiple 11073 device specializations behind a single 11073 tunnel. Device descriptions in this category shall be used for devices which implement multiple IEEE 11073 device specializations and make those available by establishing a single 11073 tunnel.

The exact list of the device types supported is provided in the *DeviceIDList* attribute of the *11073 Protocol Tunnel* cluster.

Only one Multifunction device is currently defined by this profile.

7.5.1 Generic Multifunction Healthcare Device

The Generic Multifunction Healthcare Device can combine the functionalities of any types of healthcare device from any of the other categories (excluding Data Management devices).

The mandatory and optional supported clusters are the union of the mandatory and optional clusters respectively specified for the individual device types.

7.6 Data Management Devices

7.6.1 Introduction

Data Management devices implement an IEEE 11073 manager (see [R16]).

Only one Data Management device is currently defined by this profile.

End devices implementing Data Management device shall operate as RX_ON_WHEN_IDLE devices.

7.6.1.1 Supported clusters

In addition to those clusters specified in 7.1.3.2, all Data Management devices shall support the clusters listed in Table 9.

The Generic Tunnel server, 11073 Protocol Tunnel client and server, and Partition server clusters are mandatory so that they are able to communicate with any medical sensing device.

A data management device may also optionally employ the Voice over ZigBee (VOZ) cluster to enable voice communication with Aging Independently devices, e.g. a Fall detector in an emergency. The client and server VOZ clusters are responsible for sending and receiving speech, respectively. Thus, if two way speech communication is required, both server and client clusters are needed. It is thus recommended that data management devices support both server and client VOZ clusters.

Table 9 – Additional Clusters Supported by Medical Data Management devices

Server side	Client side
Mandatory	
Partition	-
Optional	
Voice over ZigBee	Voice over ZigBee
-	Basic
-	Identify
-	Partition

7.6.2 Data Collection Unit (DCU)

The Data Collection Unit (DCU) gathers the data from a number of on-body medical and non-medical devices. The DCU may perform local aggregation and/or compression before sending the data to a gateway or directly to a backend server.

Additionally, a DCU may support some of the functions defined in the TA profile ([R11]), HA profile ([R10]) and Gateway specification ([R13]).

Typically, a DCU is a portable device; it may be removed from the network and reintroduced to it, or may be moved to another part of the network, with only temporary loss of connection.

8 Commissioning

Commissioning is the process of initializing the devices to join a network and to work together.

The ZigBee Alliance has recognized the importance of commissioning and, in particular, the importance of specifications for commissioning in a multi-vendor environment. A general commissioning framework specification may be found in [R12], and a commissioning cluster is specified in [R4].

This section gives details of how the general techniques described in these documents are applied to the HC domain.

8.1 Deployment scenarios

Commissioning details depend on the deployment scenario. Three deployment scenarios are addressed by this profile, as follows.

1. Service provider scenario. In this scenario, a service provider that provides patient monitoring services is responsible for providing all the devices that are part of the network, and preloading these devices with all the information that they need to securely join the network and work together.
2. In-house commissioning scenario. In this scenario, the network owner (e.g. a medical care facility) has its own in-house commissioning facility, to configure the devices with all the information that they need to securely join the network and work together.
3. Consumer scenario. This scenario covers the case of networks, where the network owner does not have a service provider, and wishes to purchase devices from multiple providers and install them themself.

The Consumer scenario shall be supported. The Service provider and In-house commissioning scenarios may be supported.

These scenarios are described in the following sections. For each scenario, the actions involved are grouped as follows:-

1. "Actions before delivery": vendor actions before the device is delivered to the customer.
2. "Commissioning phase": commissioning at the customer site.
3. "Joining the network": actions the device performs when/after joining the operational network.
4. "Application Pairing of devices": actions at the customer site to pair the device with other devices that it should communicate with. Device discovery is carried out by the Data Management device, and may be achieved using any of the mandatory and optional features of the ZDO or ZCL. When an agent device has been selected, the Data Management device uses the connect request command of the 11073 tunnel cluster to connect a tunnel.

8.1.1 Common initial features for all scenarios

'Out of the factory' the device, for all three scenarios, is loaded with a set of Startup Attribute values defined by the commissioning cluster, known as an SAS (Startup Attribute Set). See [R4] for details of the commissioning cluster and all its attributes.

Table 10 shows the common set of initial attributes that are the same for all three scenarios.

1 **Table 10 – Startup Attribute Values – common set**

Attribute	Value	Notes
Channel Mask	All channels supported by the device	Join any channel
PANId	0xffff	This means that the device has not yet joined a network.
ProtocolVersion	0x02	ZigBee 2007
StackProfile	0x01 or 0x02	ZigBee or ZigBee PRO
NetworkManagerAddress	0x0000	This means that the Network Layer Function Manager address is by default that of the ZigBee Coordinator.
ShortAddress	0xffff	This means that the device has not been allocated a network address.
TrustCenterMasterKey	0 (128 bits long)	The Trust Center Master Key. This value indicates it is unspecified.
NetworkKeyType	0x01	0x01 means Standard Security mode. This allows interoperability with e.g. the HA profile.
NetworkKey	0 (128 bits long)	This value indicates that the network key is unspecified.
NetworkKeySeqNum	0x00	Default value.

2 ## 8.2 Service Provider scenario

3 ### 8.2.1 Actions before delivery

4 Before delivery to the customer, the service provider loads (or overwrites as applicable) the SAS
5 with the additional values given in

6 Table 11.

Table 11 – Startup Attribute Values –Service Provider scenario.

Attribute	Value	Notes
ExtendedPANId	Customer's network EPID	Note – if the customer has more than one network, several startup value sets may be needed.
TrustCenterAddress	The address of the customer's Trust Center	Note – if the customer has more than one network, several startup value sets may be needed. This may be set to unspecified (0xffffffffffffffff) if not known – the value will be communicated to the device when it joins the network.
PreconfiguredLinkKey	Chosen by service provider.	The Trust Center link key for the operational network. May be shared between a number of devices, though this is not recommended.
UseInsecureJoin	FALSE	The device shall not use insecure join as a fallback option.
StartupControl	0x02 (If the device is the Trust Center, this should be 0x01)	The device shall come "out-of-the-box" attempting to rejoin (or form) a network with characteristics given by this SAS.

