



**Publicly Available Specification (PAS);
Smart Machine-to-Machine communications (SmartM2M)
Home Gateway Initiative RD036-Smart Home architecture and
system requirements**

CAUTION

The present document has been submitted to ETSI as a PAS produced by HGI and approved by the ETSI Technical Committee Smart Machine-to-Machine communications (SmartM2M).

HGI was owner of the copyright of the document (RD036) and/or had all relevant rights and had assigned said rights to ETSI on an "as is basis". Consequently, to the fullest extent permitted by law, ETSI disclaims all warranties whether express, implied, statutory or otherwise including but not limited to merchantability, non-infringement of any intellectual property rights of third parties. No warranty is given about the accuracy and the completeness of the content of the present document.

Reference

DTS/smartM2M-103424

Keywords

gateway, home gateway, intelligent home & buildings

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important noticeThe present document can be downloaded from:
<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:
<https://portal.etsi.org/People/CommiteeSupportStaff.aspx>

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2016.
All rights reserved.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.
3GPP™ and **LTE™** are Trade Marks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.
GSM® and the GSM logo are Trade Marks registered and owned by the GSM Association.

Contents

Foreword.....	4
Modal verbs terminology.....	4
1 Scope	5
2 References	5
2.1 Normative references	5
2.2 Informative references.....	5
3 Definitions, symbols and abbreviations	6
3.1 Definitions.....	6
3.2 Abbreviations	7
3.3 Relationship with other HGI Specifications	7
4 Business requirements and Use Cases.....	8
5 Architecture.....	8
6 Overview	8
6.1 High-level view of smart home architecture	9
7 HG with specific software execution environment (Open Platform 2.0).....	12
7.1 Detailed HG software stack architecture	13
7.2 Access Control and Policies	14
7.3 Appliance States	15
7.4 Grouping of Appliances	16
8 Functional requirements	17
8.1 Requirements for RP1 Device Abstraction Layer	17
8.2 Requirements for RP1 Access Control.....	18
8.3 Requirements for DAL.....	18
8.4 Requirements for RP2 Appliance Driver Interface	18
8.5 Requirements for RP4 Remote Representation	19
8.6 Requirements for RP5 Local representation of appliances.....	20
8.7 Requirements for RP6 Native Driver Interface	20
8.8 Requirements for RP7 Reference point in the cloud	20
8.9 Requirements for RP8 WAN Network Connectivity Interface	20
History	21

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<https://ipr.etsi.org/>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Smart Machine-to-Machine communications (SmartM2M), as result of the PAS process for document HGI-RD036 developed by the Home Gateway Initiative.

The Home Gateway Initiative, a non-profit organization closed on June 2016, produced guidelines, requirements documents, white papers, vision papers, test plans and other documents concerning broadband equipment and services which are deployed in the home.

HGI worked on Specifications for home connectivity and Services enablement, in particular to encompass a delivery framework for Smart Home services. The defined architecture includes support for a standard, general purpose software execution environment in the HG (for third party applications), API definitions, device abstraction and interfacing with Cloud based platforms.

The HGI's methodology ensured that projects undertaken reflected items of strong interest to the Broadband Service Providers (BSPs), as well as brought in opportunities at every stage for vendor input, suggestions and participation.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

1 Scope

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.

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

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at <http://docbox.etsi.org/Reference>.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

Not applicable.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] ETSI TS 103 426 (11-2016): "Publicly Available Specification (PAS); Smart Machine-to-Machine communications (SmartM2M) Home Gateway Initiative RD048-HG Requirements For HGI Open Platform 2.1".
- [i.2] ETSI TS 103 425 (11-2016): "Publicly Available Specification (PAS); Smart Machine-to-Machine communications (SmartM2M) Home Gateway Initiative RD039-Requirements for Wireless Home Area Networks (WHANs) Supporting Smart Home Services".

