Smart Home Architecture and System Requirements

CONFIDENTIAL NON-BALLOTED WORKING DRAFT FOR REVIEW AND COMMENT BY ONEM2M SEPTEMBER 2015

Track	Smart Home
Editor:	Lindsay Frost (NEC), Sofia Ilieva (ProSyst)
Date:	12 August 2015
Document number:	HGI02029R52
Version:	1.0
Status:	Approved by SHTF for sending to MC
Keywords:	Smart Home Architecture, System Requirements, Device Abstraction Layer

Abstract

This document describes an open Home Gateway (HG) architecture bringing a wide variety of services to the home by creating a bridge between the cloud and the appliances in the in-home network.

 © Home Gateway Initiative – 2015 – All rights reserved

PAGE LEFT INTENTIONALLY BLANK

© Home Gateway Initiative – 2015 – All rights reserved

Contents

1 Important notice, IPR statement, disclaimer and copyright 4
2 Acronyms and terminology 5
2.1 Acronyms 5
2.2 Terminology 5
3 Scope and Purpose of the Document 7
3.1 Relationship with other HGI Specifications 7
4 Business requirements and Use Cases 8
5 Architecture 9
5.1 Overview 9
5.2 High-level view of smart home architecture 10
5.3 HG with specific Software Execution Environment (Open Platform 2.0) 13
5.4 Detailed HG Software Stack architecture 14
5.5 Access Control and Policies 17
5.6 Appliance States 17
5.7 Grouping of Appliances 19
6 Functional requirements 20
6.1 Requirements for RP1 Device Abstraction Layer 20
6.2 Requirements for RP1 Access Control 20
6.3 Requirements for DAL 21
6.4 Requirements for RP2 Appliance Driver Interface 21
6.5 Requirements for RP4 Remote Representation 22
6.6 Requirements for RP5 Local representation of appliances 22
6.7 Requirements for RP6 Native Driver Interface 22
6.8 Requirements for RP7 Reference point in the cloud 23
6.9 Requirements for RP8 WAN Network Connectivity Interface 23
7 References 24

© Home Gateway Initiative – 2015 – All rights reserved

1 Important notice, IPR statement, disclaimer and copyright

The Home Gateway Initiative (HGI) is a non-profit making organization which publishes guidelines, requirements documents, and test plans for broadband equipment and services which are deployed in the connected home. The scope of HGI's work covers Home Gateways, interconnection with the broadband access network, home networks requirements, software requirements, and other technology building blocks required to deliver broadband services.

This document is the output of the Working Groups of the HGI and its members as of the date of release. Readers of this document must be aware that it can be revised, edited or have its status changed according to the HGI working procedures.

The HGI makes no representation or warranty on the contents, completeness and accuracy of this publication.

This document, though formally approved by the HGI member companies, is not binding in any part on the HGI members.

IPRs essential or potentially essential to the present document may have been declared in conformance to the HGI IPR Policy and Statutes available at the HGI website www.homegateway.org.

Any parts of this document may be freely reproduced (for example in RFPs and ITTs) by HGI and non-HGI members subject only to the following:

- HGI Requirement numbers not being changed
- An acknowledgement to the HGI being given in the resulting document.

Trademarks and copyrights mentioned in this document are the property of their respective owners.

© Home Gateway Initiative – 2015 – All rights reserved

2 Acronyms and terminology

2.1 Acronyms

AAL	Ambient Assisted Living (home automation to assist aged/infirm persons)
API	Application Program Interface
CPE	Consumer Premises Equipment
CWMP	CPE WAN Management Protocol
DAL	Device Abstraction Layer
HAN	Home Area Network
HG	Home Gateway
HGI	Home Gateway Initiative
IP	Internet Protocol
IPR	Intellectual Property Rights
ISP	Internet Service Provider
LAN	Local Area Network
RP	Reference Point
OS	Operating System
OSGi	OSGi Alliance
SDT	Smart Device Template (a HGI term related to appliance function modelling)
WAN	Wide Area Network

