NOTICE

This document is not a transfer of work into oneM2M.

This document is strictly for use as resource material for oneM2M on the conditions contained in the included NOTICE OF DISCLAIMER AND LIMITATION OF LIABILITY.

For more information, contact:

TIA Standards
703-907-7700
standards@tiaonline.org

NOTICE OF DISCLAIMER AND LIMITATION OF LIABILITY

The document to which this Notice is affixed has been prepared by one, or more, Engineering Committees of the Telecommunications Industry Association ("TIA"). TIA is not the author of the document contents, but publishes and claims copyright to the document pursuant to licenses and permission granted by the authors of the contents.

TIA Engineering Committees are expected to conduct their affairs in accordance with the TIA Engineering Manual ("Manual"), the current and predecessor versions of which are available at http://www.tiaonline.org/standards/sfg/engineering_manual.cfm. TIA's function is to administer the process, but not the content, of document preparation in accordance with the Manual and, when appropriate, the policies and procedures of the American National Standards Institute ("ANSI").

THE USE OR PRACTICE OF CONTENTS OF THIS DOCUMENT MAY INVOLVE THE USE OF INTELLECTUAL PROPERTY RIGHTS ("IPR"), INCLUDING PENDING OR ISSUED PATENTS, OR COPYRIGHTS, OWNED BY ONE OR MORE PARTIES. TIA MAKES NO SEARCH OR INVESTIGATION FOR IPR. WHEN IPR CONSISTING OF PATENTS AND PUBLISHED PATENT APPLICATIONS ARE CLAIMED AND CALLED TO TIA'S ATTENTION, A STATEMENT FROM THE HOLDER THEREOF IS REQUESTED, ALL IN ACCORDANCE WITH THE MANUAL. TIA TAKES NO POSITION WITH REFERENCE TO, AND DISCLAIMS ANY OBLIGATION TO INVESTIGATE OR INQUIRE INTO, THE SCOPE OR VALIDITY OF ANY CLAIMS OF IPR. ALL WARRANTIES, EXPRESS OR IMPLIED, ARE DISCLAIMED, INCLUDING WITHOUT LIMITATION, ANY AND ALL WARRANTIES CONCERNING THE ACCURACY OF THE CONTENTS, ITS FITNESS OR APPROPRIATENESS FOR A PARTICULAR PURPOSE OR USE, ITS MERCHANTABILITY AND ITS NON-INFRINGEMENT OF ANY THIRD PARTY'S INTELLECTUAL PROPERTY RIGHTS. TIA EXPRESSLY DISCLAIMS ANY AND ALL RESPONSIBILITIES FOR THE ACCURACY OF THE CONTENTS AND MAKES NO REPRESENTATIONS OR WARRANTIES REGARDING THE CONTENT'S COMPLIANCE WITH ANY APPLICABLE STATUTE, RULE OR REGULATION. TIA SHALL NOT BE LIABLE FOR ANY AND ALL DAMAGES, DIRECT OR INDIRECT, ARISING FROM OR RELATING TO ANY USE OF THE CONTENTS CONTAINED HEREIN, INCLUDING WITHOUT LIMITATION ANY AND ALL INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS, LOSS OF PROFITS, LITIGATION, OR THE LIKE), WHETHER BASED UPON BREACH OF CONTRACT, BREACH OF WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCT LIABILITY OR OTHERWISE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE FOREGOING NEGATION OF DAMAGES IS A FUNDAMENTAL ELEMENT OF THE USE OF THE CONTENTS HEREOF, AND THESE CONTENTS WOULD NOT BE PUBLISHED BY TIA WITHOUT SUCH LIMITATIONS.

ADVANCING GLOBAL COMMUNICATIONS

TIA STANDARD

Smart Device Communications Reference Architecture

TIA-4940.005

December 2011

TELECOMMUNICATIONS INDUSTRY ASSOCIATION

tiaonline.org

NOTICE

TIA Engineering Standards and Publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for their particular need. The existence of such Standards and Publications shall not in any respect preclude any member or non-member of TIA from manufacturing or selling products not conforming to such Standards and Publications. Neither shall the existence of such Standards and Publications preclude their voluntary use by Non-TIA members, either domestically or internationally.

Standards and Publications are adopted by TIA in accordance with the American National Standards Institute (ANSI) patent policy. By such action, TIA does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the Standard or Publication.

This Standard does not purport to address all safety problems associated with its use or all applicable regulatory requirements. It is the responsibility of the user of this Standard to establish appropriate safety and health practices and to determine the applicability of regulatory limitations before its use.

(From Project No. TIA-PN-4940.005, formulated under the cognizance of the TIA TR-50 Smart Device Communications TR-50.1 Subcommittee on Smart Device Communications; Requirements and Architecture).

Published by
©TELECOMMUNICATIONS INDUSTRY ASSOCIATION
Standards and Technology Department
2500 Wilson Boulevard
Arlington, VA 22201 U.S.A.

PRICE: Please refer to current Catalog of
TIA TELECOMMUNICATIONS INDUSTRY ASSOCIATION STANDARDS
AND ENGINEERING PUBLICATIONS
or call IHS, USA and Canada
(1-877-413-5187) International (303-397-2896)
or search online at http://www.tiaonline.org/standards/catalog/

NOTICE OF DISCLAIMER AND LIMITATION OF LIABILITY

The document to which this Notice is affixed (the "Document") has been prepared by one or more Engineering Committees or Formulating Groups of the Telecommunications Industry Association ("TIA"). TIA is not the author of the Document contents, but publishes and claims copyright to the Document pursuant to licenses and permission granted by the authors of the contents.

TIA Engineering Committees and Formulating Groups are expected to conduct their affairs in accordance with the TIA Engineering Manual ("Manual"), the current and predecessor versions of which are available at http://www.tiaonline.org/standards/procedures/manuals/TIA's function is to administer the process, but not the content, of document preparation in accordance with the Manual and, when appropriate, the policies and procedures of the American National Standards Institute ("ANSI"). TIA does not evaluate, test, verify or investigate the information, accuracy, soundness, or credibility of the contents of the Document. In publishing the Document, TIA disclaims any undertaking to perform any duty owed to or for anyone.

If the Document is identified or marked as a project number (PN) document, or as a standards proposal (SP) document, persons or parties reading or in any way interested in the Document are cautioned that: (a) the Document is a proposal; (b) there is no assurance that the Document will be approved by any Committee of TIA or any other body in its present or any other form; (c) the Document may be amended, modified or changed in the standards development or any editing process.

The use or practice of contents of this Document may involve the use of intellectual property rights ("IPR"), including pending or issued patents, or copyrights, owned by one or more parties. TIA makes no search or investigation for IPR. When IPR consisting of patents and published pending patent applications are claimed and called to TIA's attention, a statement from the holder thereof is requested, all in accordance with the Manual. TIA takes no position with reference to, and disclaims any obligation to investigate or inquire into, the scope or validity of any claims of IPR. TIA will neither be a party to discussions of any licensing terms or conditions, which are instead left to the parties involved, nor will TIA opine or judge whether proposed licensing terms or conditions are reasonable or non-discriminatory. TIA does not warrant or represent that procedures or practices suggested or provided in the Manual have been complied with as respects the Document or its contents.

If the Document contains one or more Normative References to a document published by another organization ("other SSO") engaged in the formulation, development or publication of standards (whether designated as a standard, specification, recommendation or otherwise), whether such reference consists of mandatory, alternate or optional elements (as defined in the TIA Engineering Manual, 4th edition) then (i) TIA disclaims any duty or obligation to search or investigate the records of any other SSO for IPR or letters of assurance relating to any such Normative Reference; (ii) TIA's policy of encouragement of voluntary disclosure (see Engineering Manual Section 6.5.1) of Essential Patent(s) and published pending patent applications shall apply; and (iii) Information as to claims of IPR in the records or publications of the other SSO shall not constitute identification to TIA of a claim of Essential Patent(s) or published pending patent applications.

TIA does not enforce or monitor compliance with the contents of the Document. TIA does not certify, inspect, test or otherwise investigate products, designs or services or any claims of compliance with the contents of the Document.

ALL WARRANTIES, EXPRESS OR IMPLIED, ARE DISCLAIMED, INCLUDING WITHOUT LIMITATION, ANY AND ALL WARRANTIES CONCERNING THE ACCURACY OF THE CONTENTS, ITS FITNESS OR APPROPRIATENESS FOR A PARTICULAR PURPOSE OR USE, ITS MERCHANTABILITY AND ITS NONINFRINGEMENT OF ANY THIRD PARTY'S INTELLECTUAL PROPERTY RIGHTS. TIA EXPRESSLY DISCLAIMS ANY AND ALL RESPONSIBILITIES FOR THE ACCURACY OF THE CONTENTS AND MAKES NO REPRESENTATIONS OR WARRANTIES REGARDING THE CONTENT'S COMPLIANCE WITH ANY APPLICABLE STATUTE, RULE OR REGULATION, OR THE SAFETY OR HEALTH EFFECTS OF THE CONTENTS OR ANY PRODUCT OR SERVICE REFERRED TO IN THE DOCUMENT OR PRODUCED OR RENDERED TO COMPLY WITH THE CONTNTS.