- [i.3] HGI GD-017-R3 (August 2011): "Use Cases and Architecture for a Home Energy Management Service".
- NOTE: http://www.homegatewayinitiative.org/userfiles/file/downloads/GD-017-R3_use-cases-and-architecture-for-home-energy-Management-service.pdf.
- [i.4] HGI-GWD035: "Smart Home Use Cases".
- NOTE: www.homegateway.org
- [i.5] ISO/IEC FDIS 15067-3: "Home Electronic System (HES) application model - Part 3: Model of a demand-response energy management system for HES".
- NOTE: <http://www.iso.org>
- [i.6] HGI-RD044 Home Gateway Base Requirements: "Residential Profile 2".
- NOTE: http://www.homegatewayinitiative.org/userfiles/file/downloads/P_HGI02723R11.pdf
- [i.7] Void.
- [i.8] "OSGi Service Platform Core Specification", Release 4, Version 4.2 or later.
- NOTE: www.osgi.org
- [i.9] HGI-RWD016-R3 (April, 2013): "HG and Home Network Diagnostics Module Requirements".
- NOTE: http://www.homegatewayinitiative.org/userfiles/file/downloads/HGI-RD016_HG-Home-Network-Diag-Modul-Req.pdf.
- [i.10] "Echonet Lite".
- NOTE: https://echonet.jp/spec_v112_lite_en/
- [i.11] "Zigbee™".
- NOTE: <http://www.zigbee.org/zigbee-for-developers/applicationstandards/zigbeehomeautomation/>.
- [i.12] "EnOcean".
- NOTE: http://www.enocean-alliance.org/en/enocean_standard/.
- [i.13] "DECT Ultra-Low-Energy (ULE)".
- NOTE: <http://www.ulealliance.org/specifications>.
- [i.14] Void.
- [i.15] Void.
- [i.16] IEEE 802.11™: "Wireless LANs".

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in Open Platform 2.0 [i.1] and the following apply:

action: A function that can be executed by an appliance.

appliance: A domestic electrical/mechanical machine which accomplishes some function used in smart home services.

data point: A property of an appliance, which can change while the appliance is in operation.

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.

virtual appliance: An appliance implemented purely in software.

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.

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

Service 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: Is a business entity. The Service Provider supplies the necessary means to provide the business related support of a specific Service Application.

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.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AAL Ambient Assisted Living

NOTE: 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
NTP	Network Time Protocol
OS	Operating System
OSGi	OSGi Alliance
RP	Reference Point
SDT	Smart Device Template

NOTE: A HGI term related to appliance function modelling.

ULE	Ultra Low Energy
WAN	Wide Area Network

3.3 Relationship with other HGI Specifications

The present document contains requirements over and above those in the HGI Residential Profile [i.6], Open Platform 2.0 [i.5] and Wireless HAN Services [i.2]. All the requirements of those documents still apply, except that support of Open Platform 2.0 is only mandatory with regard to clause 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 [i.3] is one of the possible appliances in the home network

4 Business requirements and Use Cases

The business requirements and use cases used as the basis for the architecture described in the present document are described in [i.4].

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 table 1 for convenience.

Table 1: Overview of Reference Points

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 via 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

6 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.