2.2 Terminology

The following new terms have been defined by HGI for the purposes of this document:

- **Home Gateway** (HG) – Device connecting the HAN to the Internet, Operator Platform and Service Platforms.
- **Smart Home Functions** – Hardware/software implementing smart home functionality, which is typically part of the HG but could in some cases be implemented in a separate piece of hardware
- **Appliance** - A domestic electrical/mechanical machine which accomplishes some function used in smart home services.
- **Virtual appliance** – an appliance implemented purely in software
- **Data Point** – A property of an appliance, which can change while the appliance is in operation.
- **Action** – A function that can be executed by an appliance.
- **Reference point** - A point of interaction between two entities, which may be subject to standardization e.g. in the form of an API or protocol

The following role definitions are copied from [1] Open Platform 2.0:

- **Module** - A Service Module (or just module) is a downloadable, packaged collection of resources and/or code needed to provide a specific function.

 © Home Gateway Initiative – 2015 – All rights reserved

- **Service Application** - A Service Application (or just application) is a set of software modules and configurations that collectively implement a specific function or set of functions, possibly across several distributed platforms
- **Service Provider** - The Service Provider is a business entity. The Service Provider supplies the necessary means to provide the business related support of a specific Service Application.
- **Network Provider** - The Network Provider provides and manages wide area network connectivity between the Service Platform and other parties, which include the Gateway Operator and other Service Providers. In the case where the Service Platform is connected via the Internet, the Network Provider also supplies the Internet Service Provider (ISP) functionality.

© Home Gateway Initiative – 2015 – All rights reserved

3 Scope and Purpose of the Document

The rapid increase in the number of smart devices in the home, from entertainment systems to appliances and medical devices, and the now near ubiquitous broadband connectivity have created a significant new potential market for service providers. However, in order to exploit this market and provide an integrated customer experience there is a central component which is still missing – a set of standards for the home gateway which will enable interoperability between smart home entities.

Defining smart home related standards for the home gateway can benefit the market in several ways:

- Avoiding duplication of hardware and software, and the associated costs of having several disparate systems
- Providing opportunities for new ‘blended’ services based on combining data from different HANs and appliances
- Enabling automated, intelligent reaction to the environment – e.g. demand-side energy management.

This document defines a smart home system architecture and derives requirements for the Home Gateway.

3.1 Relationship with other HGI Specifications

This document contains requirements over and above those in the HGI Residential Profile [6], Open Platform 2.0 [5] and Wireless HAN Services [2]. All the requirements of those documents still apply, except that support of Open Platform 2.0 is only mandatory with regard to Section §5.3 "HG with specific Software Execution Environment (Open Platform 2.0)".

Note that the Energy Gateway mentioned in Figure 7 of the HGI Energy Management document [3] is one of the possible appliances in the home network, but for simplicity is not explicitly included in figures shown here.

© Home Gateway Initiative – 2015 – All rights reserved

4 Business requirements and Use Cases

The business requirements and use cases used as the basis for the architecture described in this document are described in "HGI-GWD035 Smart Home Use Cases" [4].

© Home Gateway Initiative – 2015 – All rights reserved

5 Architecture

The architecture is described first in terms of the entities and interfaces of the smart home eco-system, and then in terms of the software execution environment within the HG. There is a particular focus on the Device Abstraction Layer (DAL) which mediates between high-level applications and the specific local area network protocols and technologies in the home network.

The following reference points in the architecture are explained in more detail in the following sections, but brief definitions are given in the table below for convenience.

RP Number	Reference Point Description
RP1	A common representation of, and unified APIs to control and query, home appliances (in the software execution environment)
RP2	The reference point between the appliance drivers for various HAN technologies and the device abstraction layer.
RP3	(RP3 is a reference point which no longer features in this architecture, but the notation is retained for historical reasons)
RP4	A common representation of and control over home appliances in the local network for access by the Operator Platform viar a remote-access protocol
RP5	Support for local clients to communicate with the HG to access home appliances.
RP6	The reference point for the native drivers towards the smart home software
RP7	Reference point in the cloud between an Operator Platform and Cloud Service Applications
RP8	A description of the WAN connection(s) on the HG is available at this Reference Point.
RP9	The home area network (HAN) physical interface to appliances.