TIA SHALL NOT BE LIABLE FOR ANY AND ALL DAMAGES, DIRECT OR INDIRECT, ARISING FROM OR RELATING TO ANY USE OF THE CONTENTS CONTAINED HEREIN, INCLUDING WITHOUT LIMITATION ANY AND ALL INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS, LOSS OF PROFITS, LITIGATION, OR THE LIKE), WHETHER BASED UPON BREACH OF CONTRACT, BREACH OF WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCT LIABILITY OR OTHERWISE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE FOREGOING NEGATION OF DAMAGES IS A FUNDAMENTAL ELEMENT OF THE USE OF THE CONTENTS HEREOF, AND THESE CONTENTS WOULD NOT BE PUBLISHED BY TIA WITHOUT SUCH LIMITATIONS.

Contents

List of Figures

Foreword

(This foreword is not part of this Standard.)

This document was formulated under the cognizance of the TIA Subcommittee TR-50·1, Smart Device Communications; Requirements and Architecture.

The contents of the present document are subject to continuing work within the Formulating Group and may change following formal approval. Should the Formulating Group approve modification, the present document will be re-released with an identifying change of release level, for example:

TIA-4940.005-A

- A — revision level
- 005 — part number
- 4940 — standard number

The document contains informative annexes.

Suggestions for improvement of this document are welcome, and should be sent to:

Telecommunications Industry Association,
Standards and Technology,
2500 Wilson Boulevard, Suite 300
Arlington, VA 22201-3834

Scope

This document is a member of a multi-part standard that, when taken in total, defines the requirements for communications pertaining to the access agnostic (e.g. PHY and MAC agnostic) monitoring and bi-directional communication of events and information between smart devices and other devices, applications and networks.

This document provides a reference architecture for Smart Device Communications.

1 Introduction

This document is a member of a multi-part standard that, when taken in total, defines the requirements for communications pertaining to the access agnostic (e.g., PHY and MAC) monitoring and bi-directional communication of events and information between logical entities, such as Point-of-Attachment and applications or networks.

This document provides an M2M smart device communication reference architecture, describing functional elements and their interconnection. The reference architecture assumes some level of IP addressability as described herein. The Annexes provide identified use cases and demonstrate the applicability of the reference architecture to the support of those use cases.

The terms marked with *italicized fonts* are intended to show a logical entity.

2 References

2.1 Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. ANSI and TIA maintain registers of currently valid national standards published by them.

References are either specific (identified by date of publication, release level, etc.) or non-specific. For a specific reference, subsequent revisions do not apply. For a non-specific reference, the latest version applies: a non-specific reference implicitly refers to the latest version.

[1] TIA-4940.000: Smart Device Communications;
List of Parts.

[2] Hypertext Transfer Protocol -- HTTP/1.1
http://tools.ietf.org/html/rfc2616

[3] HTTP Over TLS
http://tools.ietf.org/html/rfc2818

2.2 Informative References

The following documents may be useful to the reader

[a] HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV)
http://tools.ietf.org/html/rfc4918

[b] Calendaring Extensions to WebDAV (CalDAV)
http://tools.ietf.org/html/rfc4791

[c] Architectural Styles and the Design of Network-based Software Architectures
http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

[d] Remote Monitoring Detailed Use Case, March 21, 2008
U.S. Department of Health and Human Services, Office of the National Coordinator for Health Information Technology
http://healthit.hhs.gov/portal/server.pt/community/use_cases_and_requirements_documents/1202/remote_monitoring/15669

[e] Continua Health Alliance
http://www.continuaalliance.com

3 Definitions, Symbols and Abbreviations

This section contains definitions, symbols and abbreviations that are used in this document.

3.1 Definitions

AAA-SD: provide authentication, authorization and accounting services to other entities in the network to establish and enforce security policies. The services may include generation of keys, generation and validation of certificates, validation of signatures, etc.

Home Application: The *home application* is a logical entity that is responsible for the business logic, either directly or via supervision and interaction with *node applications* and *PoA applications* and with *PoA devices*.

Node Application: The *node application* is a logical entity that acts as an intermediary between the *home application* and the *PoA application* and between the *home application* and the *PoA device*. The *node application* interacts with *home application*, other node applications, *PoA application* or *PoA device*, and may perform functions such as a data aggregation, storage, load balancing, etc.

PoA Application: *PoA application* is a logical entity that provides resources to *node* or *home applications* or to other *PoA applications*. The *PoA application* interacts with *home*, *node*, other *PoA applications* or with *PoA devices*. The *PoA application* may perform functions such as autonomous reporting of values reported by devices, monitoring for values reported by devices that exceed specified limits, trend analysis of values reported by devices, etc.

Container: The container is a logical entity that provides services to the applications that operate within it, and enforce security policies.

3.2 Abbreviations

AAA-SD	AAA of Smart Device
ACL	Access Control List
API	Application Programming Interface
DAP	Data Aggregation Point
EHR	Electronic Health Record
PHR	Personal Health Record
PoA	Point of Attachment
SDC	Smart Device Communications

4 Protocol Stack

This document pertains to the access agnostic monitoring and bi-directional communication of events and information between smart devices and other devices, applications or networks. Layers in the protocol stack at and below the transport layer are assumed to exist (including but not limited to TCP/IP, UDP/IP, HTTP, HTTPS, DHCP, Diff-Serv, MPLS, XMPP) and their descriptions are beyond the scope of this document.

To maintain a consistent interface to the transport layers (over fixed-point wireless, over wireless local area network, over digital subscriber line, etc…) a convergence layer is introduced into the protocol stack, as illustrated in Figure 4-1. (The dotted lines in the node indicate optional capability.)

PoA, node and server are considered above the access networks.

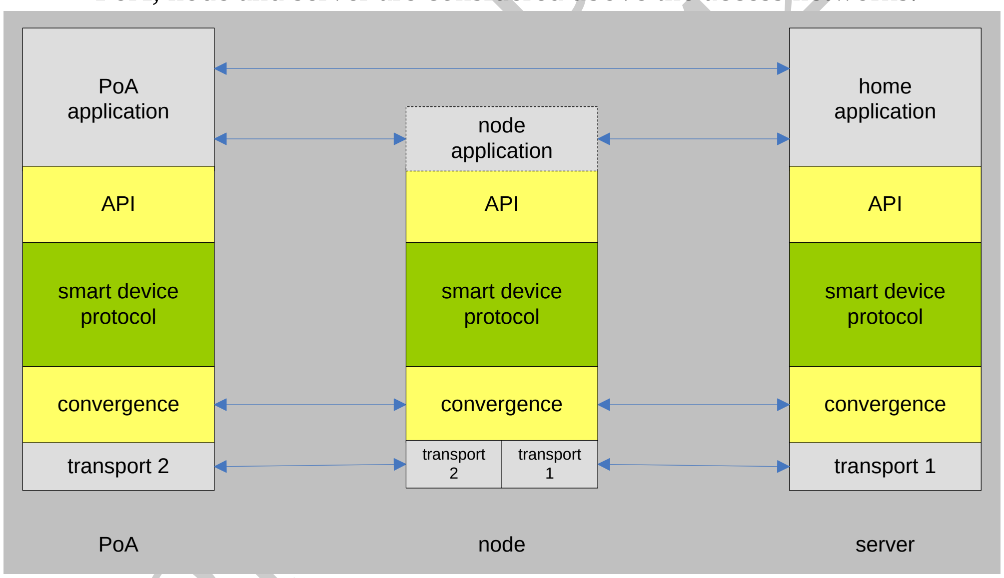

Figure 4-1 SDC Protocol stack

5 High Level System Architecture

Figure 5-1 depicts the high level Smart Device Communication system architecture.

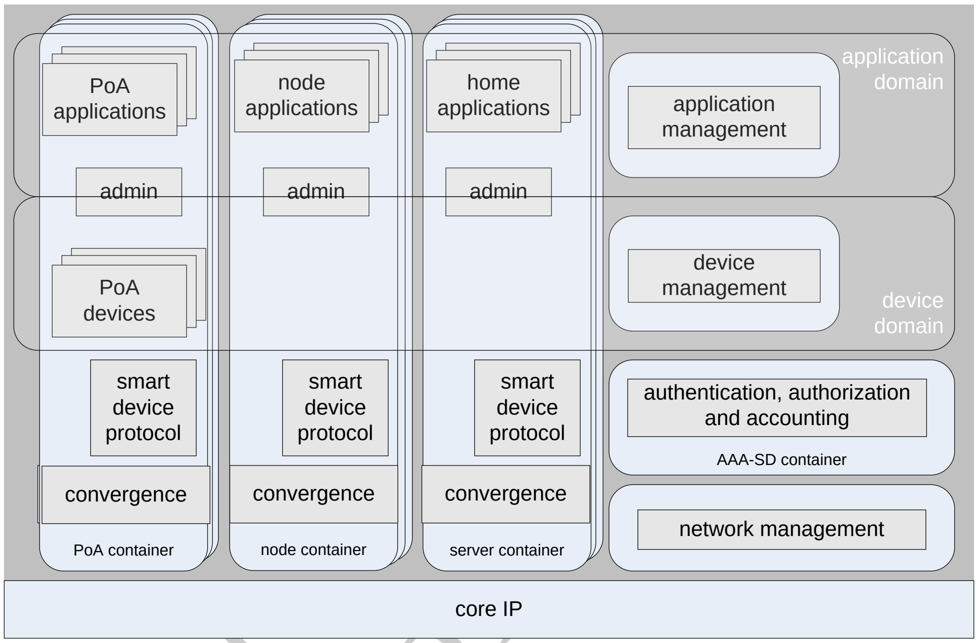

Figure 5-1 High Level System Architecture

The high level system architecture shown above may be described as a distributed cooperative computing system. The container provides services to the application(s) that operate within it, and enforce security policies.

