HTNG Intelligent Guest Room Specification Version 2.0

19 April 2013

About HTNG

Hotel Technology Next Generation (HTNG) is a non-profit association with a mission to foster, through collaboration and partnership, the development of next-generation systems and solutions that will enable hoteliers and their technology vendors to do business globally in the 21st century. HTNG is recognized as the leading voice of the global hotel community, articulating the technology requirements of hotel companies of all sizes to the vendor community. HTNG facilitate the development of technology models for hospitality that will foster innovation, improve the guest experience, increase the effectiveness and efficiency of hotels, and create a healthy ecosystem of technology suppliers.

Manufacturers and software providers shall not claim compliance with portions of the requirements of any HTNG specification or standard, and shall not use the HTNG name or the name of the specification or standard in any statements about their respective product(s) unless the product(s) is (are) certified as compliant to the specification or standard.

Permission is granted for implementers to use the names, labels, etc. contained within the specification. The intent of publication of the specification is to encourage implementations of the specification.

This specification has not been verified for avoidance of possible third-party proprietary rights. In implementing this specification, usual procedures to ensure the respect of possible third-party intellectual property rights should be followed. Visit http://htng.org/ip-claims to view third-party claims that have been disclosed to HTNG. HTNG offers no opinion as to whether claims listed on this site may apply to portions of this specification.

The names Hotel Technology Next Generation and HTNG, and logos depicting these names, are trademarks of Hotel Technology Next Generation. Permission is granted for implementers to use the aforementioned names in technical documentation for the purpose of acknowledging the copyright and including the notice required above. All other use of the aforementioned names and logos requires the permission of Hotel Technology Next Generation, either in written form or as explicitly permitted for the organization’s members through the current terms and conditions of membership.

Table of contents

1 This Specification at a Glance

This specification is used by hotel information systems to determine the operational status or health of guest room devices. Described are message exchange patterns that make it possible to request the status of guest room devices (ad-hoc query) and register to receive notification messages (status change events) when the health of devices in a guest room change, as well as messages for registering and discovering devices.

This Intelligent Guest Room specification is the bridge between the HTNG Guest & Room Messaging Status (GRSM) specification and the HTNG Device Messaging Structure (DMS) specification. The GRSM specification is a high-level, guest stay-related informational messaging, whereas the DMS specification is a low-level device protocol.

2 Document Information

2.1 Document History

Version	Date	Author	Comments
1.0	19 Oct 2012	Intelligent Guest Room (IGR) Workgroup	Released spec – addressed Guest Room Status Check and Guest Room Status Event Notification use cases
1.01	28 Oct 2012	IGR Workgroup	Added Registration and Discovery scenarios
1.02	12 Dec 2012	IGR Workgroup	Updated Sections 1, 2 & 5
1.03-1.08	09 Jan 2013 – 13 Mar 2013	IGR Workgroup	Updated sample messages
1.90	21 Mar 2013	Kylene Reese	Prepared for member review period
1.95	10 Apr 2013	Kylene Reese	Prepared spec for Workgroup vote
2.0	19 Apr 2013	IGR Workgroup	Release spec – includes centralized registration and discovery of in-room devices

2.2 Document Purpose

More and more, hoteliers are expected to provide guests with all the technology, connectivity and devices that they are used to experiencing in their home environment. Advanced in-room technologies have become a requirement rather than a luxury. But creating that at-home experience is much more difficult for hoteliers than it is for their tech-savvy guests. For the consumer, installing devices and connecting systems is a simple, straight-forward process. However, hoteliers must face the challenge of implementing and maintaining an on-property infrastructure that supports in-room devices and gives all their guests the at-home experience they expect. This specification provides hoteliers with web services that facilitate the deployment and management of their in-room technology infrastructure.

2.3 Scope

This document targets the release of a specification providing the necessary components, including design recommendations, WSDLs, XSDs and scenarios allowing a hospitality operator or hospitality system provider to design and deploy a set of device-based services in-line with best practices, methods and procedures that meet or exceed current industry web services standards specifications. Basic functionality of the Intelligent Guest Room services makes it possible to remotely monitor the status of devices in a room. This specification addresses the ability to remotely query the current and last status of a device or group of devices and subscribe to device status change notifications as well as the ability to register and discover devices.

2.4 Relationship to Other Standards

This specification and its supporting schemas leverage the existing OpenTravel Alliance methodology for message construction and incorporate a sub set of applicable OpenTravel Alliance specifications. This ensures a consistent representation of data across the various HTNG specifications.

Please refer to the HTNG Protocol & Messaging Transport Event Notification specification for details on how to subscribe to and manage notifications defined in this specification.

2.5 Useful Resources

- Implementing Web Services Using HTNG Specifications – A Quick Start Guide for Software Developers
- HTNG Discussion Board – currently available at http://www2.htng.org/discussion

2.6 Audience

This document is intended to aid in the design, integration and deployment of hospitality systems that require interaction with in-room devices in order to monitor room-based technologies. The document specifically targets hospitality systems developers, integrators and operators.