Table 1: Overview of Reference Points

5.1 Overview

In the HGI context, the Home Gateway is considered to be the heart of the smart home – it connects and controls the home appliances and serves as the main entry point for communication with the operators, service providers and utilities in the cloud. The end user has access to their services in the home either via the cloud or through a direct connection to the HG. This is illustrated in Figure 1 below.

© Home Gateway Initiative – 2015 – All rights reserved

Figure 1: HGI smart home architecture.

This document does not specify functionality to support direct connections between the Service Application Platform and the HG or home PCs/tablets (shown by dotted lines in Figure 1) for appliance access, but such connections are not precluded.

Note that the above Figure indicates that the Home Gateway itself may be implemented in more than one software and/or hardware entity, for example in a router and in a separate piece of equipment containing the smart home software functionality described within this document.

5.2 High-level view of smart home architecture

Figure 2 shows a schematic view of the high level entities and reference points of the HGI smart home architecture. This Figure depicts the smart home functions as a black box, deferring the discussion of the way in which remote requests from WAN or LAN side clients are translated to actions on smart home appliances. Only the external reference points are shown in this Figure.

© Home Gateway Initiative – 2015 – All rights reserved

Figure 2: Reference points between the HG software and software in the Cloud or local client software

Figure 2 refers to the following entities:

Entity Name	Description
Operator Platform	Service delivery platform for smart home services. Functionality includes gateway management, software management for the gateway, remote access from cloud service applications to the gateway, rule engines for home automation and other services.
Service Application (cloud)	Service application software may be distributed over several locations: parts may run in the cloud, others on the gateway. User interfaces may run on smartphones and/or tablets.
Local clients	Local client software, running on for example tablets or smartphones, interfacing directly with the HG over a local network to provide functionality for

© Home Gateway Initiative – 2015 – All rights reserved

	the user, mainly presentation of user interfaces.
HG Software Stack	This is the entire software running on an HG, including boot loaders, current and backup firmware images, secure storage, the operating system, an execution environment (e.g. Open Platform 2.0 [5], and any dynamically loaded software modules (e.g. 3rd Party services).
Smart Home Software	Software running on the operating system of the HG, which together with the HG Core Software provides the means for the user to access and/or automate the use of equipment within the home, and for remote access by the user and/or applications in the Cloud.
HG Core Software	Definition taken from [7] Section 4.2: The HG_Core uses an operating system (OS). On this operating system runs the HG native software that provides some or all of the home gateway functions as defined in the HGI Residential Profile [6]. Native drivers give direct access to hardware modules. The combination of the home gateway hardware, operating system, HG native software and drivers constitutes the HG_Core.

Table 2: Software and hardware elements in figure 2

Figure 2 also shows the following reference points for external interfaces:

RP	Name	Description
RP4	Remote representation of appliances in the home	Representation of appliances in the home network for access by the Operator Platform over a remote access protocol. It also provides access to, or pushes information about, local events from the HG software stack to the operator platform.
RP5	Local representation of appliances in the home	Support for local clients to communicate with the HG to access home appliances. Not elaborated in this document.
RP7	Reference Point in the cloud	Reference point between an Operator Platform and Service Applications in the Cloud
RP9	Physical Appliance Interface	The home area network (HAN) physical interface to appliances e.g. EchonetLite [10], ZigBee [11], EnOcean [12], IEEE 802.11, BlueTooth, DECT ULE 13], etc

Table 3: Reference points for external interfaces

The HG is shown above as a blackbox mediating between the appliances (accessible over RP9) and the environment. Appliances are represented over RP4 to the Operator Platform, which can control them and can also provide this representation to other parties inside the cloud (for example to service providers) over RP7. Cloud applications can also have direct communication with software on the HG, if allowed by the operator policy, but this communication is proprietary and out of scope for HGI.