In Figure 5-1, some containers are labeled, PoA container, node container and server container. The labels are for ease of reference and imply some level of logical grouping.

Some containers are not labeled implying that the entities within them can operate in any convenient appropriate container.

5.1 Entities

The server container hosts the *home application*. The *home application* is responsible for the business logic, either directly or via supervision and interaction with the applications hosted in containers labeled node and PoA. The container labeled server possesses an IP address. The *home application* may interact with *PoA devices* - resources that represent physical devices. Example implementations of the container labeled server include: a server farm with the JBoss container; a Google application.

The container labeled node hosts *node applications* that provide resources to the host applications. For example, the *home application* may delegate persistent storage, data aggregation, data pre-processing to *node applications*.

There may be zero or more containers labeled node in an application domain. There may be many *node applications* in each container labeled node. The container labeled node may be geographically distant from the container labeled server. The container labeled node possesses an IP address. *Node applications* may interact with *home application*, with other *node applications*, and with *PoA applications*. *Node applications* may interact with *PoA devices* – resources that represent physical devices. *Node applications* should be capable of detecting and responding to changes in their environments, such as a re-boot, or a network configuration change. An example implementation of the container labeled node includes a server with the Ruby-on-Rails application framework.

The container labeled PoA hosts *PoA applications* that provide resources to *node applications* or the *home application*. For example a *node application* may delegate averaging of readings from a device to a *PoA application*; a *home application* may delegate alarm notification to a *PoA application*. *PoA applications* shall be capable of detecting, reporting and responding as appropriate to changes in their environments, such as a re-boot, or a network configuration change. The container labeled PoA possesses an IP address. *PoA applications* may interact with *home application*, with *node applications*, and with other *PoA applications*. *PoA applications* may interact with *PoA devices* – resources that represent physical devices.

The container labeled PoA also hosts *PoA devices* – resources that represent physical devices. The devices may be interfaced to the container labeled PoA by a number of means, for example, Zigbee, Bluetooth, WiFi, or USB, etc.[1] The means by which physical devices are interfaced are outside the scope of this document. The combination of services with container labeled PoA and the software that implements a *PoA device* are responsible for whatever conversion is necessary to represent the physical device as a standardized resource. *PoA devices* shall be capable of detecting and responding to a re-boot and take appropriate action to set the physical device to a known safe state. *PoA devices* may become part of the application domain subject to security constraints. *PoA devices* may interact with *home application*, with *node applications*, with *PoA applications*, and with other *PoA devices*.

Example implementations of the PoA container include:

- for use cases involving private residence, a router/DSL modem combination; a set-top box; a smart phone;
- for use cases involving commercial buildings, a router with fiber backhaul; a router with satellite backhaul;
- for use cases involving vehicular telematics, a smart phone, a telematics control unit;

[1] Use of these trademarks does not constitute an endorsement by TIA or this Subcommittee. Wherever applicable other technologies may be substituted or included.

- for use cases involving patients, a smart phone, a home health monitoring unit.

The application domain spans software applications that operate in a number of containers. An application domain:

- shall contain at least one *home application*;
- may be associated with one or more *home applications* operating in the container labeled server;
- may contain zero or more *node applications* operating in zero or more containers labeled node;
- may contain one or more *PoA applications* operating in one or more containers labeled PoA;
- may contain one or more *PoA devices* operating in one or more containers labeled PoA;
- may contain an application management entity that cooperates with the admin entity in each of the containers to manage applications in the application domain. The application management entity may be integrated with the *home application*.

The device domain contains the devices in the system. A device domain:

- may contain a device management entity that cooperates with the admin entity in each of the containers labeled PoA to manage *PoA devices*. The device management entity may be integrated with the *home application*.

The application management entity is responsible for the ordering, configuration, customization, delivery, installation and maintenance of applications.

The device management entity is responsible for the ordering, configuration, customization, delivery, installation and maintenance of devices.

Authentication and authorization services provide services to other entities in the network to establish and enforce security policies. The services may include generation of keys, generation and validation of certificates, validation of signatures, etc.

The network management entity is responsible for the management of the access, transport and core network, including provisioning, supervision, repair and maintenance. Specification of this entity is outside the scope of this document.

The convergence entity is responsible for maintaining a standardized interface for applications operating within the container, regardless of the access network and transport network characteristics.

The smart device protocol entity provides Container-specific services to the *home application*, the *node application*, the *PoA application* and the *PoA device*. The specification for such services will be developed in other parts of this multi-part standard, and the informative Annexes provide some guidance to their development. The services supported may include but not limited to:

- REST primitives: create; read; update; and delete;
- assembly and disassembly of messages in accordance with the SDC Protocol message format (or access to their constituent parts);
- message validation;
- authentication services, likely involving interaction with AAA-SD.

5.2 Resources

HTTP [2] is most widely used by Web browsers to provide human-readable display. A growing number of applications use HTTP as a substrate protocol, for example: WebDAV [a] for network file system; and CalDAV [b] for calendaring. The SDC protocol uses HTTP as a substrate protocol and its companion HTTP over TLS [3] in cases where additional security is required.

Objects within the system are addressed via a Uniform Resource Locator, URL, which is constructed in a logical manner. By way of illustration, consider the following:

- the container labeled server has a DNS entry that corresponds to example.com. Consequently, to address the container, an application may use the URL
  http://example.com/
- The container may host many applications, so to distinguish the *home applications* an application may use the URL
  http://example.com/home/
  Hence, *node applications* and *PoA applications* may address their *home application* if they are provided with knowledge of their *home application* URL at installation, for example.

The objects within the system maintain an interface that complies with the principles of Representational State Transfer (aka RESTful interface.) [c] Consequently, they support the ability to create a resource, to read a resource, to update a resource, and to destroy a resource, mapped on to the HTTP verbs post, get, put and delete respectively. By way of illustration (and ignoring security policies that may prevent the actions) consider the following:

- the container labeled PoA possesses an IP address, say 10.10.10.10. (It may also possess a DNS entry but for the purposes of this illustration, an IP address suffices.)
- by convention, PoA container maintains a resource named applications, which responds to a RESTful read with a list of all the

applications contained within it. Consequently, an application may be created using a RESTful create (an HTTP post) to provide the signed executable using the URL

http://10.10.10.10/applications

- By convention, specific applications are distinguished by an identifier that is unique within the context of the application. Consequently, an application may be destroyed using a RESTful destroy (an HTTP delete) using the URL

  http://10.10.10.10/applications/identifier

6 Reference Architecture

6.1 Reference Architecture Diagram

Figure 6-1 depicts the SDC reference architecture diagram, showing functional elements, and the interconnection reference points. Light blue boxes represent containers while light yellow boxes represent applications and/or devices.

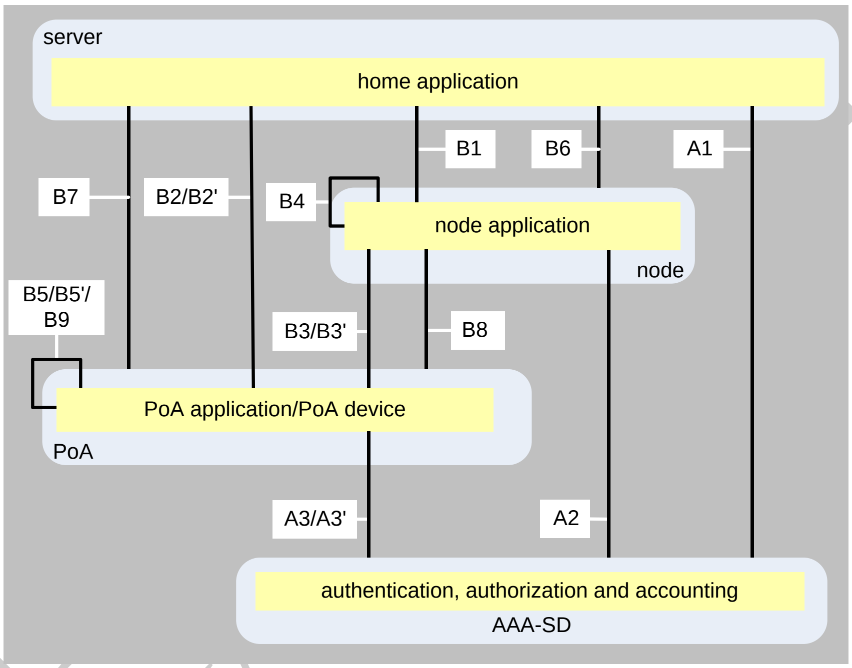

Figure 6-1 Reference architecture

6.2 Functional Elements

6.2.1 AAA-SD

A container that hosts an entity or entities that provide authentication, authorization and accounting services and interact with the admin entities in other containers (see 5.1) in establishing and enforcing security policies.

6.2.2 home application

The *home application* (see 5.1) is responsible for the business logic, either directly or via supervision and interaction with node and *PoA applications* and with *PoA devices*.

6.2.3 node application

The *node application* (see 5.1) acts as an intermediary between the *home application* and the *PoA application* and between the *home application* and the *PoA device*. It interacts with *home application*, *PoA application*, and *PoA* *device*, and may perform functions such as data aggregation, and load balancing.