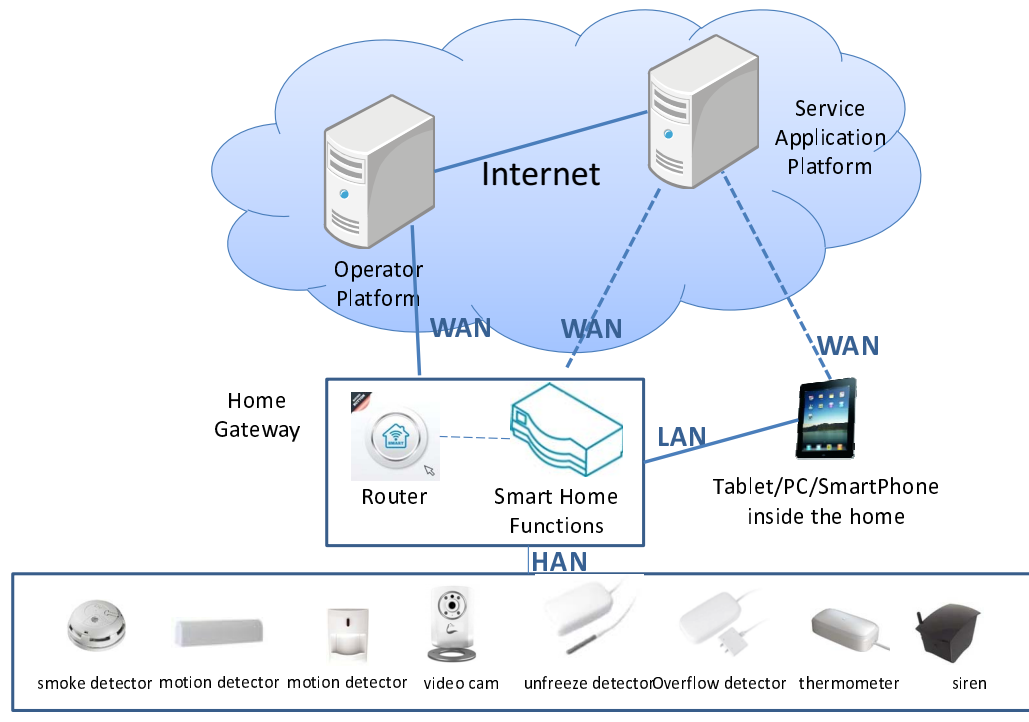


Figure 1: HGI smart home architecture

The present 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 figure 1 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 the present document.

6.1 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 Figure 2.