The HG can also provide access and control of appliances over RP5 for local clients in the home network (e.g. smart phones, tablets or PCs showing a local home portal for the user). Such functionality will not be elaborated in this document.

Note that in addition to (hardware) appliances there can also be virtual appliances within the smart home software stack, visible through RP4 and/or RP5 (see also RP2 in section 5.4), which represent some functionality without requiring a hardware component. Examples of such virtual appliances are a weather station module that takes its data from the internet, or a clock which synchronizes with an NTP server. Both hardware appliances and

 © Home Gateway Initiative – 2015 – All rights reserved

virtual appliances are represented in the same way through the reference points and can be visualized, controlled or included in automation rules in the same way.

5.3 HG with specific Software Execution Environment (Open Platform 2.0)

Note: from this point, details can only be discussed in terms of a specific software environment. For the remainder of the document, the HGI Open Platform 2.0 environment is assumed.

The relationship of the smart home software to the execution environment on the HG is shown in the figure below.

Figure 3: Architecture showing native drivers

.

© Home Gateway Initiative – 2015 – All rights reserved

In the above Figure, the following entities are added:

Entity Name	Description
Native Drivers	A native driver is a piece of software that runs on the OS of the HG and provides an OS accessible interface to one or more hardware entities over specific home area network protocols.
HG Core	The HG Core is defined in [1] HG Requirements for *HGI Open Platform 2.0*.

Table 4: Software entities within the HG Software Stack

.

Figure 3 also introduces the following new reference point:

RP	Name	Description
RP6	Native Driver Interface	Represents the interface of the native drivers towards the smart home software

Table 5: Definition of RP6 within the HG Software Stack

5.4 Detailed HG Software Stack architecture

A more detailed view of the HG shows the software modules within the smart home software, including the reference points RP1, RP2 and RP8. The general requirements on the HG software stack (modularity, rights management, etc.) are explained in [1] *HG Requirements for HGI Open Platform 2.0*.

© Home Gateway Initiative – 2015 – All rights reserved

Figure 4: Architecture showing also reference points within the HG Software Stack

© Home Gateway Initiative – 2015 – All rights reserved

Open Platform 2.0 based implementations contain the following entities:

Name	Description
Open Platform	Open Platform 2.0 is an HGI concept for a protected, isolated software runtime environment on an HG where software modules can be installed and executed, as specified in [1]. The HG internal software architecture depicted in this architecture specification is based on OP2.0.
Appliance Drivers	Appliance Drivers provide the interfaces for technology specific HAN interfaces. This specification does not define or constrain in any way the underlying device model(s) used by the HAN interfaces.
Virtual Appliance Driver	The virtual appliance driver uses the same RP (RP2) as the Appliance Drivers in order to represent virtual appliances in the Device Abstraction Layer.
Device Abstraction Layer	The Device Abstraction Layer provides unified APIs, which can be defined in the context of a specific implementation such as (here) OP2.0, for application developers to control and query home appliances. It provides common, HAN technology agnostic, representations of appliances. The Device Abstraction Layer translates technology agnostic invocations of its APIs to technology-specific actions through the appropriate driver.
Adapter for Remote Representation	Provides remote access and control of the appliances in DAL.
Adapter for Local Representation	Allows local clients to access and control the appliances in DAL over the local home network.
Service Application (local)	Smart home software application that uses the APIs of RP1
WAN Network Connectivity Mgt	The WAN network module manages the configuration of the WAN connection(s) of the HG

Table 6: Definition of elements within the HG Software Stack

This view adds another 3 reference points:

RP	Name	Description
RP1	Device Abstraction Layer	The Device Abstraction Layer provides a common representation of appliances in the Home Domain to the Execution Environment, so that applications can be independent of the different home automation technologies. For example a KNX switch and a ZWave switch could be represented in the same way through an [8] *OSGi* Service, so that an application can switch both off without dealing with KNX/ZWave specifics.
RP2	Appliance Driver Interface	The appliance driver interface represents the functionality needed for the integration of a specific HAN technology, or a virtual appliance, in the Device Abstraction Layer.
RP8	WAN Network Connectivity Interface	The WAN Network Connectivity interface provides a description of the WAN connection(s) available on the HG, which is used by RP4.