2.7 Overview

Content contained in this document is intended to assist in the design, implementation and integration of a set of applications and services whose feature set is based on the status of in-room device technologies. A brief overview of each section is as follows:

Section 3 – Scenarios

This section includes the scenario overviews, diagrams, roles, use cases, data element tables and sample messages regarding Guest Room Status Check, Guest Room Status Event Notification, Device Registration, Device Capability Profile Management and Device Discovery. The technical artifacts (XSDs and WSDLs) for this certification release of IGR v2.0 can be found in a separate ZIP file included with specification.

Section 4 – Messages

Detailed Data Element tables and sample messages are provided in this section.

Section 5 – Appendices

This section includes terms, implementation requirements, links, any referenced documents, as well as common HTNG schema components referenced in this document.

2.8 Known Limitations

There are no known limitations of this specification at the time of release.

2.9 Further Considerations

This specification references the functional set of the HTNG Device Messaging Structure (DMS) specification, with the intent of providing enterprise level web service access to devices. Future revisions may provide additional guidance in implementing the DMS spec in a standard fashion.

3 Component Scenarios

The Intelligent Guest Room (IGR) system is a set of standards that make it possible to enhance the guest experience through the use of integrated guest room device technologies. The IGR system encompasses a set of web service standards that make it possible to obtain information from intelligent guest room devices. Using this IGR specification, vendors and/or hoteliers are able to acquire information about the operational status of devices in a guest room in order to minimize guest impact when guest room technology-related problems arise.

3.1 Guest Room Status Check

3.1.1 Overview

In order to enhance the guest experience, a Hotel Information System is required to check the status of guest room technologies prior to room assignment. This may be completed at check-in, pre-arrival or even during the stay. The goal of the status check is to ensure that guest room technologies are functioning properly.

3.1.2 Roles

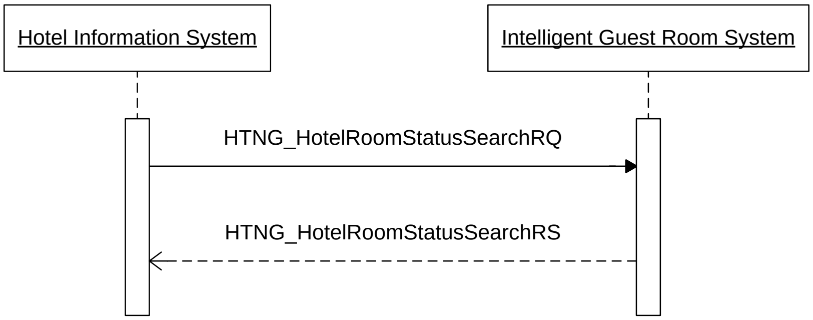
Role	Description	Examples
Hotel Information System	Any hotel software application or service containing features sets based on the status of guest room devices.	• PMS • Work order management • Device management console
Intelligent Guest Room (IGR) System	A system that provides the ability to query the status of guest room devices.	• Device Gateway • Zone Controller

3.1.3 Use Case

Assumption:	IGR System is fully functional and accessible.
Pre-conditions:	None
Trigger:	Request for room status.
Basic Course of Events:	1) The Hotel Information System constructs an appropriate request message and sends a status check request to the IGR System. 2) The IGR System enumerates guest room devices in order to determine their operational status. 3) The IGR System returns the operational status to the Hotel Information System.
Post-condition:	Hotel Information System knows the room device status.

Exception Paths:	• IGR System returns an error status indicating one or more devices are not functioning properly. The error status message includes a severity indicator used by the requesting Hotel Information system to determine a course of action. • Failure of IGR System to return a response.
Alternative Path:	None

3.1.4 Message Flows


Hotel Information System
Intelligent Guest Room System
HTNG_HotelRoomStatusSearchRQ
HTNG_HotelRoomStatusSearchRS

3.1.5 Sample Request

```
<HTNG_HotelRoomStatusSearchRQ EchoToken="Echo12345" TimeStamp="2012-08-17T09:30:47Z"
Version="1.0">
        <POS>
                <Source>
                        <RequestorID Type="10" ID_Context="ACME_PMS" ID="PMS1">
                                <CompanyName>Acme PMS Company</CompanyName>
                        </RequestorID>
                </Source>
        </POS>
        <PropertyInfo ChainCode="HotelChain" BrandCode="HotelBrand" HotelCode="HotelHotel"/>
        <Room RoomID="EE12345">
                <Devices>
                        <Device Type="TV Remote" FriendlyName="000"
Class="TelevisionIntegration"/>
                        <Device Type="Television" FriendlyName="TV-1"
Class="TelevisionIntegration"/>
                        <Device Type="Thermostat" Class="HVAC"/>
                        <Device Class="DoorLock"/>
                </Devices>
        </Room>
</HTNG_HotelRoomStatusSearchRQ>
```