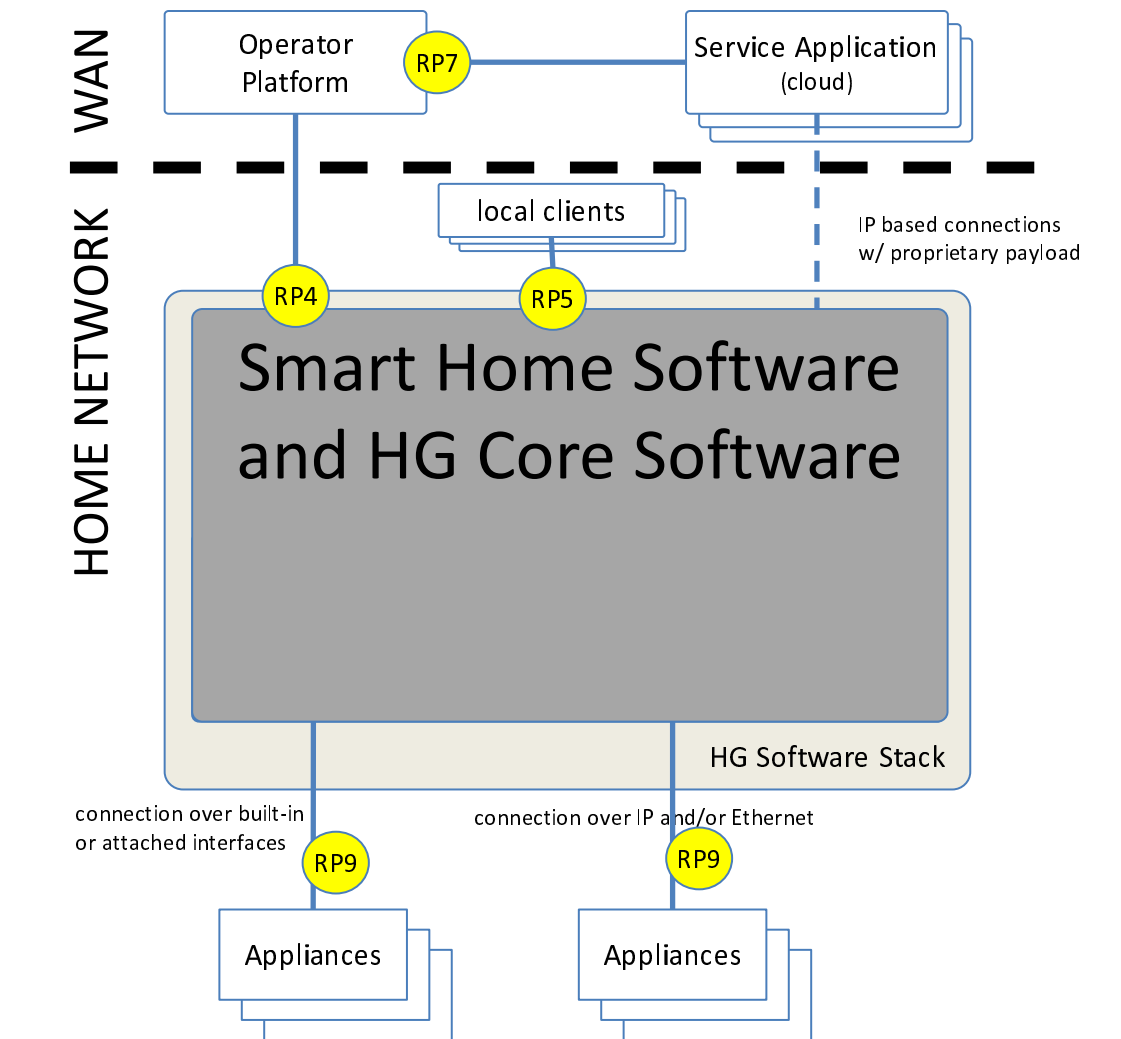


Figure 2: Reference points between the HG software and software in the Cloud or local client software

Figure 2 refers to the following entities:

Table 2: Software and hardware elements in figure 2

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 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 [i.5] and any dynamically loaded software modules (e.g. 3 rd 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 [i.1] : 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 [i.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.

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

Table 3: 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 the present 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™ [i.10], ZigBee™ [i.11], EnOcean™ [i.12], IEEE 802.11™ [i.16], BlueTooth™, DECT ULE [i.13], etc.

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 the present 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 clause 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 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.

7 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 3.