Table 7: Reference points within the HG Software Stack

© Home Gateway Initiative – 2015 – All rights reserved

5.5 Access Control and Policies

Applications' access to appliance software interfaces, and therefore implicitly the user's access, may need to be controlled, but this must not limit physical user access in the home (e.g. via a wall switch). Additionally access control provides a way to manage concurrent access to the same appliance.

The implementation of access control depends on the actual Open Platform technology. Hence, this document cannot provide guidelines how to implement access control, but only high-level functional requirements.

An Open Platform based on OSGi provides some appropriate mechanisms such as the "Conditional Permission Admin", for example when access to an appliance driver by vendor A is only allowed for an application signed by vendor A. Note that this specification neither mandates nor precludes the use of Conditional Permission Admin.

The applications themselves may also have inherent access control, for example if the control of an oven through an application is only allowed in the case of local access.

Note that the access control requirements assume that the system supports identification and authorization of applications. Furthermore, it is advisable to include mechanisms to detect identity spoofing attempts. One possible way to provide application integrity verification and identification is to use digital signatures on application binaries.

However, the policies for access control basically depend on the operator's business model and platform implementation, and could range from none at all, to a very granular model. So providing tools for access control is required in this document, however specific access control policies are out of scope.

5.6 Appliance States

The following section describes the possible states of the internal representation of appliances in the DAL. When an appliance is initially detected by a HAN controller or on an IP interface, the DAL generates a notification at RP1 and (usually) the appliance transitions by means of a pairing/configuration process to the *Connected* state i.e. into the *Ready* state, wherein the appliance can send/receive data or execute requested actions. The appliance might later transition to the *Offline* state (because of energy-saving measures, or the user manually switching off the appliance, etc.) where no communication is possible, but it is expected that the appliance will return to the *Ready* state periodically. Whenever the appliance is in the *Ready* state, the DAL or software applications are permitted to request that an update of internal firmware is performed. While in the *Update* state, it cannot be assumed that the appliance is able to send/receive data. For various reasons (appliance is broken, long-term silence, failure to respond to communications, etc.), an appliance might be considered unsuitable for further operation: the DAL or application software may then label the appliance as being in the *Removed* state i.e. no further notifications/events/actions can be performed with it.

Appliances which have been placed in the *Removed* state must first be *Detected* and paired/configured again before they can become *Connected*. Appliances which are *Blocked* can be set by an application at any time to be in the *Ready* state.

In the figure below, every blue arrow indicates a change of state, which is usually accompanied by a notification via RP1 to higher-level software. The diagram just shows communication "readiness", not any internal operations and status of appliances. So according to the needs of this document and the diagram below, a complex appliance like a washing machine for example would simply be in the *Ready* state, and <u>not</u> a much more detailed state "beginning step 4 of the cold-wash-with-pre-soak cycle".

 © Home Gateway Initiative – 2015 – All rights reserved

Figure 5: Appliance state diagram, within the DAL.

State	Definition
Detected	A newly discovered appliance, waiting to be paired and/or configured, is in the D*etected* state.
Ready	A *Ready* appliance is one which is known to the system. It is configured, already paired (if applicable) and online, ready to respond to software requests for actions.
Offline	*Offline* appliances are either switched off or will come back online at a preset time (sometimes called Sleeping mode) or in response to an external trigger event; when the appliance is back online is immediately recognized as being in the *Ready* state.
Connected	When an application is either in the *Ready* or the *Offline* state, it is said to be in the (more general) *Connected* state. Connected appliances are known to the system, are configured, paired (if applicable) and ready to be used.
Blocked	The network configuration or DAL prevents transmission of messages to and from any appliance in the blocked state.
Removed	The appliance has been removed from the DAL. This can be needed in cases where it malfunctions (broken, never-responding) or is unwanted (obsolete, no longer needed). Removal can be triggered either by an application through RP1 or by the DAL itself based on a policy. A history function may record the time when the appliance was removed. When a removed appliance attempts to rejoin a network, software (policy) will determine if the reconnection attempt is allowed.