In this scenario, the service provider knows the customer's network EPID and (optionally) the address of the Trust Center on this network, so prior to delivery he sets the relevant attributes accordingly.

The Trust Center link key is chosen by the service provider. The chosen value, together with the IEEE address of the corresponding node, is also communicated to the customer's Trust Center, e.g. by a secure internet connection or on a storage medium.

If the ASKE or ASAC cluster is used, and if the Trust Server is not co-located with the ZigBee Trust Center, the TS needs means to securely communicate with the device. In this case, prior to delivery the service provider configures the Security Domain table entry of the applicable cluster on the device with a Trust Server address and Trust Server Link Key (TS-LK), e.g. using the Configure SD command of the applicable cluster.

The TS-LK is then securely communicated to the customer's TS, together with IEEE address of the corresponding node.

The device is then delivered to the customer's site.

8.2.2 Commissioning phase

At the customer's site, no additional commissioning is performed; the device is just taken out of the box and turned on.

8.2.3 Joining the network

When the device is turned on, it scans for a network with the right EPID, joins it using a Network Rejoin command (see [R1]), and acquires a network short address. The Trust Center then sends it a Transport-Key command frame secured with the Trust Centre link key, and containing the network key. Note that in the APS auxiliary security header, the extended nonce subfield is set, and the source address field carries the IEEE address of the Trust Centre.

If the ASKE cluster is used, the Trust Server then uses a Trust Server link key to securely configure the joiner with an individual TS-assigned 16-bit identifier and ASKE keying material. Alternatively, if the ASAC cluster if used, the Trust Server then uses a Trust Server link key to securely configure the joiner with individual ASAC keying material, Lightweight Digital Certificate, and Access Control policies. Note – depending on the location of the Trust Server, this step may be carried out before delivery to the customer's site.

8.2.4 Application Pairing of devices

The device is now a member of the network. However, it is also necessary to pair it (at the application level) with the specific device(s) that it needs to communicate with.

A number of ways may be used to select one or more devices for a specific device to be paired with (note, these methods are also applicable to the other commissioning scenarios).

Some pairings are expected to remain in effect for a long period (days or weeks), such as between static devices. However, some pairings needed between devices can change frequently, e.g. in a medical center when a doctor makes checks on a number of patients in succession. For these cases the connection should be automatically terminated by the management device after each measurement session.

The ZDO and ZCL provide various facilities by which the Data Management device can discover agent devices. The following are examples:

1. A management device with a display based user interface may display information about all local devices and allow the user to choose the one(s) he wants, optionally using the Identify command of the Identify cluster (see [R4]) as a check.
2. A management device without a display based user interface may be paired with a sensing device by proximity pairing, i.e. performing a local search for high power devices, by holding it close to each device in turn and (e.g.) pressing a button on the management device.

It is recommended that devices that sleep for long periods should have a means of waking them up, such as a button.

When identifying devices to be paired with, the Match Protocol Address command of the generic tunnel cluster (see [R5] and A.1.2.1) may be used to find an appropriate device to pair with.

If the ASKE cluster is used, each such pair of devices then uses its ASKE keying material (see [R8]) to establish an end-to-end protected connection by generating a common application link key. Alternatively, if the ASAC cluster is used, each such pair of devices then uses its ASAC keying material and AC policies (see [R8]) to establish an end-to-end protected connection by generating a common application link key with authorized devices.

Otherwise, the devices being paired should contact the Trust Center using a request-key APS command to obtain an application link key for the device pair.

Once an agent device has been selected, the management device checks that the agent device has an 11073 tunnel server cluster (see A.1.2), reads its *ProtocolAddress* attribute, and communication is established at the 11073 protocol level as described in 8.2.4.1.

8.2.4.1 Connect Indication for 11073

The IEEE 11073 protocol requires that the agent is given a "connect indication" when the transport connection is established (see [R16]), after which it is up to the agent device to start the 11073 communication.

In ZigBee this is provided by the connect request command, and disconnect by the corresponding disconnect request command, see A.1.2.3.

For an agent sensor, the 11073 “connect indication” is inferred when a Data Management device connects to the agent’s ZHC endpoint. Similarly “disconnect indication” is inferred on receipt of a disconnect request command. Disconnect can also occur for other reasons.

When a connected device detects that it cannot make contact with the network, e.g. when it goes out of range, or when the target cannot be reached, e.g. when the APS sub-layer has reached the conclusion that the transmission has failed, a “disconnect indication” should be inferred. Similarly, when a device with an existing connection detects that its connection to the network has been restored (eg: when it comes back into range, a route has been reestablished to the target, or following a power cycle) then “connect indication” should be inferred.

8.3 In-house Commissioning scenario

8.3.1 Actions before delivery

In this scenario, the device is delivered to an in-house commissioning facility. Before delivery the device is loaded with a set of Startup Attribute values for use by the commissioning cluster (see [R4]). These values consist of the common set detailed in Table 10 and a set of specific values for this scenario, detailed in Table 12.

Table 12 – Startup Attribute Values (In-house Commissioning scenario) – as delivered

Attribute	Value	Notes
ExtendedPANId	0x0050c27710000000	The global commissioning EPID reserved by the ZigBee Alliance. See [R4].
TrustCenterAddress	0xffffffffffffffff	This means that the Trust Center address is not initially specified, but will be communicated to the device when it joins the network.
PreconfiguredLinkKey	0x5A 0x69 0x67 0x42 0x65 0x65 0x41 0x6C 0x6C 0x69 0x61 0x6E 0x63 0x65 0x30 0x39	The initial Trust Centre Link Key as specified by the Profile Interoperability policy. Note: The Link Key is listed in little-endian format.
UseInsecureJoin	TRUE	This setting enables the device to join any network (in an insecure way).
StartupControl	0x03	Indicates that the device should join the network using MAC association.