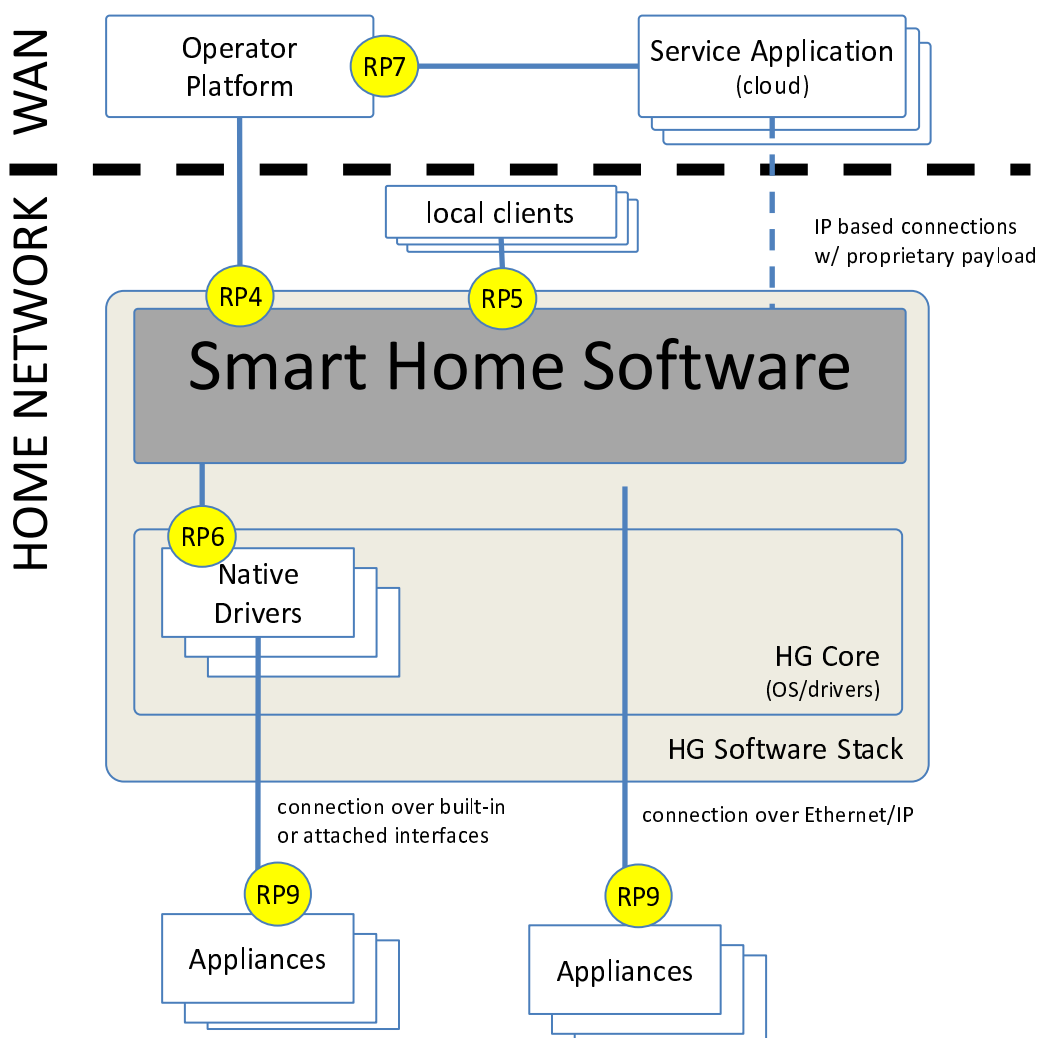


Figure 3: Architecture showing native drivers

In Figure 3, the following entities are added:

Table 4: Software entities within the HG Software Stack

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 ETSI TS 103 426 [i.1]

Figure 3 also introduces the following new reference point:

Table 5: Definition of RP6 within the HG Software Stack

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

7.1 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 ETSI TS 103 426 [i.1].