3.1.6 Sample Response

```
<HTNG_HotelRoomStatusSearchRS EchoToken="Echo12345" TimeStamp="2012-08-17T09:30:47Z"
Version="1.0">
        <Success/>
        <RoomInformationList>
                <RoomInformation>
                        <Room RoomID="EE12345">
                                <Devices>
                                        <Device ID="1129087f" FriendlyName="RMT01" Type="TVRemote"
Description="Acme television remote control" Class="TelevisionIntegration">
                                                <CurrentHealthStatus Value="WARNING"
Reason="LOW_BATTERY"/>
                                        </Device>
                                        <Device ID="ef29086f" FriendlyName="TV-1" Type="Television"
Description="Acme 52 inch Hospitality TV - Bedroom" Class="TelevisionIntegration">
                                                <CurrentHealthStatus Value="OPERATIONAL"/>
                                        </Device>
                                        <Device ID="ac3f196e-2a3b-23af-2bbe-2e67998e1e88"
FriendlyName="Tstat-R1" Type="Thermostat" Description="Room Thermostat" Class="HVAC">
                                                <CurrentHealthStatus Value="WARNING"
Reason="PARTIAL_FAILURE">An attempt to test the temperature sensor status failed. Device returned
error 1101.</CurrentHealthStatus>
                                        </Device>
                                        <Device ID="bf3f195a" FriendlyName="DL-12345"
Description="Door Lock" Class="DoorLock">
                                                <CurrentHealthStatus Value="ERROR"
Reason="COMMUNNICATION_ERROR">An attempt to test the door lock status failed. Lock Server
Offline. Error 502.</CurrentHealthStatus>
                                        </Device>
                                </Devices>
                        </Room>
                </RoomInformation>
        </RoomInformationList>
</HTNG_HotelRoomStatusSearchRS>
```

3.1.7 Sample Request

```
<HTNG_HotelRoomStatusSearchRQ EchoToken="Echo12345" TimeStamp="2012-08-17T09:30:47Z"
Version="1.0">
        <POS>
                <Source>
                        <RequestorID Type="10" ID_Context="ACME_PMS" ID="PMS1">
                                <CompanyName>Acme PMS Company</CompanyName>
                        </RequestorID>
                </Source>
        </POS>
        <PropertyInfo ChainCode="HotelChain" BrandCode="HotelBrand" HotelCode="HotelHotel"/>
        <Rooms>
                <Room RoomID="EE12345"/>
        </Rooms>
</HTNG_HotelRoomStatusSearchRQ>
```

3.1.8 Sample Response

```
<HTNG_HotelRoomStatusSearchRS TransactionIdentifier="Tran0001" EchoToken="Echo12345"
TimeStamp="2012-08-17T09:30:47Z" Version="1.0">
        <Success/>
        <Warnings>
                <Warning Type="101" Status="OK" ShortText="IGR Reset">IGR services restarted -
2012-08-10T014:00:47Z</Warning>
        </Warnings>
        <RoomInformationList>
                <RoomInformation>
                        <Room RoomID="EE12345">
                                <Devices>
                                        <Device ID="ef29086f" FriendlyName="TV-1" Type="Television"
Description="Acme 52 inch Hospitality TV" Class="TelevisionIntegration">
                                                <CurrentHealthStatus Value="OPERATIONAL"/>
                                        </Device>
                                        <Device ID="af39085e" FriendlyName="STB-001" Type="Set Top
Box" Description="Acme Set Top Box Model m7789K" Class="TelevisionIntegration">
                                                <CurrentHealthStatus Value="OPERATIONAL"/>
                                        </Device>
                                        <Device ID="1129087f" FriendlyName="RMT01" Type="TVRemote"
Description="Acme television remote control" Class="TelevisionIntegration">
                                                <CurrentHealthStatus Value="OPERTIONAL"/>
                                        </Device>
                                        <Device ID="ac3f196e-2a3b-23af-2bbe-2e67998e1e88"
FriendlyName="Tstat-R1" Type="Thermostat" Description="Room Thermostat" Class="HVAC">
                                                <CurrentHealthStatus Value="OPERATIONAL"/>
                                        </Device>
                                        <Device ID="bf3f195a" FriendlyName="DL-12345"
Description="Door Look" Class="DoorLock">
                                                <CurrentHealthStatus Value="OPERATIONAL"/>
                                        </Device>
                                        <Device ID="2341ae3c" FriendlyName="MB-12345"
Description="Mini Bar" Class="InRoomRefreshmentCenter">
                                                <CurrentHealthStatus Value="WARNING"
Reason="SERVICE_REQUIRED">Unable to determine item count.</CurrentHealthStatus>
                                        </Device>
                                        <Device ID="ef3278ea" FriendlyName="OS-12345"
Description="Ceiling Sensor" Class="OccupancyDetection">
                                                <CurrentHealthStatus Value="ERROR"
Reason="OFFLINE">Device is unreachable. Communications error 10060.</CurrentHealthStatus>
                                        </Device>
                                </Devices>
                                <TelephoneExtensions>
                                        <TelephoneExtention>12345</TelephoneExtention>
                                </TelephoneExtensions>
                        </Room>
                </RoomInformation>
        </RoomInformationList>
</HTNG_HotelRoomStatusSearchRS>
```