8.3.2 Commissioning phase

At the in-house commissioning facility, the device is turned on and it scans for the commissioning network, and joins it using MAC association. A commissioning tool connected to this network then sets the startup attributes, optionally including a unique ShortAddress (unique in the operational network), to suitable values for the target network as detailed in Table 13.

Table 13 – Startup Attribute Values (In-house Commissioning scenario) – after commissioning

Attribute	Value	Notes
ExtendedPANId	Unique operational EPID	-
TrustCenterAddress	The address of the customer's Trust Center	Note – if the customer has more than one network, several startup value sets may be needed. This may be set to unspecified (0xffffffffffffffff) if not known – the value will be communicated to the device when it joins the network.
PreconfiguredLinkKey	Unique TC-LK	The Trust Center Link Key for the operational network
UseInsecureJoin	FALSE	The device shall not use insecure join as a fallback option.
StartupControl	0x02 (If the device is the Trust Center, this should be 0x01)	The device shall come "out-of-the-box" attempting to rejoin (or form) a network with characteristics given by this SAS.
ShortAddress	Unique operational network value, or 0xffff	A value of 0xffff indicates unspecified

If the ASKE cluster is used, the commissioning tool also loads the node's individual CryptoIdentity (see [R8]), being a TS-assigned 16-bit identifier, and the ASKE keying material derived using that unique CryptoIdentity, including all related parameters and Trust Server address and Trust Server Link Key. Alternatively, if the ASAC cluster is used, the commissioning tool also loads the node's individual ASAC keying material, Lightweight Digital Certificate, and Access Control policies, and Trust Server address and Trust Server Link Key. Note – depending on the location of the Trust Server, some of these elements may be loaded after joining the operational network.

The commissioning network should be securely shielded, for example, in a screened room. The values may instead be set up by means of an out-of-band method, at the manufacturer's & customer's discretion.

The commissioning tool communicates the Trust Center link key to the Trust Center of the target network by an out-of-band method, and sends a Restart command to the commissioning cluster of the joiner.

8.3.3 Joining the network

The device proceeds to join the target network. It scans for a network with the right EPID, joins it using a Network Rejoin command, and acquires a unique network address. The Trust Center then sends it a Transport-Key command frame secured with the Trust Centre link key, and containing the network key. Note that in the APS auxiliary security header, the extended nonce subfield is set, and the source address field carries the IEEE address of the Trust Centre.

8.3.4 Application Pairing of devices

Pairing devices in this scenario may be done using any of the methods outlined in 8.2.4.

8.4 Consumer scenario

8.4.1 Actions before delivery

In this scenario, the device purchased by the customer is preloaded with a set of Startup Attribute values for use by the commissioning cluster (see [R4] for details of the commissioning cluster and all its attributes). These values consist of the common set detailed in Table 10 and a set of specific values for this scenario, detailed in Table 14.

Table 14 – Startup Attribute Values – Consumer scenario.

Attribute	Value	Notes
ExtendedPANId	0x0000000000000000	Unspecified.
TrustCenterAddress	0xffffffffffffffff	This means that the Trust Center address is not initially specified, but will be communicated to the device when it joins the network.
PreconfiguredLinkKey	0x5A 0x69 0x67 0x42 0x65 0x65 0x41 0x6C 0x6C 0x69 0x61 0x6E 0x63 0x65 0x30 0x39	The initial Trust Centre Link Key as specified by the Profile Interoperability policy. Note: The Link Key is listed in little-endian format.
UseInsecureJoin	TRUE	This setting enables the device to join any network (in an insecure way).
StartupControl	0x03	Indicates that the device should join the network using MAC association.

8.4.2 Commissioning phase

No commissioning is expected to be done by the consumer.

8.4.3 Joining the network

To join the network, the device is turned in range of the customer's network. The Trust Center of the network is instructed to allow joins for a short period (e.g. by pressing a button) and the device is instructed to join the network using MAC association (again e.g. by pressing a button, though the device may attempt to join as soon as it is turned on).

The device then acquires the EPID and a network address, and the Trust Center sends it a Transport-Key command frame secured with the default TC-LK, and containing the network key. Note that in the APS auxiliary security header, the extended nonce subfield is set, and the source address field carries the IEEE address of the Trust Centre.

The Trust Center should also replace the default TC-LK at the joining device, by sending a Transport-Key command frame secured with the default TC-LK, and containing the new (individual) TC-LK.

If the ASKE cluster is used, the Trust Server – co-located with the Trust Center and triggered by reception of the Update-Device command - uses the commands specified in [R8] (specifically, the Configure SD command) to securely load into the device a unique TS-assigned 16-bit identifier and ASKE keying material, derived using the unique TS-assigned 16-bit identifier as CryptoIdentity (see [R8]). Alternatively, if the ASAC cluster is used, the Trust Server uses the commands specified in [R8] to securely load the ASAC keying material, Lightweight Digital Certificate, and Access Control policies (specifically, the commands AC Properties Request/Response, Configure SD and Configure AC policies.

Note that this method has a short period of very limited security (initial communication protected via the default trust center link key), and care should be taken accordingly. This is a serious security weakness, and the network owner must balance this concern against the wish to add "off-the-shelf" devices to the network in a simple way. If feasible and the setup allows, it is strongly recommended to maximize security that both the joiner and the parent use very low power transmissions during the joining interval, so that the devices must be held close together.

8.4.4 Application Pairing of devices

Pairing devices in this scenario may be done using any of the methods outlined in 8.2.4.

8.5 Device indications

With respect to notification of installers and users, the following practices are specified. An HC data management device shall be able (all other HC devices should be able) to indicate to the user:

- That it is the coordinator of a network.
- That it has successfully joined a network.
- That it has failed to join a network.
- That a new connection with another device has been created.
- That it has failed to be connected to another device.
- That it is in the process of searching for or joining a network.
- That it is identifying itself (see the Identify cluster of the ZCL [R4])

These indications may be implemented in a number of ways including indicator light(s) or an audible indicator.