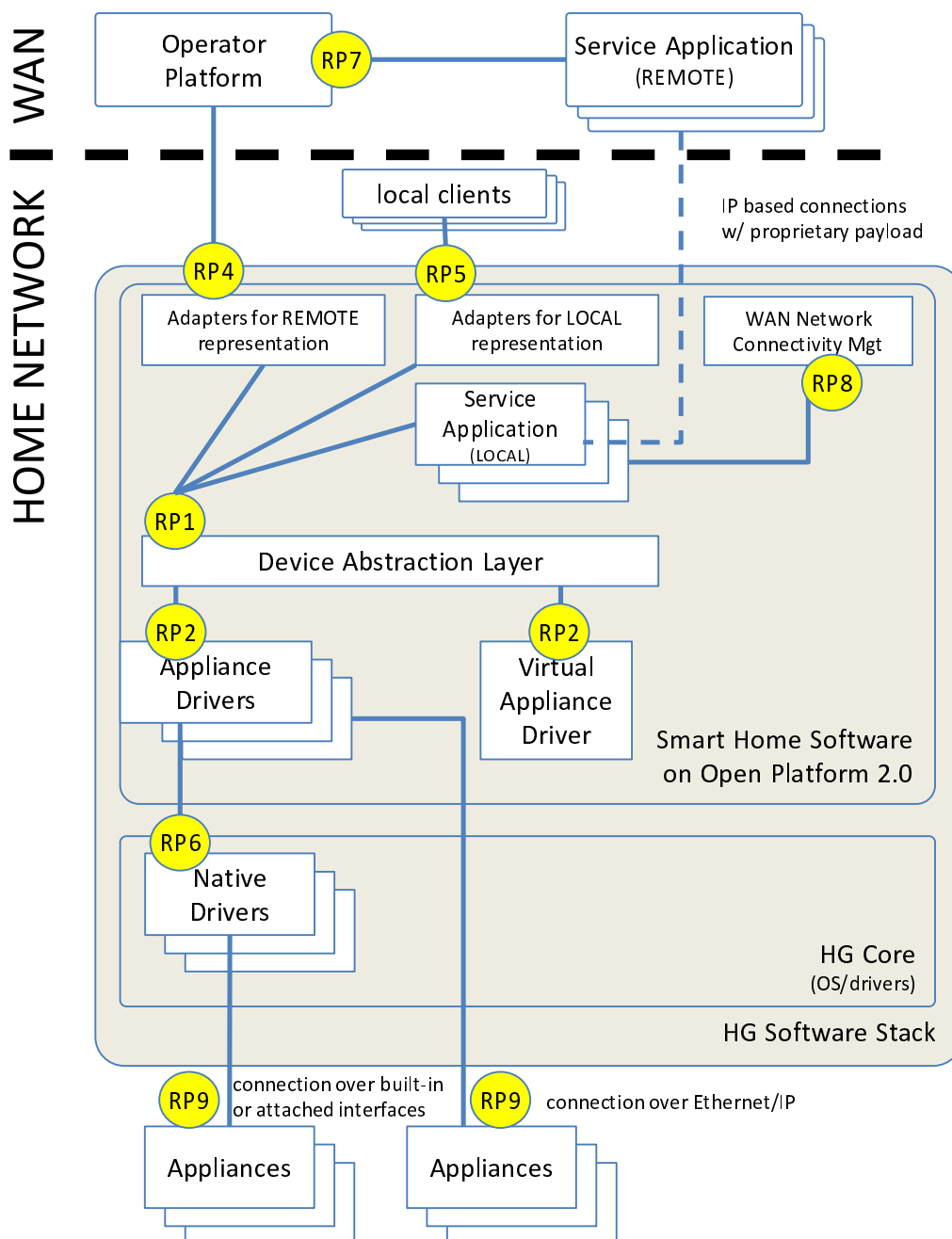


Figure 4: Architecture showing also reference points within the HG Software Stack

Open Platform 2.0 based implementations contain the following entities:

Table 6: Definition of elements within the HG Software Stack

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 [i.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 document 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.

This view adds another 3 reference points:

Table 7: Reference points within the HG Software Stack

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 [i.8]. 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.

7.2 Access Control and Policies

Applications' access to appliance software interfaces and therefore implicitly the user's access, may need to be controlled, but this shall 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, the present 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 the present document 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 the present document, however specific access control policies are out of scope.

7.3 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 shall 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 figure 5, 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 the present document and the figure 5 a complex appliance like a washing machine for example would simply be in the *Ready* state and **not** a much more detailed state "beginning step 4 of the cold-wash-with-pre-soak cycle".