3.1.9 Sample Request

```
<HTNG_HotelRoomStatusSearchRQ EchoToken="Echo12345" TimeStamp="2012-08-17T09:30:47Z"
Version="1.0">
        <POS>
                <Source>
                        <RequestorID Type="10" ID_Context="ACME_PMS" ID="PMS1">
                                <CompanyName>Acme PMS Company</CompanyName>
                        </RequestorID>
                </Source>
        </POS>
        <PropertyInfo ChainCode="HotelChain" BrandCode="HotelBrand" HotelCode="HotelHotel"/>
        <Room>
                <Devices>
                        <Device Type="Television" FriendlyName="TV-1"
Class="TelevisionIntegration">
      </Device>
                </Devices>
        </Room>
</HTNG_HotelRoomStatusSearchRQ>
```

3.1.10 Sample Response

```
<HTNG_HotelRoomStatusSearchRS EchoToken="Echo12345" TimeStamp="2012-08-17T09:30:47Z"
Version="1.0">
        <Success/>
        <RoomInformationList>
                <RoomInformation>
                        <Room RoomID="EE12345">
                                <Devices>
                                        <Device ID="ae29086f" FriendlyName="TV-1" Type="Television"
Description="Acme 52 inch Hospitality TV - Bedroom" Class="TelevisionIntegration">
                                                <CurrentHealthStatus Value="OPERATIONAL"/>
                                        </Device>
                                </Devices>
                        </Room>
                </RoomInformation>
                <RoomInformation>
                        <Room RoomID="EE22345">
                                <Devices>
                                        <Device ID="ef29086e" FriendlyName="TV-1" Type="Television"
Description="Acme 52 inch Hospitality TV - Bedroom" Class="TelevisionIntegration">
                                                <CurrentHealthStatus Value="OPERATIONAL"/>
                                        </Device>
                                </Devices>
                        </Room>
```

```
				</RoomInformation>
				<RoomInformation>
						<Room RoomID="EE32345">
								<Devices>
										<Device ID="ff29086a" FriendlyName="TV-1" Description="Acme 
52 inch Hospitality TV - Bedroom" Class="TelevisionIntegration">
												<CurrentHealthStatus Value="OPERATIONAL"/>
										</Device>
								</Devices>
						</Room>
				</RoomInformation>
		</RoomInformationList>
</HTNG_HotelRoomStatusSearchRS>
```

3.2 Guest Room Status Event Notification

3.2.1 Overview

In order to enhance the guest experience, the Room Status Publisher monitors the status of guest room devices and raises an event to a Room Status Subscriber when device failures are detected.

Note: Please refer to the Protocol & Messaging Transport Event Notification specification for details regarding how to subscribe to and manage notifications.

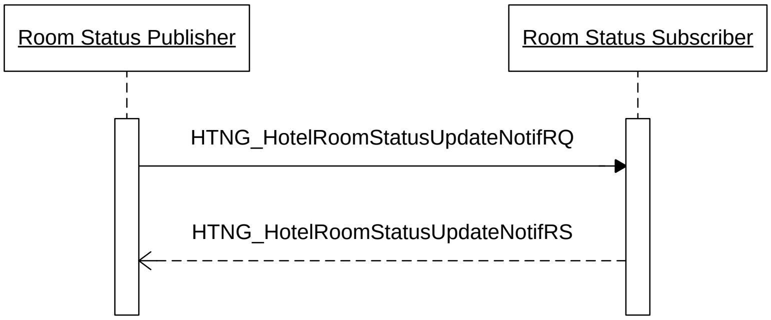
3.2.2 Roles

Role	Description	Examples
Room Status Publisher	A system that provides the ability to register for and publish events relating to the guest room devices.	• Device Gateway • Zone Controller
Room Status Subscriber	A system interested in receiving real-time updates to the health and status of guest room devices.	• PMS • Work order management • Device management console

3.2.3 Use Case

Assumptions:	• Room Status Publisher is fully functional and accessible. • The Room Status Subscriber is functional and has an active Room Status Publisher event subscription.
Pre-condition:	A Room Status Subscriber has registered interest in receiving event notifications from the Room Status Publisher.
Trigger:	Guest room device failure and corrections.
Basic Course of Events:	As guest room device failures occur or are corrected, the Room Status Publisher detects the status changes and sends an appropriate notification message to registered Room Status Subscribers.
Post-condition:	Hotel staff is informed of guest room device statuses.
Exception Paths:	• Room Status Publisher returns an error if an attempt to subscribe for notification(s) fails. • Room Status Publisher sends a notification to subscribers when the Room Status Publisher is no longer capable of sending notifications to Room Status Subscribers.
Alternative Path:	None

3.2.4 Message Flows


Room Status Publisher
Room Status Subscriber
HTNG_HotelRoomStatusUpdateNotifRQ
HTNG_HotelRoomStatusUpdateNotifRS