8.6 Application Layer Security

8.6.1 Introduction

The Alpha-Secure Key Establishment (ASKE) cluster and the Alpha-Secure Access Control (ASAC) cluster (see [R8]) allow for distributed and efficient application-layer security, without delay and potential single-point-of-failure unreliability introduced by the centralized Trust Center, thus fulfilling the requirements specified in section 7.1.1 of the Health Care TRD [R7].

Therefore, it is highly recommended that HC networks use ASKE or ASAC to establish application link keys, and, in the case of ASAC, to also configure and manage access control identities, roles and policies. Which of the clusters is used depends on the security requirements of the particular network.

Devices in an HC network may be configured to reject any HC commands (except ASKE or ASAC key establishment commands) they receive that are not encrypted and/or authenticated with an application link key. This includes ASKE or ASAC configuration commands, which are preferably protected by the Trust Server link key. To do this, it is recommended that the mechanism in the ZigBee specification r18 4.6.3.8 “ZigBee Permissions Configuration Table” (see [R1]) is used. Using this mechanism, application commands are conditionally accepted/rejected based on the value of the ApplicationCommands element (table 4.4.1 of [R1]).

The ASKE keying material, together with the related parameters, should be stored in a non-volatile memory, as it is may be used over long time periods and should persist across power failures and device resets.

8.6.2 Trust Server role

The Trust Server is an entity responsible for assignment and maintenance of the keying material required by the ASKE or ASAC cluster, including polynomial keying material shares, related parameters, access control policies, cryptographic identifiers, lightweight digital certificates and access control roles. The Trust Server can be located on the ZigBee Trust Center or a Commissioning Tool, but also on any other node. The Trust Server is not required to be always on or even to be part of the operational ZigBee network, since its main role is to initialize the devices with the keying material - this allows the devices to independently carry out distributed establishment of unique pairwise application link keys and, in case of ASAC, also authorization of the exchanged data.

The Trust Server is responsible for generation, secure storage and maintenance of the root keying material, used to generate the keying material shares that allow for key establishment (ASKE, ASAC) or verification of access control roles (ASAC).

During the initialization of a particular device, the Trust Server has the following tasks:

- Assuring uniqueness of the cryptographic identities used by the nodes
- Assignment of unique keying material shares, derived using the unique cryptographic identities and the root keying material, to the nodes

and in case of ASAC, additionally:

- Assignment of lightweight digital certificates, corresponding to node’s capabilities and roles
- Assignment of access control policies, corresponding to node’s privacy requirements.

During operation, the Trust Server may maintain the Revocation List for each Security Domain, and update the Revocation Lists stored by the devices, preferably by sending a unicast Update Revocation List command, protected with the TS-LK. This may be performed constantly or at selected time periods.[3]

8.6.3 Using the ASKE cluster

The ASKE cluster (see [R8]) allows efficient and distributed key agreement for establishment of Application Link Keys, as required for a number of HC use cases (see [R7]). Using the ASKE cluster, a pair of ZigBee devices belonging to the same ZigBee network can agree on a common Application Link Key without intervention of a centralized Trust Center. ZigBee devices carry keying material that may be generated and distributed to nodes before joining (e.g. by means of a commissioning tool) or at joining (e.g. by the Trust Center).

[3] Note that the Revocation List is only one of the possible methods of keeping the revoked nodes out of the network. Other methods include updating security domain keying material and/or triggering secure network key update. The method choice, as well as exact values and timing are part of Trust Server policy and thus out of scope for this document.

Note - The following subsections assume some familiarity with the details of the ASKE cluster. It is recommended that the reader familiarize himself/herself with the material in [R8].. For proper selection of the parameters of the ASKE cluster, see [R14].

8.6.3.1 ASKE support by HC devices

For networks employing ASKE, the following requirements shall be satisfied.

Devices implementing Trust Server role (e.g. the Trust Centre or Commissioning Tool):

- Shall implement generation of the configuration commands (client side), if in-band configuration of the ASKE Security Domain table entries is desirable;
- May implement the generation and reception of key establishment commands (i.e. client and server side).

All other HC devices:

- Shall not implement generation of the configuration commands (client side),
- Shall implement reception of the configuration commands (i.e. server side), if in-band configuration of the ASKE Security Domain table entries is desirable;
- Shall implement the generation and reception of key establishment commands (i.e. client and server side).

8.6.3.2 ASKE Key agreement

The ASKE key agreement handshake may be carried out according to two different procedures as described in [R8]. On the one hand, the ASKE may be performed as a part of the symmetric-key key exchange (SKKE) protocol, if SKKE is supported by the required stack profile or application profile. On the other hand, the ASKE key agreement handshake may be implemented at the application layer by means of four new messages. Both approaches are functionally identical, i.e. enable generation of unique pairwise Application Link Keys (ALKs) between the Initiator and the Responder nodes.

As this profile employs Standard Security, one would not be able to rely on support of SKKE by the lower layers, so it is mandatory for ZHC devices implementing the ASKE cluster to implement the application-layer key exchange protocol. Thus, all the commands listed in the Mandatory Commands for Application Layer ASKE Key Agreement table of [R8] are mandatory.

To achieve interoperability, the implementation of the finite field $F_{q'}$, where *q'* is equal to 2^{16} (see [R8]), is mandatory.

8.6.3.3 Configuration of ASKE

ASKE is intended for applications requiring strong and efficient application layer security. Thus, to enable ASKE to fulfill its role of establishing unique pairwise keys for application layer communication, the keying material has to be distributed or installed in a secure way as well, guaranteeing its secrecy.

For configuration of ASKE Security Domain table entries, usage of a secure channel (out-of-band or in-band, e.g. protected via Trust Server key) or a trusted, isolated environment is recommended.