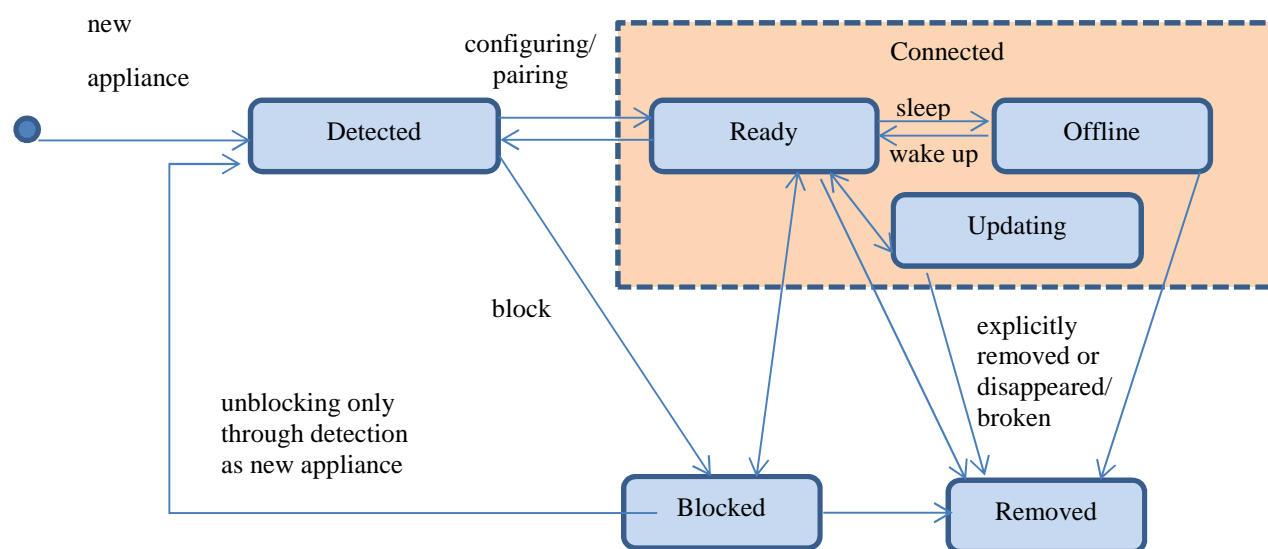


Figure 5: Appliance state diagram, within the DAL

Table 8: Appliance states

State	Definition
Detected	A newly discovered appliance, waiting to be paired and/or configured, is in the Detected 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 re-join a network, software (policy) will determine if the reconnection attempt is allowed.

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

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 the present document in the *Ready* state (able to respond).

7.4 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. 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 the present document.

8 Functional requirements

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

8.1 Requirements for RP1 Device Abstraction Layer

NOTE: All occurrences of "appliance" in table 9 also include "virtual appliance".

Table 9

N°	Requirement
ARCH1. a	The API at RP1 SHALL support all the mandatory requirements in [i.9] " <i>HG and Home Network Diagnostics</i> " clause 8.12 Device Discovery, §8.14 Topology Discovery, clause 8.15 Connectivity Testing
ARCH2.	The API at RP1 SHALL support the command and control of smart home appliances.
ARCH3.	The API at RP1 SHALL support reading and writing appliance data points.
ARCH4.	The API at RP1 SHALL support applications receiving "data point change" events originating from appliances.
ARCH5.	The API at RP1 SHALL support applications receiving "change of state" events originating from appliances (e.g. <i>Offline</i> , <i>Ready</i> , etc.) and from the HAN controller (e.g. appliance <i>Detected</i> , appliance <i>Removed</i> , appliance <i>Blocked</i>).
ARCH6.	The API at RP1 SHALL support retrieving the last known state change of an appliance, including the local time at which it occurred.
ARCH7.	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 SHALL be configurable.
ARCH8.	The API at RP1 SHALL be able to provide a list of all appliances in the <i>Ready</i> and/or <i>Offline</i> state. Note that this requirement does not mandate polling of all appliances at the time of the request.
ARCH9.	The API at RP1 SHALL support triggering discovery of appliances. Discovery SHALL be able to be disabled. If an attached HAN cannot support the function then a corresponding notification SHALL be generated, including an identifier for that HAN.
ARCH10.	The API at RP1 SHALL support (logically) removing an appliance i.e. move its state to <i>Removed</i> , so that applications no longer read/write to the that appliance.
ARCH11.	The API at RP1 SHALL support <i>Blocking</i> an appliance (i.e. move its state to <i>Blocked</i>) so that applications no longer read/write to the device appliance. This function SHOULD operate at the lowest possible layer, e.g. to prevent the appliance from transmitting on the network.
ARCH12.	The API at RP1 SHALL support updating appliance firmware, including notifying applications that the appliance is in the <i>Updating</i> state.
ARCH13.	The API at RP1 SHALL associate a single unique identifier with each appliance that has been detected. The identifier SHALL be unique in the scope of the HG and all its HANs. The identifier SHALL remain the same after the HG or the Appliance reboots or undergoes a firmware update, at least within the scope of the history function.

8.2 Requirements for RP1 Access Control

Table 10

N°	Requirement
ARCH14.	RP1 SHALL support appliance access control.
ARCH15.	Access control SHALL be able to be invoked per application.
ARCH16.	The same access control policy SHOULD be able to be applied to a specified list of applications.
ARCH17.	Access control for a given application SHALL be able to be invoked on a per action basis.
ARCH18.	Access control SHALL be able to specify read/write access on an individual data point basis.