3.2.5 Sample Request

```
<HTNG_HotelRoomStatusUpdateNotifRQ EchoToken="Echo22222" TimeStamp="2012-08-17T09:30:47Z"
Version="1.0">
        <POS>
                <Source>
                        <RequestorID Type="10" ID_Context="ACME_PMS" ID="PMS1">
                                <CompanyName>Acme</CompanyName>
                        </RequestorID>
                </Source>
        </POS>
        <PropertyInfo ChainCode="HotelChain" BrandCode="HotelBrand" HotelCode="HotelHotel"/>
        <Room RoomID="EE12345">
                <Devices>
                        <Device ID="2341ae3c" FriendlyName="MB-12345" Description="Mini Bar"
Class="InRoomRefreshmentCenter">
                                <PriorHealthStatus Value="ERROR" Reason="UNKNOWN"/>
                                <CurrentHealthStatus Value="OPERATIONAL"
Reason="POWER_CYCLED">Manual device reset performed</CurrentHealthStatus>
                        </Device>
                        <Device ID="ef3278ea" FriendlyName="OS-12345" Description="Ceiling Sensor"
Class="OccupancyDetection">
                                <PriorHealthStatus Value="ERROR" Reason="LOW_BATTERY"/>
                                <CurrentHealthStatus Value="OPERATIONAL"
Reason="BATTERY_REPLACEMENT">Device online.</CurrentHealthStatus>
                        </Device>
                </Devices>
        </Room>
</HTNG_HotelRoomStatusUpdateNotifRQ>
```

3.2.6 Sample Response

```
<HTNG_HotelRoomStatusUpdateNotifRS TransactionIdentifier="Tran0002" EchoToken="Echo22222"
TimeStamp="2012-08-17T09:30:47Z" Version="1.0">
        <Success/>
</HTNG_HotelRoomStatusUpdateNotifRS>
```

3.3 Device Registration

3.3.1 Overview

Device registration is the process of defining a device and all relevant information to a Device Registration System. Each intelligent device must be registered with the Device Registration System. The registration system provides a means to store device information for the purpose of addressing devices and determining their capabilities and physical location.

3.3.2 Roles

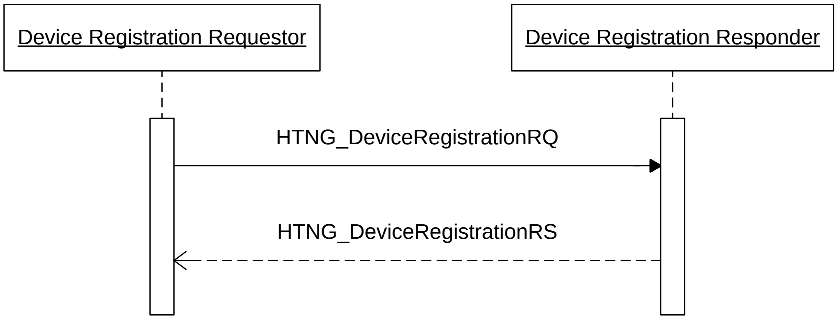
Role	Description	Examples
Device Registration Requester	A system or device capable of publishing device information to be made available to other systems or devices.	• Device Gateway • Zone Controller
Device Registration Responder	A repository of references (for the purpose of Addressing) to servers, systems, services and devices and their capabilities.	• Device Registration Server

3.3.3 Use Case

Assumptions:	• The device to be added has the capability to be addressed via the Device Registration System. • The device end point is uniquely identified. Atomic and hierarchical addresses are allowed. • The address is an URL. • If Device Capability Profiles are being used, the relevant Device Capability Profile has already been established in the Device Registration System. • Registration has been authorized.
Pre-condition:	None
Trigger:	Device is ready to be integrated into the Device Registration System.
Basic Course of Events:	1. A device installer registers the device in the Device Registration System. a. A device is added to Device Registration System or an existing device is modified. i. If Device Capability Profiles are being used, the appropriate Profile is indicated in the registration request. ii. If Device Capability Profiles are not being used, individual capabilities are included in the message

	payload. b. All pertinent information, including end point address, is added to the Device Registration System.
Post-condition:	Device is registered with the Device Registration System and may be addressed.
Exception Path:	None
Alternative Path:	None

3.3.4 Message Flows


Device Registration Requestor
Device Registration Responder
HTNG_DeviceRegistrationRQ
HTNG_DeviceRegistrationRS