8.6.3.4 Choosing security level

Depending on the security requirements and network size of a particular medical installation, keying material configurations can be chosen to assure optimal performance and resource utilization. Security requirements include i.a. the key length and assumptions on the number of compromised nodes. Knowing the to-be-supported network size and the number of nodes that have been revoked and that could be compromised without being detected allows for deriving the system resiliency parameters, keying material parameters and keying material update policies. Note that the system resiliency does impact the memory requirements for the keying material, and that it is recommended to provide appropriate resources for a required security level (and not vice versa). For detailed recommendations and guidelines on keying material sizes as well as exemplary parameter combinations, see [R14].

Common ASKE security parameters for the entire profile are not specified, because of the varying sizes and security requirements of the medical sensor networks to be supported by the HC profile, ranging from few to thousands of devices, and from fitness/wellness applications to hospital monitoring. The parameter configuration is dependent on the to-be-supported scenario, and as such is part of Trust Server policy (out of scope for the current document).

8.6.3.5 Key update policies

For HC networks, frequent key updates are recommended. For the Keying Material, the update frequency depends mainly on the risk that devices are lost and their Keying Material extracted, as well as on the parameters choice for the ASKE algorithm. As those are highly use case specific, a more detailed discussion is out of scope for this document and is provided in [R14]. The Trust Server controls the update of the Application Link Key derived form the keying material, by setting the *ALKUpdateSchedule* field of the Configure SD command to the appropriate value.

8.6.4 Using the ASAC cluster

The ASAC cluster (see [R8]) allows efficient distributed key agreement for establishment of Application Link Keys. Based on the ASAC, a pair of ZigBee devices belonging to the same ZigBee network can agree on a common Application Link Key without intervention of a centralized Trust Center. ZigBee devices carry keying material that may be pre-distributed (e.g. at the factory) or generated and distributed to nodes before joining (e.g., by means of a commissioning tool) or at joining (e.g. by the Trust Center).

In addition, ASAC enables configuration and management of access control identities, roles and policies. For example, a ‘doctor’ role can be set up such that data from a particular medical device may only be read by a data management device that is set up with that role. See [R8] for more details on this example.

Note - The following subsections assume some familiarity with the details of the ASAC cluster. It is recommended that the reader familiarize himself/herself with the material in [R8].

8.6.4.1 ASAC support by devices

For networks employing ASAC, the following requirements shall be satisfied.

Trust Server devices (e.g. the Trust Centre or Commissioning Tool):

- Shall implement generation of the configuration commands (client side) if they are to support in-band configuration of the ASAC information;
- May implement the generation and reception of access control commands (i.e. client and server side).

All other HC devices:

- Shall not implement generation of the configuration commands (client side),

- Shall implement reception of the configuration commands (i.e. server side), IF in-band configuration of the ASAC table entries is desirable;
- Shall implement the generation and reception of access control commands (i.e. client and server side).

8.6.4.2 Configuration of ASAC

The security level for underlying network security is out of scope for this cluster. However, for in-band configuration of ASAC Security Domain table entries, a security level corresponding to the security requirements of the intended application should be provided (e.g. for very secure keying material configuration a secure channel protected via Trust Server Link Keys or an isolated environment should be used).

8.6.4.3 ASAC Key agreement

The ASAC key agreement handshake may be carried out according to two different procedures as described in [R8]. On the one hand, the ASAC may be performed as a part of the symmetric-key key exchange (SKKE) protocol if SKKE is supported by the required stack profile or application profile. On the other hand, the ASAC key agreement handshake may be implemented at the application layer by means of four new messages. Both approaches are functionally identical, i.e. enable generation of unique pairwise Application Link Keys (ALKs) between the Initiator and the Responder nodes.

As this profile employs Standard Security, one would not be able to rely on support of SKKE by the lower layers, so it is mandatory for HC devices implementing the ASAC cluster to implement the application-layer key exchange protocol. Thus, all the commands listed in the Mandatory Commands for Application Layer ASAC Key Agreement table of [R8] are mandatory.

8.6.4.4 Choosing ASAC Keying Material parameters

Selection of the proper Keying Material parameters is the task of the Trust Server and has to be performed with care, considering the required security level of particular medical application and site; for detailed guidelines on that selection and example configurations, see [R14].

To achieve interoperability, the implementation of the finite field $F_{q'}$, where q' is equal to 2^{32} (see [R8]) is mandatory. No other ASAC security parameters common for the entire profile are specified, because of the varying sizes and security requirements of the medical sensor networks to be supported by the Health Care profile, ranging from few to thousands of devices, and from fitness/wellness applications to hospital monitoring. The parameter configuration is dependent on the to-be-supported scenario, and as such part of Trust Server policy (out of scope for the current document).

8.6.4.5 Keying material update policy

For HC networks, frequent key updates are recommended. For the Keying Material, the update frequency depends mainly on the risk that devices are lost and their Keying Material extracted, as well as on the parameters choice for the ASKE algorithm. As those are highly use case specific, a more detailed discussion is out of scope for this document and is provided in [R14]. The Trust Server controls the update of the Application Link Key derived form the keying material, by setting the *ALKUpdateSchedule* field of the Configure SD command to the appropriate value.

8.6.4.6 Lightweight Digital Certificate properties

Table 15 below lists the HC-profile-specific LDC properties.

1 **Table 15 – HC-specific LDC property types**

ID	Property Name	Property Type	Range	Default	Description
0x41	11073SystemID	Unsigned 64-bit integer	Any valid	N/A	The SystemID as defined by the 11073 application protocol
0x42	11073ConfigID	16-bit enumeration	Any valid	N/A	The ConfigID as defined by the 11073 application protocol
0x43 - 0x80	Reserved (for further HC-specific LDC properties)	-	-	-	-

2 Note: the values 0x00-0x40 and 0x81-0xff are defined in the ASAC cluster itself ([R8]).

9 Candidate ZCL Material for use with this Profile

The candidate material in the following annex and in [R8], when approved, may be merged into the ZigBee Cluster Library (ZCL) by the Cluster Library Development Board.