8.3 Requirements for DAL

Table 11

N°	Requirement
ARCH19.	The DAL SHALL be able to support additional HAN technologies without changes to the API at RP1 and SHOULD NOT <ul style="list-style-type: none"> require a complete reboot interrupt services e.g. a burglar alarm service while doing so.
ARCH20.	Changes to the DAL SHOULD NOT <ul style="list-style-type: none"> require a complete reboot interrupt services e.g. a burglar alarm service
ARCH21.	Changes to the DAL HANs SHOULD NOT require a complete HG firmware update.
ARCH22.	The DAL SHALL be able to <ul style="list-style-type: none"> detect flooding of notifications from appliances prevent notifications from a given appliance being transmitted to applications Block a given appliance. unBlock 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.
ARCH23.	Upon being notified via RP2 of a new HAN network, the DAL SHALL initiate the associated discovery process.

8.4 Requirements for RP2 Appliance Driver Interface

Note that the present 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.

Table 12

N°	Requirement
ARCH24.	The API at RP2 SHALL support both hardware and virtual appliances.
ARCH25.	The API at RP2 SHALL provide a persistent unique identifier for each connected appliance within a specific HAN network.
ARCH26.	The API at RP2 SHALL support rejecting connection and/or pairing requests.
ARCH27.	The successful installation of a new appliance driver SHALL be notified to the DAL.
ARCH28.	RP2 SHOULD notify DAL upon detecting a (new) home area network (HAN),
ARCH29.	The API at RP2 SHALL support notification of changes of appliance state to the DAL.

Appliance identifiers SHALL 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 SHALL be unique across both ZigBee networks.

8.5 Requirements for RP4 Remote Representation

Table 13

N°	Requirement
ARCH30.	RP4 SHALL support the remote command and control of smart home appliances.
ARCH31.	RP4 SHALL support the remote reading and writing of appliance data points.
ARCH32.	RP4 SHALL 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
ARCH33.	RP4 SHALL support remote applications to get the last known state change of one or more appliances, together with the associated local time(s)
ARCH34.	RP4 SHALL provide a mechanism for the immediate transfer of notifications to the operator platform.
ARCH35.	RP4 SHALL support suppressing specific types of notifications, as configured by the operator platform.
ARCH36.	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 SHALL be configurable.
ARCH37.	RP4 SHALL be able to provide to the operator platform a list of all appliances in the <i>Ready</i> and/or <i>Offline</i> state. Note that this requirement does not mandate polling of all appliances at the time of the request.
ARCH38.	RP4 SHALL be able to provide a list of all appliances in the <i>Ready</i> and/or <i>Offline</i> state.
ARCH39.	RP4 SHALL support triggering discovery of appliances. Discovery SHALL be able to be disabled. If an attached HAN cannot support the function then a corresponding notification SHALL be generated, including an identifier for that HAN.
ARCH40.	RP4 SHALL include in the appliance information the persistent locally unique identifier assigned by the Device Abstraction Interface.
ARCH41.	The API at RP4 SHALL support the operator platform removing an appliance, so that applications shall no longer read/write to the appliance.
ARCH42.	RP4 SHALL support a remote command to <i>Block</i> one or more appliances.
ARCH43.	The API at RP4 SHALL support updating appliance firmware, including notifications that the appliance is in the Updating state.
ARCH44.	The API at RP4 SHALL provide an unambiguous mapping between unique appliance identifiers used in the operator platform and those identifiers used at RP1.

8.6 Requirements for RP5 Local representation of appliances

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

8.7 Requirements for RP6 Native Driver Interface

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

8.8 Requirements for RP7 Reference point in the cloud

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

8.9 Requirements for RP8 WAN Network Connectivity Interface

The specific requirements for the RP8 interface may be elaborated in a future version of the present 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.

History

Document history		
V1.1.1	November 2016	Publication