Table 8: Appliance states

Some appliances broadcast data without requiring configuring/pairing and hence can only be in the *Ready*, *Offline*, *Blocked* or the *Removed* states.

© Home Gateway Initiative – 2015 – All rights reserved

Previous HGI documents have discussed a *Standby* state for energy saving, whereby the appliance is listening to the network but most energy-consuming functions are set to the lowest or zero power: such appliances are by definition in this document in the *Ready* state (able to respond).

5.7 Grouping of Appliances

HGI members found that there are various common approaches to the grouping of appliances. For example:

- A number of separate "smart" lamps might be considered as a group, to be switched on/off together.
- The "on" state of (any) lamps in a house might be considered as a single input for a burglar detection system.
- If a lamp is connected by the user such that it can be activated by a specific "smart" switch, then the combination of "lamp and its switch" might be recorded as a "grouped appliance.
- All detected smart appliances (independent of their functionality) might be grouped in some applications as "my appliances".
- All detected appliances which had been recorded by the user (or other means) as having the property "in the kitchen" might be considered as a group. This depends on the intended use and the context in which the devices are used.

These topics are obviously complex. It must also be noted that a single devices might be a member of various groups, which could in some circumstances result in conflicting signaling. Deriving requirements for some/all of the approaches is beyond the scope of version 1.0 of this document.

© Home Gateway Initiative – 2015 – All rights reserved

6 Functional requirements

The following sections contain functional requirements for the different reference points of the system.

6.1 Requirements for RP1 Device Abstraction Layer

Note: all occurrences of "appliance" in the table below also include "virtual appliance".

No.	Requirement
1.	The API at RP1 MUST support all the mandatory requirements in [9] "*HG and Home Network Diagnostics*" sections §8.12 Device Discovery, §8.14 Topology Discovery, §8.15 Connectivity Testing
2.	The API at RP1 MUST support the command and control of smart home appliances.
3.	The API at RP1 MUST support reading and writing appliance data points.
4.	The API at RP1 MUST support applications receiving "data point change" events originating from appliances.
5.	The API at RP1 MUST support applications receiving “change of state” events originating from appliances (e.g. *Offline*, *Ready*, etc) and from the HAN controller (e.g. appliance *Detected*, appliance *Removed,* appliance *Blocked*).
6.	The API at RP1 MUST support retrieving the last known state change of an appliance, including the local time at which it occurred.
7.	The API at RP1 SHOULD be able to report a history of state changes for an appliance, with an associated local time for each change, with the most recent value at the beginning of the list. The length of the list MUST be configurable.
8.	The API at RP1 MUST be able to provide a list of all appliances in the *Ready* and/or *Offline* state. Note that this requirement does not mandate polling of all appliances at the time of the request.
9.	The API at RP1 MUST support triggering discovery of appliances. Discovery MUST be able to be disabled. If an attached HAN cannot support the function then a corresponding notification MUST be generated, including an identifier for that HAN.
10.	The API at RP1 MUST support (logically) removing an appliance i.e. move its state to *Removed*, so that applications no longer read/write to the that appliance.
11.	The API at RP1 MUST support *Blocking* an appliance (i.e. move its state to *Blocked*) so that applications no longer read/write to the deviceappliance. This function SHOULD operate at the lowest possible layer, e.g. to prevent the appliance from transmitting on the network.
12.	The API at RP1 MUST support updating appliance firmware, including notifying applications that the appliance is in the *Updating* state.
13.	The API at RP1 MUST associate a single unique identifier with each appliance that has been detected. The identifier MUST be unique in the scope of the HG and all its HANs. The identifier MUST remain the same after the HG or the Appliance reboots or undergoes a firmware update, at least within the scope of the history function.