6.2.4 PoA application

PoA application (see 5.1) provides resources to *node application*, *home applications* or to other *PoA applications*. *PoA applications* interact with *home application*, *node applications*, *PoA applications* and with *PoA devices*, and may perform functions such as autonomous reporting of values reported by devices, monitoring for values reported by devices that exceed specified limits, trend analysis of values reported by devices, etc.

PoA applications may be instances of a standardized class to facilitate their invocation by, for example, specifying parameters for their operation. For the purposes of illustration, consider the following:

stream: an instance to this class autonomously reads data from a specified source and streams it to a specified target according to some specified criteria.

average: an instance to this class autonomously reads data from a specified source, computes the average according to some specified criteria, and reports the result to a specified target according to some specified criteria.

limit: an instance to this class autonomously reads data from a specified source, compares the data with some specified limits, and reports to a specified target if the limits are exceeded.

trend: an instance to this class autonomously reads data from a specified source, computes the trend according to some specified criteria, and reports to a specified target if the computed trend exceeded some specified criteria.

6.2.5 PoA device

A *PoA device* (see 5.1) is a resource that represents a physical device. The means by which physical devices are interfaced are outside the scope of this document. The combination of services provided by the PoA container and the software that implements a *PoA device* are responsible for whatever conversion is necessary to represent the physical device as a standardized resource.

6.3 Reference Points

A1: provides for interaction between the AAA-SD container and the *home application*.

A2: provides for interaction between the AAA-SD container and the *node application*.

A3: provides for interaction between the AAA-SD container and the *PoA application*.

A3': provides for interaction between the AAA-SD container and the *PoA device*.

The realization of **A3** and **A3**' may be identical.

The realization of **A1, A2, A3** and **A3'** may be identical.

B1: provides for interaction between the *home application* and a *node application*, including bi-directional communication of control information, events and data.

B2: provides for interaction between a *PoA application* and the *home application*, including bi-directional communication of control information, events and data.

B2': provides for interaction between a *PoA device* and the *home application*, including bi-directional communication of control information, events and data.

The realization of **B2** and **B2**' may be identical.

B3: provides for interaction between a *PoA application* and a *node application*, including bi-directional communication of control information, events and data.

B3': provides for interaction between a *PoA device* and a *node application*, including bi-directional communication of control information, events and data.

The realization of **B3** and **B3**' may be identical.

B4: provides for interaction between the different *node applications*, possibly in different containers, including bi-directional communication of control information, events and data.

The realization of **B1** and **B4** may be identical.

B5: provides for interaction between the different *PoA applications*, possibly in different containers, including bi-directional communication of control information, events and data.

B5': provides for interaction between the different *PoA devices*, possibly in different containers, including bi-directional communication of control information, events and data.

The realization of **B5** and **B5**' may be identical.

The realization of **B2, B2', B3,** and **B3'** may be identical.

B6: provides for interaction between the *home applications* and a *node container*, including bi-directional communication of control information, events and data.

B7: provides for interaction between the *home application* and a *PoA container*, including bi-directional communication of control information, events and data.

B8: provides for interaction between *node applications* and a *PoA container*, including bi-directional communication of control information, events and data.

B9: provides for interaction between a *PoA application* and a *PoA device*, including bi-directional communication of control information, events and data.

Annex A. Application of the reference model (informative)

A.1. Introduction

This Annex is informative.

In a series of informative Annexes, Annexes A thru C, the applicability of the reference model is demonstrated using pseudo sequence diagrams that provide illustrative examples of how the reference architecture addresses the needs of specific use cases. Further examples, including examples of the use of the *node application*, may be provided in subsequent revisions to this document.

Light blue boxes represent containers while light yellow boxes represent applications and/or devices. In this Annex, the term “interface” is synonymous with “reference point” used in previous sections.

This Annex provides information regarding common aspects of the use of the reference architecture.

A.2. Common Aspects

This section provides an illustration of the message exchanges that are considered common to all vertical applications. In all cases, the actions of the application in the PoA are assumed to occur within the security policy; for brevity, interactions with the AAA-SD are not included.

These pseudo sequence diagrams are not intended to replace a Stage-2 Description, or to define message syntax or semantics. The names used in the diagrams have no significance other than to serve as a label to aid discussion. An indivisible message sequence is assumed to be a RESTful request/response. The pseudo sequence diagrams are not intended to specify exact message sequence, since a sequence of RESTful requests may be issued without waiting for a response from a previous request. Responses may arrive in sequence different from the sequence of requests.

We assume that the necessary preconditions to support RESTful request/response have already been established.

A.2.1. Application Registration

The *PoA container* may host a number of applications, for example, an application for the smart grid that concerns itself specifically with the electrical grid. There may also be applications that concern themselves with water supply, gas supply, intrusion detection, broadband service, etc.

We assume that the application is configured at deployment with knowledge of its *home application*, together with the necessary credentials to identify itself. Its *home application* may be a *home application* or a *node application*, or a third party, such as a SIP service.

The *PoA application* should respond to events such as power-on, re-boot or connected-to-network by registering with its home application. Figure A-1 illustrates a message exchange to provide registration of the *PoA application* with the *home application*. A similar exchange could be used for the *node application* to register with the *home application* as required. The *home application* may acknowledge receipt of the registration.

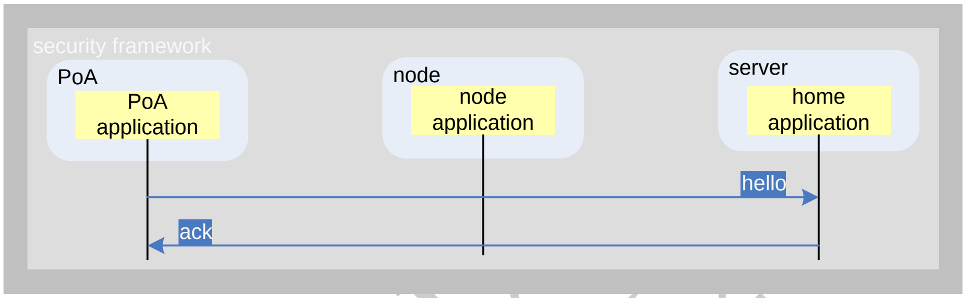

Figure A-1 Application Registration

A.2.2. Devices

As shown in Figure 5-1, the *PoA device* belongs to the *PoA container*. Consequently, the *PoA container* may expose a RESTful resource ***devices***.

As noted in §5.1, *PoA device* may become part of the application domain and may then be considered to belong to a *PoA application*. That *PoA application* may also expose a RESTful resource ***devices***.

A.2.2.1. Device Discovery

The *PoA container* should maintain a RESTful resource named devices that responds to a RESTful read with a list of devices with a standardized response that provides sufficient detail for the application.

The *PoA application* should maintain a RESTful resource named devices that responds to a RESTful read with a list of devices to which it has access with a standardized response that provides sufficient detail for the application.

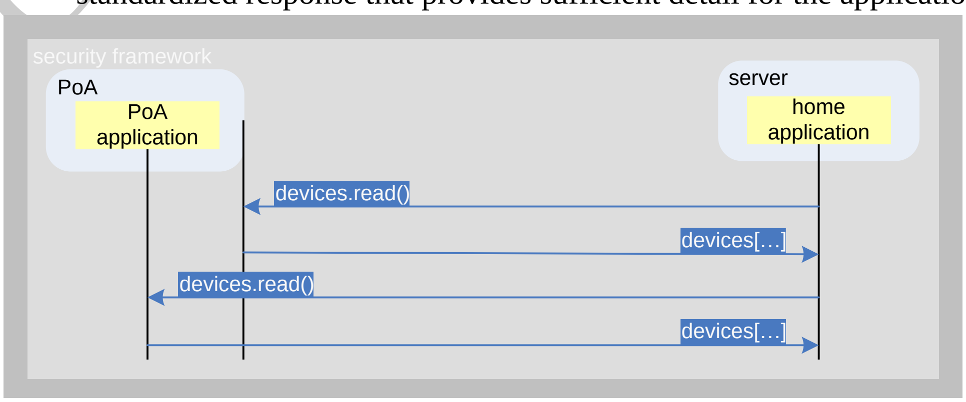

Figure A-2 List Devices

A.2.2.2. Add Device

When new hardware is added to the PoA, a *PoA device* may be added to the *PoA container* using the RESTful create primitive on the ***devices*** resource, which may, for example, add driver software to the platform.

The *PoA device* may be made known to the *PoA application* using the RESTful create primitive on the ***devices*** resource, which should respond to a request to add a device with a standardized response that provides sufficient detail for the application.

Potential interactions between the *PoA application* and the *PoA device* is for future study.

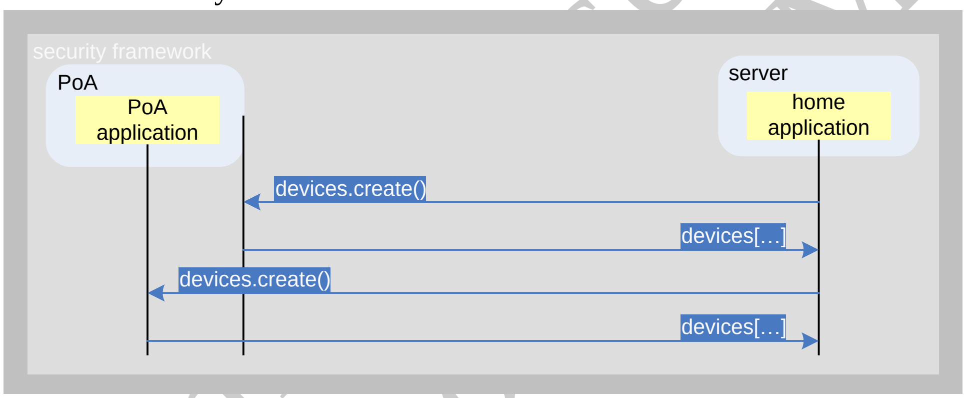

Figure A-3 Add Device

A.2.2.3. Delete Device

When hardware is removed from the PoA, the associated *PoA device* may be deleted from the *PoA container* using the RESTful delete primitive on the ***devices*** resource, which may, for example, remove driver software from the platform.