The ASKE and ASAC clusters are specified in Part 2 of this profile specification [R8] for the following reasons

- Separation allows the main specification to be more compact and clear.
- Though highly recommended, the clusters are optional for the Health Care profile.
- The clusters are not specific to the Health Care Profile, but are designed to be of general use for any ZigBee application that has similar security requirements to those of the Health Care Profile.

The new clusters to be included in the ZCL have been provisionally allocated the cluster IDs indicated in Table 16.

Table 16 – Provisional Clusters ID allocation for Candidate clusters

Functional Domain	Cluster Name	Provisional ClusterID	Where specified
Protocol Interfaces	11073 Protocol Tunnel	0x0614	A.1
General	Alpha-Secure Key Establishment (ASKE)	0x0017	[R8]
General	Alpha-Secure Access Control (ASAC)	0x0018	[R8]

A.1 11073 Protocol Tunnel cluster

A.1.1 Overview

The 11073 Protocol Tunnel cluster provides the commands and attributes required to tunnel the 11073 protocol. The server cluster receives 11073 APDUs and the client cluster generates 11073 APDUs, thus it is necessary to have both server and client on an endpoint to tunnel 11073 messages in both directions.

Commands and attributes are provided for establishing, querying the status of, and removing an 11073 tunnel connection between two devices.

Devices that support this cluster shall also comply with the ISO/IEEE 11073-20601 standard for Personal Health Device Communication [R16] and the applicable ISO/IEEE 11073 device specialization documents [R17] - [R24].

Typical usage of the 11073 Protocol Tunnel cluster is illustrated in **Figure 1**.

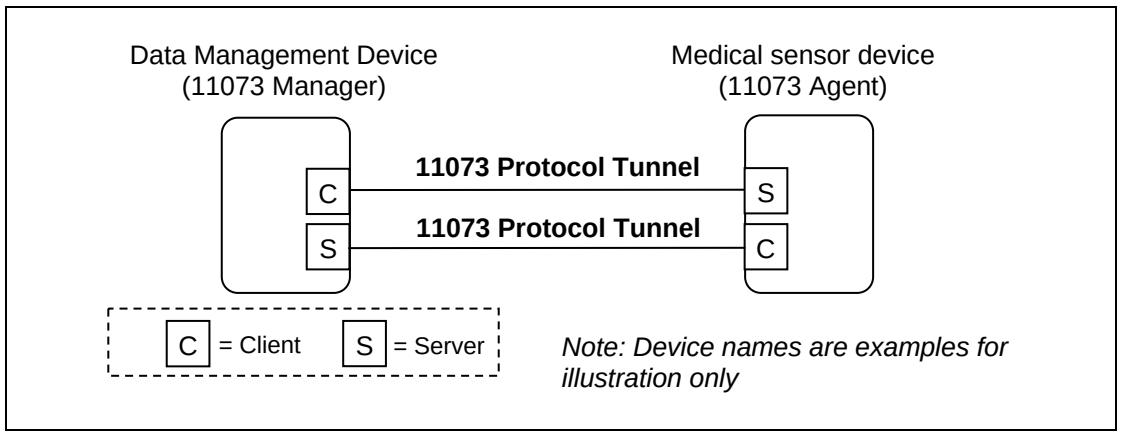

Figure 1 Typical Usage of the 11073 Protocol Tunnel cluster

Note that all 11073 protocol tunnel cluster specific commands are generated by the client and received by the server. A typical sequence of events to initiate an 11073 interaction might be:

- DMD transmits Connect Request command to sensor (client→server)
- Sensor responds with a Connect Status Notification command with status CONNECTED (client→server)
- Sensor and DMD carry out an 11073 layer interaction by exchanging Transfer APDU commands in each direction (client→server in each case)

A.1.2 Server

A.1.2.1 Dependencies

Any endpoint that supports the 11073 Protocol Tunnel server cluster shall also support the Generic Tunnel server cluster (see [R5]).

The value of the *ProtocolAddress* attribute of the associated Generic Tunnel server cluster shall be set equal to the system ID of the 11073 device represented on that endpoint (see [R16]). The system ID, represented as a ZigBee octet string, shall be 8 octets in Big Endian order.

The value of the *MaximumIncomingTransferSize* attribute and the value of the *MaximumOutgoingTransferSize* attribute of the associated Generic Tunnel server cluster shall be set equal to or greater than the maximum APDU size specified in the applicable ISO/IEEE 11073 device specialization document ([R17] - [R24]).

A.1.2.2 Attributes

The 11073 Protocol Tunnel server cluster contains the attributes shown in Table 17.

Table 17 – Attributes of the 11073 Protocol Tunnel server cluster

ID	Name	Type	Range	Access	Default	M/O	Description
0x0000	DeviceIDList	array of unsigned 16-bit integer	Any valid	Read only	0xffff	O	List of devices supported behind the 11073 tunnel. Note that this attribute is mandatory for multifunction devices as defined in section 7.5.
0x0001	Manager target	IEEE address	Any valid IEEE address	Read only	-	O	Address of the currently or most recently connected Data Management device.
0x0002	Manager endpoint	Unsigned 8-bit integer	0x01-0xff	Read only	-	O	Endpoint of the currently or most recently connected Data Management device.
0x0003	Connected	Boolean	TRUE / FALSE	Read only	FALSE	O	Whether this endpoint is currently connected
0x0004	Preemptible	Boolean	TRUE / FALSE	Read only	TRUE	O	Whether this connection can be overridden
0x0005	Idle timeout	Unsigned 16-bit integer	0x0001 – 0xffff	Read only	0x0000	O	Number of idle minutes before automatic disconnection. 0xffff = never
0x0006-0xffff	Reserved	-	-	-	-	-	-

Although the *manager target*, *manager endpoint*, *connected*, *preemptible* and *idle timeout* attributes are listed above as optional, if any one of them is implemented then all of them shall be implemented, and also reception of the connect and disconnect request commands shall be implemented, and the 11073 Protocol Tunnel client cluster implemented on the same endpoint shall implement transmission of the connect status notification command.