3.3.5 Sample Request

```
<HTNG_DeviceRegistrationRQ EchoToken="TOKEN234" TimeStamp="2001-12-17T09:30:47.0Z" Version="1.0">
        <POS>
                <ota:Source>
                        <ota:RequestorID Type="0" ID_Context="a" ID="12345678">
                                <ota:CompanyName CompanyShortName="a"></ota:CompanyName>
                        </ota:RequestorID>
                </ota:Source>
        </POS>
        <PropertyInfo ChainCode="a" BrandCode="a" HotelCode="a" HotelCodeContext="a"/>
        <Devices>
                <Device Type="LAMP" ID="LMP1234" ModelID="GX-1" SerialNumber="123-SER-789"
MAC_Address="00-B0-D0-86-BB-F7" FriendlyName="LAMP1234" Description="IGR Lamp" Class="LIGHTING"
ModelName="GX-1a">
                        <EndpointAddress>http://htng.org/gateway/00-B0-D0-86-BB-
F7</EndpointAddress>
                        <Manufacturer Type="LAMPMANU" ID_Context="a" ID="a" Instance="a"
URL="http://www.lampco.com">
                                <ota:CompanyName CompanyShortName="LAMPCO" Code="LMPCO">LAMPCO
Inc.</ota:CompanyName>
                        </Manufacturer>
                        <DeviceLocation RoomID="1024">
                                <RoomType Wing="WEST" Building="A" Floor="10"
RoomLocationCode="0"/>
                                <TelephoneExtensions>
                                        <TelephoneExtention>1024</TelephoneExtention>
                                </TelephoneExtensions>
                        </DeviceLocation>
                        <DeviceProfile>
                                <Capabilities>
                                        <Capability Type="STATUS" Operation="getState">
                                                <OutboundParameters>
                                                        <OutboundParameter DataType="xs:boolean"
Property="State">
                                                                <KeyValueItems>
                                                                        <KeyValueItem Key="On"
Value="true"/>
                                                                        <KeyValueItem Key="Off"
Value="false"/>
                                                                </KeyValueItems>
                                                        </OutboundParameter>
```

```
                                             </OutboundParameters>
                                      </Capability>
                                      <Capability Type="COMMAND" Operation="setState">
                                             <InboundParameters>
                                                     <InboundParameter DataType="xs:boolean"
Property="State">
                                                            <KeyValueItems>
                                                                    <KeyValueItem Key="On"
Value="true"/>
                                                                    <KeyValueItem Key="Off"
Value="false"/>
                                                            </KeyValueItems>
                                                     </InboundParameter>
                                             </InboundParameters>
                                             <OutboundParameters>
                                                     <OutboundParameter DataType="xs:boolean"
Property="RESULT">
                                                            <KeyValueItems>
                                                                    <KeyValueItem Key="SUCCESS"
Value="true"/>
                                                                    <KeyValueItem Key="FAILURE"
Value="false"/>
                                                            </KeyValueItems>
                                                     </OutboundParameter>
                                             </OutboundParameters>
                                      </Capability>
                                      <Capability Type="EVENT" Operation="stateChanged">
                                             <InboundParameters>
                                                     <InboundParameter DataType="xs:boolean"
Property="State">
                                                            <KeyValueItems>
                                                                    <KeyValueItem Key="On"
Value="true"/>
                                                                    <KeyValueItem Key="Off"
Value="false"/>
                                                            </KeyValueItems>
                                                     </InboundParameter>
                                             </InboundParameters>
                                      </Capability>
                                      <Capability Type="STATUS" Operation="getDimLevel">
                                             <OutboundParameters>
                                                     <OutboundParameter DataType="xs:int"
Property="DimLevel">
                                                            <KeyValueItems>
                                                                    <KeyValueItem Key="Min"
Value="0"/>
                                                                    <KeyValueItem Key="Max"
Value="100"/>
                                                            </KeyValueItems>
                                                     </OutboundParameter>
                                             </OutboundParameters>
                                      </Capability>
                                      <Capability Type="COMMAND" Operation="setDimLevel">
                                             <InboundParameters>
                                                     <InboundParameter DataType="xs:int"
Property="DimLevel">
                                                            <KeyValueItems>
                                                                    <KeyValueItem Key="Min"
Value="0"/>
                                                                    <KeyValueItem Key="Max"
Value="100"/>
                                                            </KeyValueItems>
                                                     </InboundParameter>
                                             </InboundParameters>
                                             <OutboundParameters>
                                                     <OutboundParameter DataType="xs:boolean"
Property="RESULT">
                                                            <KeyValueItems>
                                                                    <KeyValueItem Key="SUCCESS"
Value="true"/>
                                                                    <KeyValueItem Key="FAILURE"
Value="false"/>
                                                            </KeyValueItems>
                                                     </OutboundParameter>
                                             </OutboundParameters>
                                      </Capability>
                                      <Capability Type="EVENT" Operation="dimLevelChanged">
                                             <InboundParameters>
                                                     <InboundParameter DataType="xs:int"
Property="DimLevel">
```

```
                                                            <KeyValueItems>
                                                                   <KeyValueItem Key="Min"
Value="0"/>
                                                                   <KeyValueItem Key="Max"
Value="100"/>
                                                            </KeyValueItems>
                                                     </InboundParameter>
                                             </InboundParameters>
                                      </Capability>
                                      <Capability Type="STATUS" Operation="getScene">
                                             <OutboundParameters>
                                                     <OutboundParameter DataType="xs:int"
Property="Scene">
                                                            <KeyValueItems>
                                                                   <KeyValueItem Key="Min"