The *PoA application* may be requested to release resources associated with a *PoA device* using the RESTful delete primitive on the ***devices*** resource, which should respond to a request to delete a device with a standardized response that provides sufficient detail for the application.

Potential interactions between the *PoA application* and the *PoA device* is for future study.

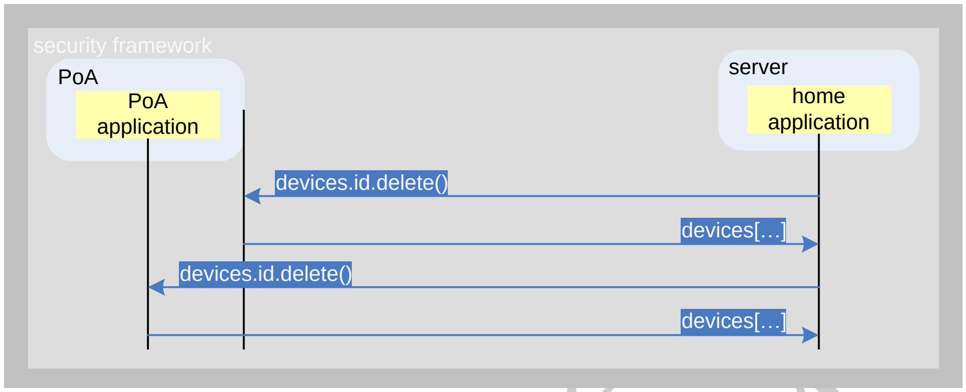

Figure A-4 Delete Device

A.2.2.4. Direct Device Interaction

The *PoA device* resource may be exposed by both the *PoA container* and the *PoA application*.

As a RESTful resource, the *PoA device* should respond to a standardized set of requests applied to a specific device. For the purposes of illustration, we consider the following two requests:

read: responds with the current reading of the device in a standardized format specific to the device and with sufficient detail for the application.

update: sets one or more parameters for the device in a standardized format specific to the device and with sufficient detail for the application

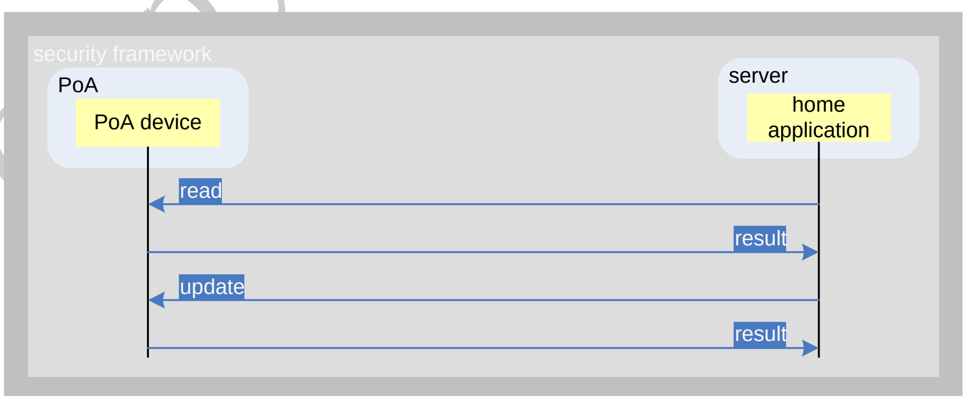

Figure A-5 Direct Device Interaction

A.2.3. PoA application

The techniques used for PoA device discovery, add device, delete device and direct device interaction may be extended to accommodate PoA applications.

For the purposes of illustration, we assume that the PoA container exposes a RESTful resource named ***applications***, and security policies that would typically prevent disclosure of *PoA applications* outside a particular application domain are not considered.

A.2.3.1. Application Discovery

Application discovery is accomplished using the RESTful read primitive on the RESTful resource named ***applications*** of the *PoA container*.

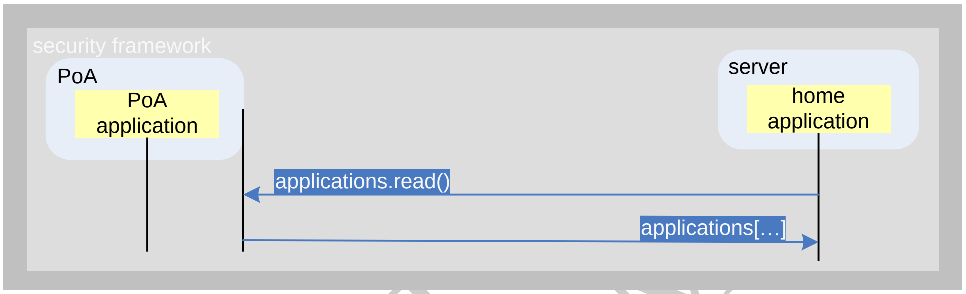

Figure A-6 List PoA Applications

A.2.3.2. Add Application

Adding an application is accomplished using the RESTful create primitive on the RESTful resource ***applications*** of the *PoA container*.

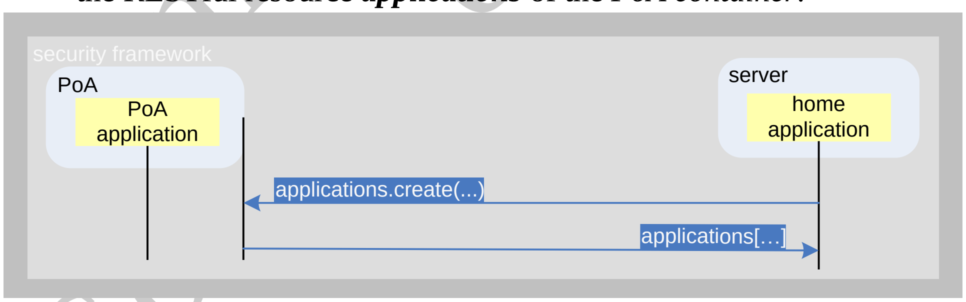

Figure A-7 Add PoA Application

Depending on the parameters in the *applications.create(…)* request (for example, signed executable code) the PoA container will create an object to represent a new *PoA application*.

A.2.3.3. Delete Application

Deleting an application is accomplished using the RESTful delete primitive on the RESTful resource ***applications*** of the *PoA container*. In this example, the PoA application responds with a list of application after the delete.

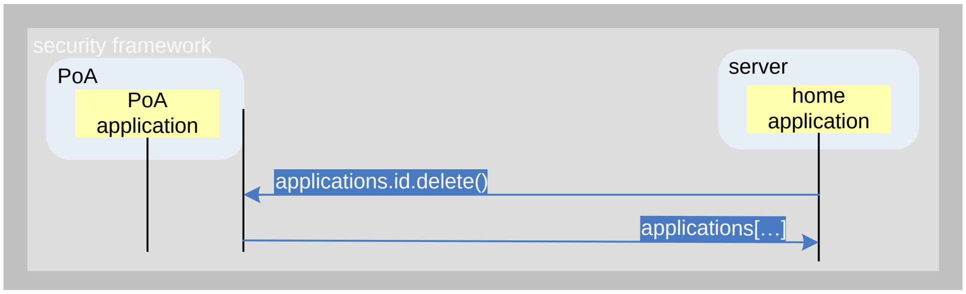

Figure A-8 Delete PoA Application

A.3. Verification of Reference Model with Connection Scenarios

The Reference Architecture is shown in Figure 6-1 with various interfaces linking different entities. This section provides a functional check on the entities and the interfaces based on various common connection scenarios so as to validate the need for each interface. Not all interfaces are used at the same time for all connection scenarios. Moreover, the interfaces linking entities are considered as logical links rather than the physical connections although there are occasions where physical and logical links are implemented on the same physical interface.

The scenarios under examination are the most popular ones include:

- A *PoA application*/*PoA device* connects to the home application directly;
- A *PoA application*/*PoA device* connects to the *home application* through an intermediate node such as a Local Gateway, a Data Aggregation Point (DAP), or, a Relay node (each with a node application);
- A *PoA application* connects to other *PoA application* directly.

A.3.1. A PoA application/PoA device Connects to the home application Directly

In this scenario, a *PoA application/PoA device* is connected to the *home application*, and hence, interface B2/B2' will be the bearer to convey all the control and configuration information as well as the data. We may assume that the *PoA application/PoA device* is configured with the URL of its *home application*, either at manufacture or at a subsequent configuration operation.

In order to ensure the integrity and security of the communications, the identity of the *PoA application/PoA device* needs to be authenticated and authorized to have the access to the *home application*. The architecture supports a variety of security models to assist the home application in the authentication and authorization. Interface A1, supports the exchange of information between the *home application* and the AAA-SD[2]. Interfaces A3/A3' support the exchange of information between the *PoA application* and the AAA-SD.

[2] Notice that the AAA-SD function here is for Smart Device Communications, which may be different from the "AAA function in wireless or broadband communications".