A.1.2.2.1 DeviceIDList attribute

The *DeviceIDList* attribute specifies all devices supported behind a single instance of the11073 tunnel, on a single endpoint. It allows for discovering the functionality of 11073 devices on a ZigBee level, e.g. prior to establishment of an 11073 tunnel. The *DeviceIDList* attribute can be read using generic ZCL commands.

For a multifunction device as defined in section 7.5, the *DeviceIDList* attribute is mandatory and shall contain a complete list of supported ZigBee Device IDs (as defined in Table 2) supported by the single instance of the *11073 Protocol Tunnel* on this particular end point. The DeviceID contained in the Simple Descriptor of the multifunction sensor shall contain the value corresponding to the multifunction device itself (see Table 2).

For all other devices types, the *DeviceIDList* attribute is optional. If implemented, it shall contain the single ZigBee DeviceID allocated to that device (see Table 2).

A.1.2.2.2 Manager target attribute

The *Manager Target* attribute specifies the IEEE address of the currently or most recently connected Data Management device.

A.1.2.2.3 Manager endpoint attribute

The *Manager Endpoint* attribute specifies the endpoint used by the currently or most recently connected Data Management device.

A.1.2.2.4 Connected attribute

The *Connected* attribute specifies whether or not the 11073 tunnel on this endpoint is currently connected.

If this attribute takes the value TRUE, then the tunnel is currently connected.

If this attribute takes the value FALSE, then the tunnel is not currently connected.

Whenever the value of this attribute changes the 11073 layer shall be informed via the transport connected and transport disconnected indications.

A.1.2.2.5 Preemptible attribute

The *Preemptible* attribute specifies whether or not the current connection can be disconnected by a Data Management device other than by the one currently connected.

If this attribute takes the value TRUE, then a disconnect request from a device other than the Data Management device indicated by the *manager target* attribute shall be accepted and if the 11073 tunnel on this endpoint is currently connected then it shall become disconnected, and a connect status notification command with status DISCONNECTED shall be sent to the currently connected Data Management device.

If this attribute takes the value FALSE, then a disconnect request from a device other than the Data Management device indicated by the *manager target* attribute shall be rejected and a connect status notification command with status NOT_AUTHORIZED sent to the requester.

A.1.2.2.6 Idle timeout attribute

The *Idle Timeout* attribute specifies the inactivity time in minutes which the Data Management device will wait without transmitting or receiving any tunneled frames to or from the connected target, before it disconnects the connection.

If the Data Management device does not intend to timeout this connection after a specific idle period then this attribute shall take the value 0xffff.

If the indicated timeout period passes with no data on the 11073 tunnel, then the agent device shall set its *connected* attribute to FALSE and a connect status notification command with status DISCONNECTED shall be sent to the currently connected Data Management device. In order to continue to use the tunnel, the agent device shall send the Data Management device a further connect status notification command with status RECONNECT_REQUEST, and wait for the Data Management device to respond.

A.1.2.3 Commands Received

The cluster specific commands received by the 11073 Protocol Tunnel server cluster are listed in Table 18.

Table 18 – Command IDs for the 11073 protocol tunnel cluster

Command identifier field value	Description	Mandatory/Optional
0x00	Transfer APDU	M
0x01	Connect request	O
0x02	Disconnect request	O
0x03	Connect status notification	O
0x04 – 0xff	Reserved	-

Although the connect request and disconnect commands are listed above as optional, if reception of either of them is implemented then reception of both of them shall be implemented, and also the *manager target*, *manager endpoint*, *connected*, *preemptible* and *idle timeout* attributes shall be implemented, and the 11073 Protocol Tunnel client cluster implemented on the same endpoint shall implement transmission of the connect status notification command.

Although reception of the connect status notification command is listed above as optional, if reception of this command is implemented then also the connect request command shall be implemented by the 11073 Protocol Tunnel server on the same endpoint.

A.1.2.3.1 Transfer APDU Command

The Transfer APDU command payload shall be formatted as illustrated in Figure 2.

Bits	Variable
Data Type	long octet string
Field Name	APDU

Figure 2 – Transfer APDU payload

The APDU field is of variable length and is a 11073 APDU as defined in the ISO/IEEE 11073 standard [R16].

A.1.2.3.1.1 When generated

This command is generated when an 11073 network layer wishes to transfer an 11073 APDU across a ZigBee tunnel to another 11073 network layer.

The most stringent reliability characteristic of a given transport technology is “Best” reliability. Note - For ZigBee, this corresponds to use of APS-ACKs.

The least stringent reliability characteristic of a given transport technology is “Good” reliability. Note - For ZigBee, this corresponds to no use of APS-ACKs.

The application is responsible for transmitting at a reliability level appropriate for each frame.

This command shall always be transmitted with the disable default response bit in the ZCL frame control field set to 1.

A.1.2.3.1.2 Effect on Receipt

On receipt of this command, a device shall process the 11073 APDU as specified in [R16] and the applicable device specialization [R17] to [R27]

A.1.2.3.2 Connect Request Command

The Connect Request command payload shall be formatted as illustrated in Figure 2.

Octets	1	2	8	1
Data Type	8-bit bitmap	16-bit unsigned integer	IEEE Address	8-bit unsigned integer
Field Name	Connect control	Idle timeout	Manager target	Manager endpoint

Figure 3 – Connect Request command payload

A.1.2.3.2.1 Connect control

The *connect control* field shall be formatted as illustrated in Figure 3.

Bit	0	1-7
Field Name	Preemptible	Reserved

Figure 4 – Connect control field format

The *Preemptible* bit shall indicate whether or not this connection can be removed by a different Data Management device.

A.1.2.3.2.2 Idle timeout

The *idle timeout* field shall indicate the inactivity time in minutes which the Data Management device will wait without receiving any tunneled frames from the connected target, before it disconnects the connection.

A.1.2.3.2.3 Manager target

The *Manager target* field shall indicate the IEEE address of the Data Management device transmitting this frame.