Value="1"/>
                                                                   <KeyValueItem Key="Max"
Value="32"/>
                                                            </KeyValueItems>
                                                     </OutboundParameter>
                                             </OutboundParameters>
                                      </Capability>
                                      <Capability Type="COMMAND" Operation="setScene">
                                             <InboundParameters>
                                                     <InboundParameter DataType="xs:int"
Property="Scene">
                                                            <KeyValueItems>
                                                                   <KeyValueItem Key="Min"
Value="1"/>
                                                                   <KeyValueItem Key="Max"
Value="32"/>
                                                            </KeyValueItems>
                                                     </InboundParameter>
                                             </InboundParameters>
                                             <OutboundParameters>
                                                     <OutboundParameter DataType="xs:boolean"
Property="RESULT">
                                                            <KeyValueItems>
                                                                   <KeyValueItem Key="SUCCESS"
Value="true"/>
                                                                   <KeyValueItem Key="FAILURE"
Value="false"/>
                                                            </KeyValueItems>
                                                     </OutboundParameter>
                                             </OutboundParameters>
                                      </Capability>
                                      <Capability Type="EVENT" Operation="sceneChanged">
                                                     <InboundParameters>
                                                            <InboundParameter DataType="xs:int"
Property="Scene">
                                                                   <KeyValueItems>
                                                                          <KeyValueItem
Key="Min" Value="1"/>
                                                                          <KeyValueItem
Key="Max" Value="32"/>
                                                                   </KeyValueItems>
                                                            </InboundParameter>
                                                     </InboundParameters>
                                      </Capability>
                                      <Capability Type="STATUS" Operation="getGreenMode">
                                             <OutboundParameters>
                                                     <OutboundParameter DataType="xs:boolean"
Property="RESULT">
                                                            <KeyValueItems>
                                                                   <KeyValueItem Key="Active"
Value="true"/>
                                                                   <KeyValueItem Key="Inactive"
Value="false"/>
                                                            </KeyValueItems>
                                                     </OutboundParameter>
                                             </OutboundParameters>
                                      </Capability>
                                      <Capability Type="COMMAND" Operation="setGreenMode">
                                             <InboundParameters>
                                                     <InboundParameter DataType="xs:boolean"
Property="GreenMode">
                                                            <KeyValueItems>
                                                                   <KeyValueItem Key="Active"
Value="true"/>
```

```
                                                                        <KeyValueItem Key="Inactive"
Value="false"/>
                                                                </KeyValueItems>
                                                        </InboundParameter>
                                                </InboundParameters>
                                                <OutboundParameters>
                                                        <OutboundParameter DataType="xs:boolean"
Property="RESULT">
                                                                <KeyValueItems>
                                                                        <KeyValueItem Key="SUCCESS"
Value="true"/>
                                                                        <KeyValueItem Key="FAILURE"
Value="false"/>
                                                                </KeyValueItems>
                                                        </OutboundParameter>
                                                </OutboundParameters>
                                        </Capability>
                                        <Capability Type="EVENT" Operation="greenModeChanged">
                                                <InboundParameters>
                                                        <InboundParameter DataType="xs:boolean"
Property="GreenMode">
                                                                <KeyValueItems>
                                                                        <KeyValueItem Key="Active"
Value="true"/>
                                                                        <KeyValueItem Key="Inactive"
Value="false"/>
                                                                </KeyValueItems>
                                                        </InboundParameter>
                                                </InboundParameters>
                                        </Capability>
                                        <Capability Type="STATUS" Operation="getBulbOut">
                                                <OutboundParameters>
                                                        <OutboundParameter DataType="xs:boolean"
Property="BulbOut">
                                                                <KeyValueItems>
                                                                        <KeyValueItem Key="Yes"
Value="true"/>
                                                                        <KeyValueItem Key="No"
Value="false"/>
                                                                </KeyValueItems>
                                                        </OutboundParameter>
                                                </OutboundParameters>
                                        </Capability>
                                        <Capability Type="EVENT" Operation="bulbOutChanged">
                                                <InboundParameters>
                                                        <InboundParameter DataType="xs:boolean"
Property="BulbOut">
                                                                <KeyValueItems>
                                                                        <KeyValueItem Key="Yes"
Value="true"/>
                                                                        <KeyValueItem Key="No"
Value="false"/>
                                                                </KeyValueItems>
                                                        </InboundParameter>
                                                </InboundParameters>
                                        </Capability>
                                        <Capability Type="STATUS" Operation="getUsage">
                                                <OutboundParameters>
                                                        <OutboundParameter DataType="xs:int"
Property="Usage"/>                      </OutboundParameters>
                                        </Capability>
                                </Capabilities>
                        </DeviceProfile>
                </Device>
        </Devices>
</HTNG_DeviceRegistrationRQ>
```