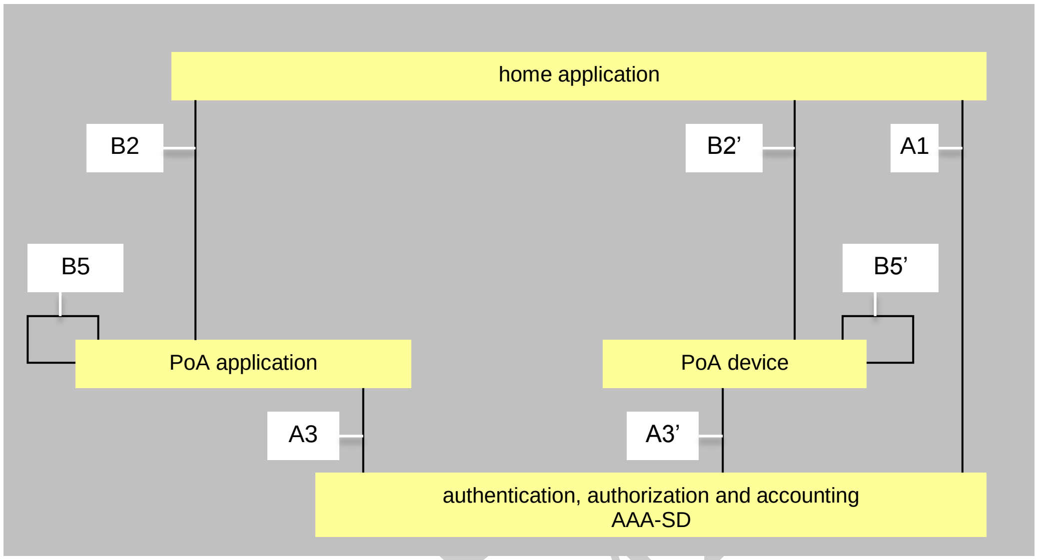

Figure A-9 PoA application/PoA device connects to home application

One example for this scenario is an application for the smart grid that concerns itself specifically with the electrical grid. There may also be applications that concern themselves with water supply, gas supply, intrusion detection, broadband service, etc.

For this example, we assume that the application is configured at deployment with knowledge of its *home application*, together with the necessary credentials to identify itself. The *home application* may be an application in the server container or a proxy, such as a *node application*.

A.3.2. A PoA application/PoA device connects to the home application via a node application

In this scenario, a *PoA application* is connected to a *node application*. The *node application* may act as, for example, a simple multiplexor, a data aggregator, or persistent storage. The *home application* may interact with the *node application* via interface B1.

In such a connection scenario, we may assume that the *PoA application* is configured with the URL of its *node application* during a configuration operation. Interfaces B3/B3’ supports the exchange of information between the *PoA application* and the *node application*. (The *PoA application* may also be configured with the URL of its *home application*, and may simultaneously maintain sessions with the *node application* over B3/B3’ and with the *home application* over B2/B2’.) Interface B4 supports exchange of information between *node applications* running in different containers.

The *node application* may or may not be configured with the URL of its *home application*. It may, for example, be capable of providing service to many *home applications* by checking the credentials of the *home applications* as they attempt to access services provided by the *node application*.

In order to ensure the integrity and security of the communications, the identity of the *PoA application* needs to be authenticated and authorized to have the access to the *node application*. The architecture supports a variety of security models to assist the node application in the authentication and authorization. Interface A2 supports exchange of information between the *node application* and the AAA-SD.

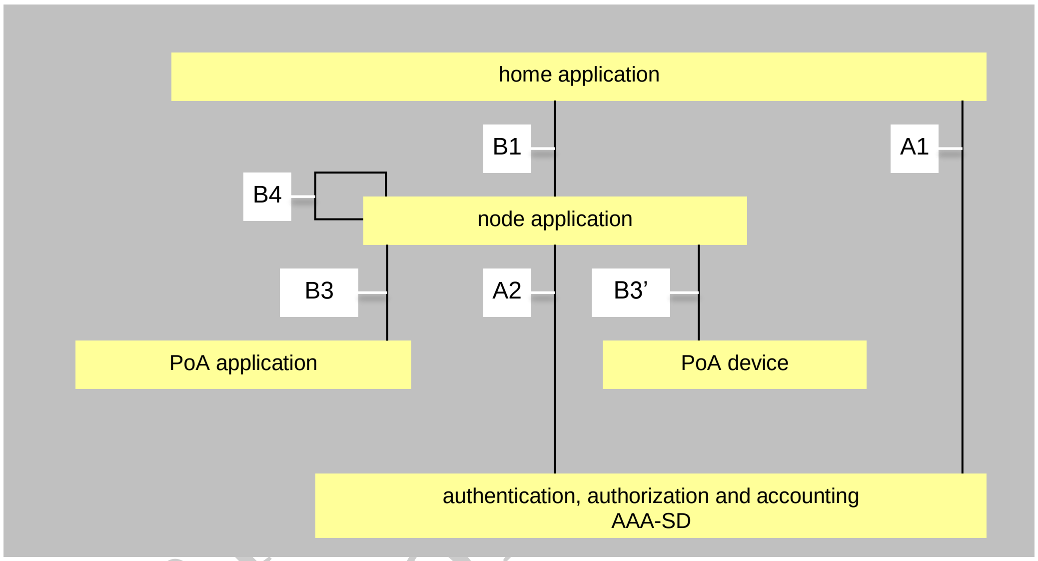

Figure A-10 PoA application/PoA device connects to the home application via a node application

A.3.3. PoA applications interconnected with PoA devices

In this scenario, a *PoA application* is connected to one or more *PoA applications*.or one or more *PoA devices*. In the case where the *PoA application* and *PoA device* run in the same container, the SDC Container may provide efficient mechanisms for inter application communications. In addition, it is likely that the container supports connection to its loopback address, *localhost*. Provided that the security policies permit it, *PoA applications* may discover and communicate with other applications using the same mechanisms that are available to, for example, the *home application*.

In the case where the *PoA application* and *PoA device* run in different containers, interfaces B5/B5’/B9 support the exchange of information between *PoA application* and *PoA application*, *PoA device* and *PoA device*, and *PoA application* and *PoA device* respectively.

In order to ensure the integrity and security of the communications, the identity of the *PoA application* needs to be authenticated and authorized to have the access to the peer *PoA application*. The architecture supports a variety of security models. Interfaces A3/A3' supports exchange of information between the *node application* and the AAA-SD.

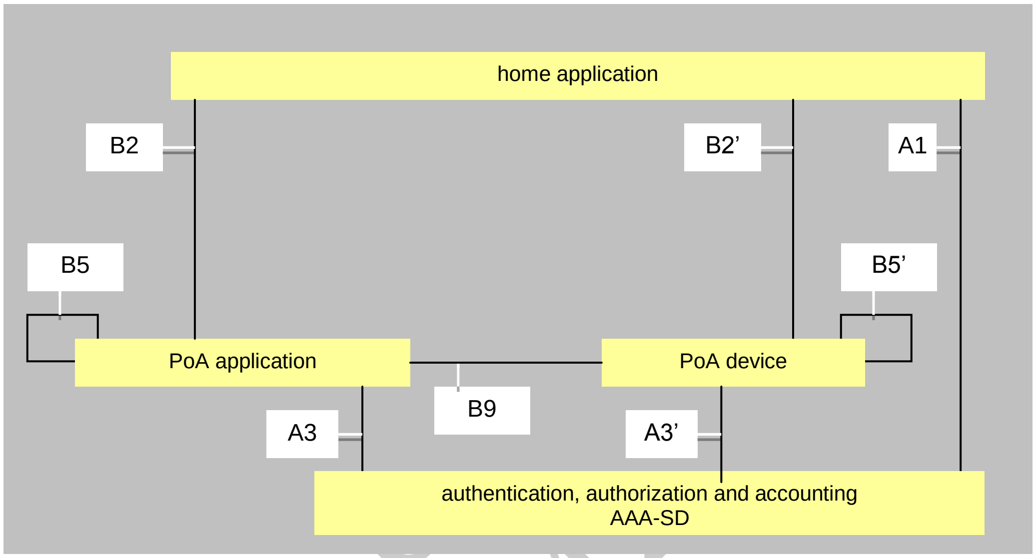

Figure A-11 PoA applications/PoA devices connect with each other

A.3.4. Conclusion

With these scenarios, the needs for all interfaces, A1 through A3 and B1 through B5, and B9, in Figure 6-1 have been confirmed.

Annex B. SDC Application Example: In-Building Control (informative)

B.1. Introduction

This Annex is informative.

In a series of informative Annexes, the applicability of the reference model is demonstrated using sequence diagrams that provide illustrative examples of how the reference architecture addresses the needs of specific use cases.

This Annex provides information regarding use of the reference architecture in a building automation application.

B.2. Scenario of the Application

On the quest of building a smarter and greener world, power management system at the user level bears as much weight as the energy transmission loss management. A well-designed Building Automation System not only affords tenants enhanced energy savings via effectively run climate control systems, but additional values, such as Security and Life Safety services, can be added to improve the quality of life..

Conventionally, the overall automation system is a conglomeration of several subsystems such as “HVAC Control” and “Light Control”. Each of the subsystem is implemented with proprietary solutions. With SDC, our attempt is to identify the common control elements needed for each subsystem and tie all the common elements through a centralized modular way so that new elements or functions can be added easily in the future.