6.2 Requirements for RP1 Access Control

No.	Requirement
14.	RP1 MUST support appliance access control.
15.	Access control MUST be able to be invoked per application.
16.	The same access control policy SHOULD be able to be applied to a specified list of

 © Home Gateway Initiative – 2015 – All rights reserved

	applications.
17.	Access control for a given application MUST be able to be invoked on a per action basis.
18.	Access control MUST be able to specify read/write access on an individual data point basis.

6.3 Requirements for DAL

No.	Requirement
19.	The DAL MUST be able to support additional HAN technologies without changes to the API at RP1 and SHOULD NOT • require a complete reboot • interrupt services e.g. a burglar alarm service while doing so.
20.	Changes to the DAL SHOULD NOT • require a complete reboot • interrupt services e.g. a burglar alarm service
21.	Changes to the DAL HANs SHOULD NOT require a complete HG firmware update.
22.	The DAL MUST be able to • detect flooding of notifications from appliances prevent notifications from a given appliance being transmitted to applications • *Block* a given appliance. • un*Block* an appliance The parameters defining "flooding" SHOULD be configurable for the DAL environment. The notifications of changes in the *Blocked* state SHOULD include indications of the cause.
23.	Upon being notified via RP2 of a new HAN network, the DAL MUST initiate the associated discovery process.

6.4 Requirements for RP2 Appliance Driver Interface

Note that this document does not consider specific requirements for the profiling of particular HAN protocols, so that they can be integrated on the HG. Examples of such detailed requirements would be the minimum level of security that the protocol has to support, or a specific function that is required for the successful integration in the Device Abstraction Layer and the Remote Protocol.

No.	Requirement
24.	The API at RP2 MUST support both hardware and virtual appliances.
25.	The API at RP2 MUST provide a persistent unique identifier for each connected appliance within a specific HAN network.
26.	The API at RP2 MUST support rejecting connection and/or pairing requests.
27.	The successful installation of a new appliance driver MUST be notified to the DAL.
28.	RP2 SHOULD notify DAL upon detecting a (new) home area network (HAN),
29.	The API at RP2 MUST support notification of changes of appliance state to the DAL.

Appliance identifiers must be unique in the scope of all the HAN(s) for which an appliance driver is responsible i.e. even if there are two separate ZigBee networks in a big house, the RP2 identifier id for a light switch on ZigBee network 1 must be unique across both ZigBee networks.

© Home Gateway Initiative – 2015 – All rights reserved

6.5 Requirements for RP4 Remote Representation

No.	Requirement
30.	RP4 MUST support the remote command and control of smart home appliances.
31.	RP4 MUST support the remote reading and writing of appliance data points.
32.	RP4 MUST support subscribing, unsubscribing to and forwarding of “data point change” and “change of state” event notifications from the HG Software Stack to the Operator Platform
33.	RP4 MUST support remote applications to get the last known state change of one or more appliances, together with the associated local time(s)
34.	
35.	RP4 MUST provide a mechanism for the immediate transfer of notifications to the operator platform.
36.	RP4 MUST support suppressing specific types of notifications, as configured by the operator platform.
37.	RP4 SHOULD support reporting to the operator platform a history of state changes for an appliance, with an associated local time for each change, with the most recent value at the beginning of the list. The length of such a list MUST be configurable.
38.	RP4 MUST be able to provide to the operator platform a list of all appliances in the *Ready* and/or *Offline* state. Note that this requirement does not mandate polling of all appliances at the time of the request.
39.	RP4 MUST be able to provide a list of all appliances in the *Ready* and/or *Offline* state.
40.	RP4 MUST support triggering discovery of appliances. Discovery MUST be able to be disabled. If an attached HAN cannot support the function then a corresponding notification MUST be generated, including an identifier for that HAN.
41.	RP4 MUST include in the appliance information the persistent locally unique identifier assigned by the Device Abstraction Interface.
42.	The API at RP4 MUST support the operator platform removing an appliance, so that applications shall no longer read/write to the appliance.
43.	RP4 MUST support a remote command to *Block* one or more appliances.
44.	The API at RP4 MUST support updating appliance firmware, including notifications that the appliance is in the Updating state.
45.	The API at RP4 must provide an unambiguous mapping between unique appliance identifiers used in the operator platform and those identifiers used at RP1.