A.1.2.3.2.4 Manager endpoint

The *Manager endpoint* field shall indicate the source endpoint from which the Data Management device is transmitting this frame.

A.1.2.3.2.5 When generated

This command is generated when a Data Management device wishes to connect to an 11073 agent device.

This may be in response to receiving a connect status notification command from that agent device with the connect status field set to RECONNECT_REQUEST.

A.1.2.3.2.6 Effect on Receipt

On receipt of this command, a device shall first check if it is already connected by examining its *connected* attribute.

If the tunnel is already connected then the device shall generate a connect status notification command with status set to ALREADY_CONNECTED and transmit it to the sender of this connect request frame. No other attributes shall be affected, and no further processing shall be carried out.

If the tunnel is not currently connected then the device shall copy the preemptible bit of connect control field into the preemptible attribute, the idle timeout value into the idle timeout attribute, the manager target value into the *manager target* attribute and the manager endpoint value into the *manager endpoint* attribute.

It shall set the connected attribute to TRUE, and generate a connect status notification command with status set to CONNECTED and transmit it to the sender of this connect request frame.

Finally, if the idle timeout field is set to a value other than 0xffff, the device shall set a timer for the timeout time indicated. This timer shall be restarted at any time that data is transmitted or received over the tunnel. If the timer expires then the device shall set the *connected* attribute to FALSE and a connect status notification command with status DISCONNECTED shall be sent to the currently connected Data Management device. In order to continue to use the tunnel, the agent device shall send the Data Management device a further connect status notification command with status RECONNECT_REQUEST, and wait for the Data Management device to respond.

A.1.2.3.3 Disconnect Request Command

The Disconnect Request command payload shall be formatted as illustrated in Figure 4.

Octets	8
Data Type	IEEE Address
Field Name	Manager IEEE address

Figure 5 – Disconnect Request command payload

A.1.2.3.3.1 Manager IEEE address

The *Manager IEEE address* field shall indicate the IEEE address of the Data Management device transmitting this frame.

A.1.2.3.3.2 When generated

This command is generated when a Data Management device wishes to disconnect a tunnel connection existing on an agent device.

A.1.2.3.3.3 Effect on Receipt

On receipt of this command, a device shall first check if it is already connected by examining its *connected* attribute.

If it is not currently connected then the device shall generate a connect status notification command with status set to DISCONNECTED and transmit it to the sender of this disconnect request frame. No other attributes shall be affected, and no further processing shall be carried out.

If it is currently connected then the device shall check whether the requesting device is authorized to remove this connection. A device is authorized to remove the connection if the value of the manager IEEE address field is the same as the value in the *manager target* attribute or if the *preemptible* attribute is set to TRUE.

If the requester is not authorized then the device shall generate a connect status notification command with status set to NOT_AUTHORIZED and transmit it to the sender of this disconnect request frame. No other attributes shall be affected, and no further processing shall be carried out.

If the requester is authorized then the device shall initiate disconnection. A short period of time is permitted in order to allow the higher layer to finalize its activities, but within 12 seconds the device shall generate a connect status notification command with status set to DISCONNECTED and transmit it to the target indicated in the *manager target* attribute. The *connected* attribute shall be set to FALSE and the tunnel shall be disconnected. The device shall now generate a further connect status notification command with status set to DISCONNECTED and transmit it to the sender of this disconnect request frame,

A.1.2.3.4 Connect Status Notification Command

The Connect Status Notification command payload shall be formatted as illustrated in Figure 5.

Octets	1
Data Type	8-bit enumeration
Field Name	Connect status

Figure 6 – Connect Status Notification command payload

A.1.2.3.4.1 Connect Status

The *connect status* field shall be set to one of the values in Table 19

Table 19 – Connect status values

Value	Designation	Description
0x00	DISCONNECTED	Indicates that this agent device has been disconnected from the tunnel.
0x01	CONNECTED	Indicates that this agent device has been connected to the tunnel.
0x02	NOT_AUTHORIZED	Indicates that a request to disconnect the tunnel is not authorized from this requester at this time.
0x03	RECONNECT_REQUEST	Indicates that the agent device wishes the Data Management device to reconnect the tunnel.
0x04	ALREADY_CONNECTED	Indicates that the request to connect this tunnel has failed as the agent device is already connected.

A.1.2.3.4.2 When generated

This command is generated by an agent device in response to a connect request command, disconnect command, or in response to some other event that causes the tunnel to become connected or disconnected.

It is also sent by the agent device to request the Data Management device to reconnect a tunnel.

A.1.2.3.4.3 Effect on Receipt

On receipt of this command, a device shall be informed of the new status of the tunnel connection or of its attempt to modify the status of the connection.

If the connect status field takes the value RECONNECT_REQUEST then, depending on available resources being available, the Data Management device should attempt to reconnect the tunnel by generating a connect request command and transmitting it to the agent device sending this connect status notification command.

A.1.2.4 Commands Generated

No cluster specific commands are generated by the server cluster.

A.1.3 Client

A.1.3.1 Dependencies

Any endpoint that supports the 11073 Protocol Tunnel client cluster shall also support the Generic Tunnel client cluster (see [R5]).

A.1.3.2 Attributes

The client cluster has no attributes.

A.1.3.3 Commands Received

The client does not receive any cluster specific commands.

A.1.3.4 Commands Generated

The cluster specific commands generated by the client cluster are listed in A.1.2.3.

In order to reduce the burden on implementations, some commands and attributes are conditionally mandated, as follows:

- Transmission of the transfer APDU command is mandatory.
- Transmission of the connect request and disconnect request commands is optional unless specified otherwise.
- If the 11073 Protocol Tunnel server cluster implemented on the same endpoint implements any of the *manager target*, *manager endpoint*, *connected*, *preemptible* and *idle timeout* attributes, or implements reception of the connect request or disconnect request commands, then transmission of the connect status notification command is mandatory.
- Transmission of non-cluster specific commands to manipulate attributes is optional unless specified otherwise.

1