B.3. Building Automation System

A Building Automation System typically requires the following control functions:

- **Manages HVAC Operation:**
    - Through adjusting the temperature, humidity, AC/heating units, and vents, etc.
- **Manages Water Distribution:**
    - Through the on off control of chillers, heaters, valves, and pumps, etc.
- **Power Management System:**
    - Records power usage information in the building (usually at the building level today, moving to per floor and per space)
    - Participates in utility smart metering and dynamic pricing policies

- **Lighting Management System:**
    - Manages the lighting schedules for all public spaces (hallways, bathrooms, etc.).
    - Manages lighting in offices.
    - Usually constrained to overhead lighting, desk/floor lighting excluded.
- **Security System:**
    - Manages physical access to the exterior and interior of the building.
    - Most individual offices are controlled with keys; however, multi-use spaces are going electronic.
- **Life Safety System:**
    - Fire alarms, pumps, elevator lockouts, etc.
- **Network Management System:**
    - Manages the network infrastructure in the building, monitors external network access, including IDS and firewalls, as well as loads on internal equipment such as switchgear and wireless access points.

In the existing practice, the aforementioned control functions are provided by multiple vendors. The following is an actual case. The HVAC Control, Security, Lighting Control, Network Management subsystems are provided by different vendors. Two issues make the system integration more challenging:

- In most cases these subsystems do not communicate with each other.
- In the cases where the subsystems need to exchange information, it is almost exclusively performed at the "system" level, not the sensor/device level.

The work on SDC is to not only enable these systems to exchange information, but also for them to share a common set of sensors within a building.

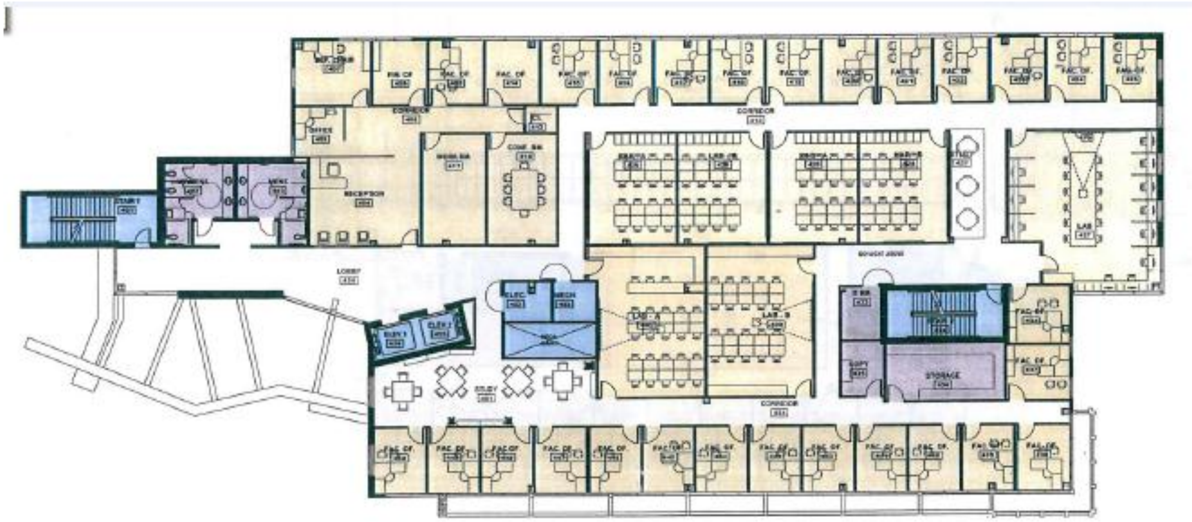

Figure B-1 Campus Map

When each of the subsystem is provided by a vendor with proprietary solutions, the data structure of all the elements and parameters are spread like the following:

HVAC Control by Vendor 1:

–Room	401
–Temperature	72
–Humidity	45%
–Louvre	Open

–Room	402
–Temperature	75
–Humidity	47%
–Louvre	Closed

Security provided by Vendor 2:

–Room	401
–Locked	True
–LastUnlockedBy	Peter
–Motion	False

–Room	402
–Locked	False
–LastUnlockedBy	Mitch
–Motion	True

Lighting Control provided by Vendor 3:

–Room	401
–Lights	Full
–Room	402
–Lights	Half

Network Management provided by Vendor 4:

–Room	401
–Port1.MacAdd	00:11:43:AB:32:FE
–Port1.Connected	True
–Room	402
–Port1.MacAdd	-
–Port1.Connected	False

Power Control provided by Vendor 5:

–Room	401
–Power:	1200W
–Room	402
–Power:	185W

The data structure can be easily organized by rooms as follows:

	Room 401	Room 402
Temperature	72	75
Humidity	45%	47%
Louvre	Open	Closed
Locked	True	False
LastUnlockedBy	Peter	Mitch
Motion	False	True
Lights	Full	Half
Port1.MacAdd	00:11:43:AB:32:FE	-
Port1.Connected	True	False
Power:	1200W	185W

Annex C. Remote Patient Monitoring (informative)

This Annex is informative.

In a series of informative Annexes, the applicability of the reference model is demonstrated using sequence diagrams that provide illustrative examples of how the reference architecture addresses the needs of specific use cases.

This Annex provides information regarding use of the reference architecture in an eHealth application [d]. Section 8 of the Use Case is examined in detail.

Much of the workflow in this example is in the applications, and we postulate an application structure simply to provide an illustration of how the application of the reference model could support the Use Case.

C.1. Register device, patient and data recipient

"Capability to maintain information describing the remote monitoring device, the patient being monitored, and the individuals who will be reviewing the monitoring data. For example, this may include registering the device with the manufacturer or data intermediary and performing other functions to uniquely identify the individual being monitored." See [d], §8.1.

C.1.1. Pre conditions

For the purposes of this example, we postulate an application structure as follows:

- a *home application* is associated with the clinician, and maintains a database of patients, provides human interaction with the clinician, such as a portal. The *home application* is configured (possibly dynamically) with knowledge of at least one *node application* together with the credentials to retrieve data from the *node application*.
- a *node application* is associated with the Device Data Intermediary and provides persistent storage of the data collected during the remote patient monitoring session, together with controlled access to that data. The *node application* maintains a standard object, say *reports*, such that *reports/id* represents the data associated with a particular remote monitoring session. This *node application* is responsible for authenticating and authorizing access to the data.
- another *node application* is associated with the Personal Health Record (PHR) and provides storage. The *node application* maintains a standard object, say *phrs*, such that *phrs/id* represents the PHR of a particular individual. This *node application* is responsible for authenticating and authorizing access to the data.
- a *PoA application* is associated with the care coordinator and the remote monitoring equipment. For example, the remote monitoring

equipment could be a smartphone communicating with a medical sensor via a Continua [e] compliant interface, together with the *PoA application* in the smartphone to maintain the standard object. (That object could represent a single sensor, or a collection of clinically relevant data from a number of sensors.) The *PoA application* is configured with the URL of its *home application* during a configuration operation.

The clinician orders remote monitoring per 7.1.3 of the Use Case, which triggers a workflow in the *home application* that generates a unique id for this remote patient monitoring session, and maintains an association of that unique session id with the patient. The unique session id is communicated to the care coordinator along with the initiate request, code 7.2.1 of the Use Case.

The care coordinator takes the necessary action to attach the sensors to the patient, and initiate monitoring. For example, the care coordinator activates the *PoA application* and provides the unique *session_id*.

C.1.2. SDC interactions

When the care coordinator initiates the remote monitoring session, the PoA application registers with the *home application*, and provides the unique session id, and the identity of the care coordinator.

Although the interaction between the care coordinator and the PoA is beyond the scope of the SDC support, possible scenarios for entering data include: keyed in by the care coordinator; scanned in from a 2-D code using the camera in the smart phone; transferred from the "work-order' previously retrieved and available in the smartphone.

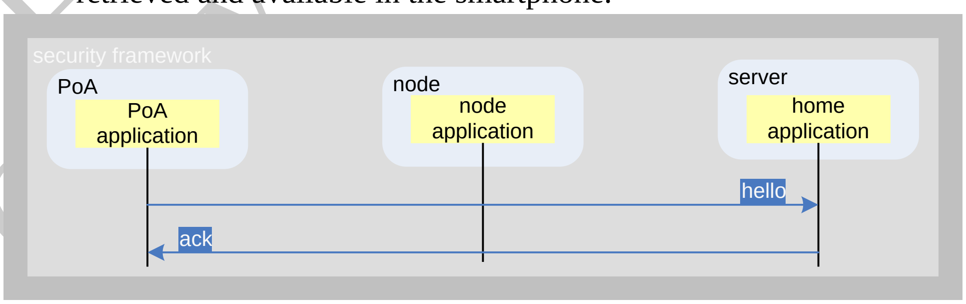

Figure C-1 Application Registration

The *home application* looks up the unique *session_id*, and instructs the *PoA application* to tag the data with the unique session id, and stream it to the intermediate node, providing the credentials to allow the *PoA application* access to the *node applications* as necessary.