6.6 Requirements for RP5 Local representation of appliances

Requirements for the RP5 interface are out of scope of the current version of this document.

.

6.7 Requirements for RP6 Native Driver Interface

The requirements for this interface are out of scope for the current version of this document.

© Home Gateway Initiative – 2015 – All rights reserved

6.8 Requirements for RP7 Reference point in the cloud

The specific requirements for this interface are out of scope for the current version of this document.

6.9 Requirements for RP8 WAN Network Connectivity Interface

The specific requirements for the RP8 interface may be elaborated in a future version of this document. Here it is sufficient to note that, if the broadband WAN access is cellular then HG smart home applications may need to detect which cellular network characteristics apply, e.g. whether data goes via GSM or 4G networks, and whether this changes over time. Another example could be to manage the connection to the WAN, e.g. for applications to trigger a re-association over the cellular connection.

It is important to stress that RP8 is not the means to open new connections. Applications can establish and use connections to the cloud without being aware of RP8. RP8 only gives additional information about the connection characteristics, so that applications can adapt their behavior accordingly.

© Home Gateway Initiative – 2015 – All rights reserved

7 References

[1] *HG Requirements for HGI Open Platform 2.0*
HGI Identifier: HGI-RD048
www.homegateway.org

[2] HGI-RD039 "*Requirements for wireless home area networks (WHANS) supporting smart home services*"
http://www.homegatewayinitiative.org/publis/RD039-Req-for-Wireless-home-area-networks.pdfservices

[3] HGI GD-017-R3 Use Cases and Architecture for Home energy Management in Figure 7
http://www.homegatewayinitiative.org/publis/GD-017-R3_use-cases-and-architecture-for-home-energy-Management-service.pdf

[4] *HGI-GWD035 Smart Home Use Cases*
HGI Identifier: 035
www.homegateway.org

[5] *ISO/IEC FDIS 15067-3 Home Electronic System (HES) application model — Part 3: Model of a demand-response energy management system for HES*
http://www.iso.org

[6] *HGI Residential Profile,* HGI-RD001-R2.01
http://www.homegatewayinitiative.org/publis/RD-001-R2-01_HG-Tech-Req-Residential_V1-01.pdf

[7] HGI-RD048 *HG Requirements for HGI Open Platform 2*
http://www.homegatewayinitiative.org/publis/HGI-RD048-HG_Requirements_for_HGI_Open_Platform_2_0_published_text.pdf

[8] *OSGi Service Platform Core Specification, Release 4, Version 4.2 or later*
www.osgi.org

[9] HGI-RWD016-R3 *HG and Home Network Diagnostics Module Requirements*, April, 2013
http://www.homegatewayinitiative.org/publis/HGI-RD016_HG-Home-Network-Diag-Modul-Req.pdf

[10] *Echonet Lite*
http://www.echonet.gr.jp/english/spec/spec_v101_lite.htm

[11] *Zigbee*
http://www.zigbee.org/en/spec_download/download_request.asp and https://docs.zigbee.org/zigbee-docs/dcn/07/docs-07-5367-02-0afg-home-automation-profile-for-public-download.pdf

[12] *EnOcean*
http://www.enocean-alliance.org/en/enocean_standard/

[13] DECT Ultra-Low-Energy (ULE)
http://www.ulealliance.org/specifications

[14] *AAL Project Smart Senior*
http://www1.smart-senior.de/dyn/mediaout.dhtml/-/server/Presse/Pressematerialien/Pressematerialien-Liste/201102_SmS-Broschuere/SmS-Broschuere/@teaser_long/@s4/SmartSenior_Projektbroschre_2012.pdf

 © Home Gateway Initiative – 2015 – All rights reserved