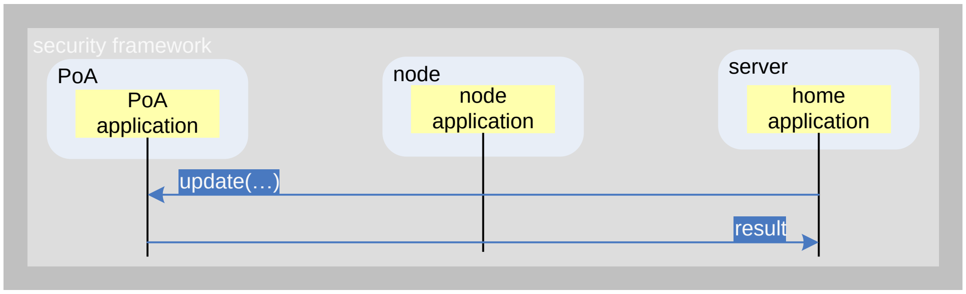

Figure C-2 Initiate Remote Monitoring

The *PoA application* streams data to the *node application*. To maintain confidentiality, the data may be streamed using REST primitives over an SSL connection.

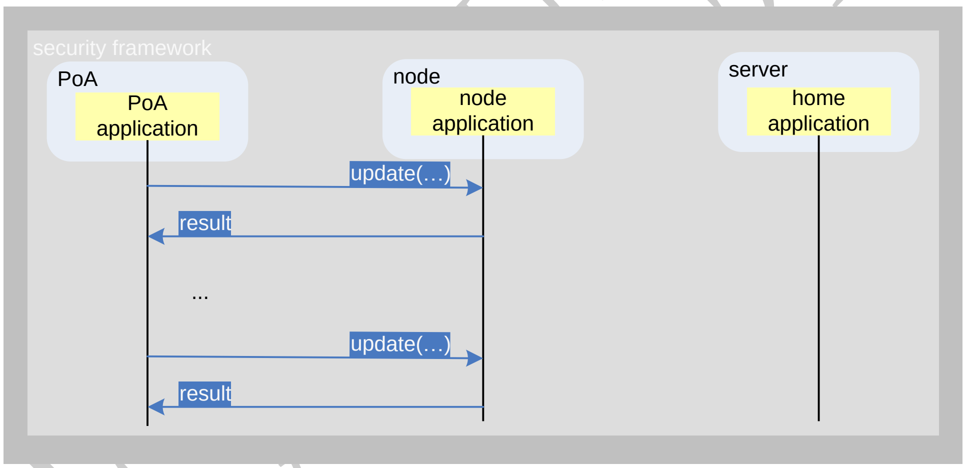

Figure C-3 Receipt of Data

The care coordinator completes the monitoring session via interaction with the *PoA application* (not specified in this example). The stream service terminates, and the *PoA application* sends notification of completion to the *home application*.

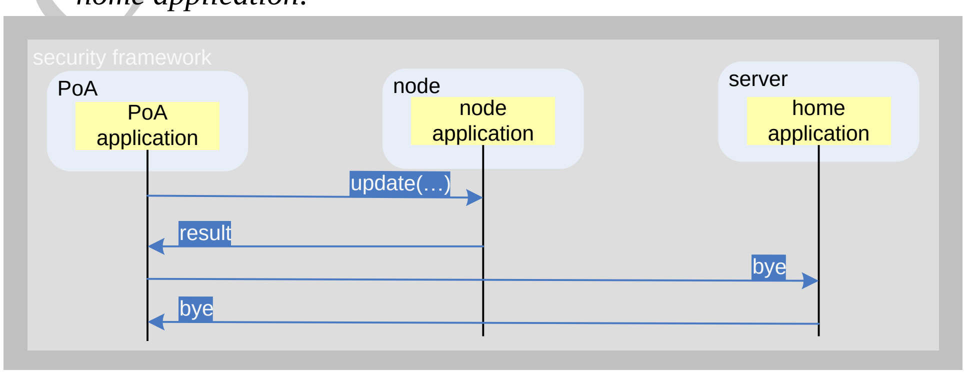

Figure C-4 Data Collection Complete

C.1.3. Post conditions

The *node application* is in possession of data from the remote patient monitoring session, with a unique tag that may be used to recover the data.

The *home application* is in possession of the care coordinators identity associated with the unique session id, which in turn is associated with the patient.

C.2. Data retrieval

"Capability to locate and retrieve requested data subject to consumer access decisions and local policies. The remote monitoring data is received via the information exchange and associated with the appropriate patient and data recipients. A clinician, care manager, or patient may access remote monitoring and clinical information directly via the information exchange using a portal if available." See [d], §8.2.

C.2.1. Pre conditions

Completion of remote patient monitoring session described in C.1.

C.2.2. SDC interactions

A clinician, care manager, or patient may access remote monitoring and clinical information as a user of a portal at the *home application*. That portal is responsible for authenticating the user, and managing access rights to the data. A user request for the data triggers a request from the *home application* to the *node application* against its standard object *reports* (see C.1.1). We assume that the *home application* has the ability to authenticate itself against all of it s *node applications*. The *node application* will authorize access to the data according to the credentials provided by the *home application*. To maintain confidentiality of the information, the exchanges may be performed using REST primitives over SSL.

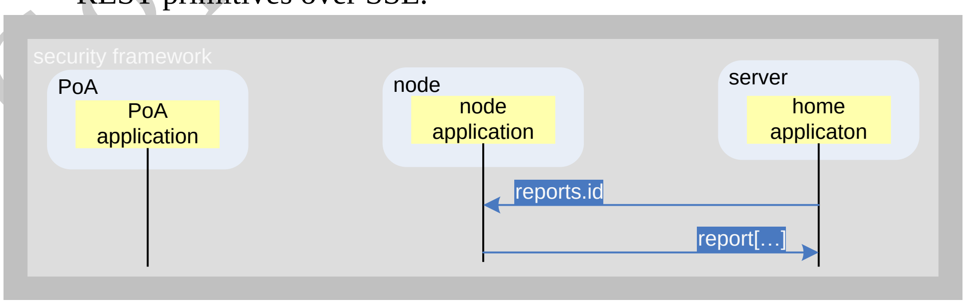

Figure C-5 Locate and Retrieve Data

The response from the *node application* may provide raw data, or a processed form of the data, depending on the application.

C.2.3. Post conditions

The user (clinician, care manager, or patient as appropriate) is in possession of the requested data, or is informed that they do not have access to that data.

C.3. Data delivery

"Capability to securely deliver data to the intended recipient and confirm delivery, including the ability to route data based on message content, if required. For example following the care coordinator's evaluation of the remote monitoring data via the information exchange, monitoring information may be delivered to the appropriate clinician's EHR or patient's personally controlled health record." See [d], §8.3.

C.3.1. Pre conditions

Completion of data retrieval described in C.2.

C.3.2. SDC interactions

If the Electronic Health Record (EHR) is maintained at the *home application*, no SDC interaction is necessary: the clinician updates the EHR via application dependent mechanism.

If the EHR or the PHR are maintained other than at the *home application*, they can be mapped onto a *node application*, which is likely a different *node application* from that supporting the Device Data Intermediary. To maintain confidentiality of the information, the exchanges may be performed using REST primitive over SSL.

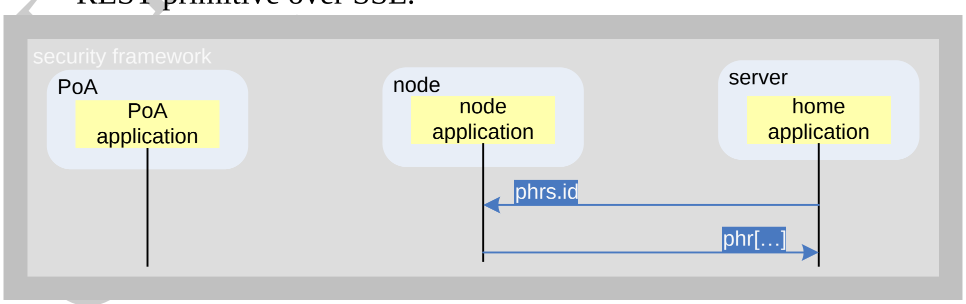

Figure C-6 Deliver Data

C.3.3. Post conditions

The clinician has updated the EHR for the patient, or the patient has updated the PHR dependent on the identity of the user.

C.4. Subject-data matching

"Capability to match available data to the appropriate person during retrieval or routing. For example, when the clinician requests additional clinical information for a specific person, the systems, processes, and policies facilitating information exchange are utilized to confirm that the data available for retrieval match the person of interest to the clinician." See [d], §8.4.

C.4.1. Pre conditions

Completion of remote patient monitoring session described in C.1.

C.4.2. SDC interactions

No SDC interactions are necessary to support this application specific requirement. However, for completeness of this example, all data is tagged, and data may be associated with the patient or the care coordinator via a database maintained at the *home application*.

C.4.3. Post conditions

Data is matched to the appropriate person.

THE TELECOMMUNICATIONS INDUSTRY ASSOCIATION

TIA represents the global information and communications technology (ICT) industry through standards development, advocacy, tradeshows, business opportunities, market intelligence and world-wide environmental regulatory analysis. With roots dating back to 1924, TIA enhances the business environment for broadband, mobile wireless, information technology, networks, cable, satellite and unified communications.

TIA members' products and services empower communications in every industry and market, including healthcare, education, security, public safety, transportation, government, the military, the environment and entertainment. TIA co-owns the SUPERCOMM® tradeshow and is accredited by the American National Standards Institute (ANSI).

ADVANCING GLOBAL COMMUNICATIONS

HEADQUARTERS
2500 Wilson Boulevard
Suite 300
Arlington, VA 22201-3834
+1.703.907.7700
+1.703.907.7727 (fax)
tiaonline.org