3.3.6 Sample Response

```
<HTNG_DeviceRegistrationRS EchoToken="TOKEN234" TimeStamp="2001-12-17T09:30:47.0Z" Version="1.0">
        <Success />
</HTNG_DeviceRegistrationRS>
```

3.4 Device Capability Profile Management

3.4.1 Overview

Device Capability Profiles are used to establish a standard definition for devices that share the same set of capabilities. This reduces the complexity of storing the capabilities of individual devices, and allows a more consistent representation of the devices across the enterprise.

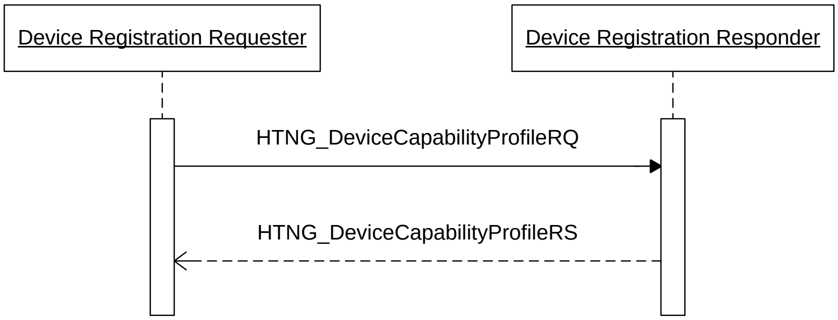
3.4.2 Roles

Role	Description	Examples
Device Registration Requester	A system or device capable of publishing device information to be made available to other systems or devices.	• Device Gateway • Zone Controller
Device Registration Responder	A repository of references (for the purpose of Addressing) to servers, systems, services and devices and their capabilities.	• Device Registration Server

3.4.3 Use Case

Assumptions:	• The Device Registration Server is operational. • Profile registration has been authorized.
Pre-condition:	None
Trigger:	A device capability profile is ready to be added or modified in the Device Registration Server.
Basic Course of Events:	1. A device installer prepares the Device Capability Profile. 2. A device installer sends the HTNG_DeviceCapabilityProfileRQ to the Device Registration Server.
Post-condition:	The Device Capability Profile is registered with the Device Registration Server and may be used.
Exception Path:	None
Alternative Path:	None

3.4.4 Message Flows


Device Registration Requester
Device Registration Responder
HTNG_DeviceCapabilityProfileRQ
HTNG_DeviceCapabilityProfileRS

3.4.5 Sample Request

```
<HTNG_DeviceCapabilityProfileRQ EchoToken="TOKEN2345" TimeStamp="2001-12-17T09:30:47.0Z"
Version="1.0" Action=”Add”>
        <POS>
                <ota:Source>
                        <ota:RequestorID Type="0" ID_Context="a" ID="12345678">
                                <ota:CompanyName CompanyShortName="a"></ota:CompanyName>
                        </ota:RequestorID>
                </ota:Source>
        </POS>
        <PropertyInfo ChainCode="HotelChain" HotelName="HotelName" BrandCode="HotelBrandCode"
HotelCode="HotelCode" HotelCodeContext="Hotel"/>
        <DeviceProfile>
                <Capabilities>
                        <Capability Type="STATUS" Operation="getState">
                                <OutboundParameters>
                                        <OutboundParameter DataType="xs:boolean" Property="State">
                                                <KeyValueItems>
                                                        <KeyValueItem Key="On" Value="true"/>
                                                        <KeyValueItem Key="Off" Value="false"/>
                                                </KeyValueItems>
                                        </OutboundParameter>
                                </OutboundParameters>
                        </Capability>
                        <Capability Type="COMMAND" Operation="setState">
                                <InboundParameters>
                                        <InboundParameter DataType="xs:boolean" Property="State">
                                                <KeyValueItems>
                                                        <KeyValueItem Key="On" Value="true"/>
                                                        <KeyValueItem Key="Off" Value="false"/>
                                                </KeyValueItems>
                                        </InboundParameter>
                                </InboundParameters>
                                <OutboundParameters>
                                        <OutboundParameter DataType="xs:boolean" Property="RESULT">
                                                <KeyValueItems>
                                                        <KeyValueItem Key="SUCCESS" Value="true"/>
                                                        <KeyValueItem Key="FAILURE" Value="false"/>
                                                </KeyValueItems>
                                        </OutboundParameter>
                                </OutboundParameters>
                        </Capability>
                        <Capability Type="EVENT" Operation="stateChanged">
                                <InboundParameters>
                                        <InboundParameter DataType="xs:boolean" Property="State">
                                                <KeyValueItems>
                                                        <KeyValueItem Key="On" Value="true"/>
                                                        <KeyValueItem Key="Off" Value="false"/>
                                                </KeyValueItems>
                                        </InboundParameter>
                                </InboundParameters>
                        </Capability>
                </Capabilities>
        </DeviceProfile>
</HTNG_DeviceCapabilityProfileRQ>
```

3.4.6 Sample Response

```
<HTNG_DeviceCapabilityProfileRS EchoToken="TOKEN234" TimeStamp="2001-12-17T09:30:47.0Z"
Version="1.0">
        <Success />
 </HTNG_DeviceCapabilityProfileRS>
```

3.5 Device Discovery

3.5.1 Overview

Hotel Information Systems may need to gather device information managed by the Device Registration System.

In order to access the functionality of an Intelligent Device, the Hotel Information System must obtain properties of a device from the Device Registration System. This may include the end-point address of the device, its capabilities and physical location.

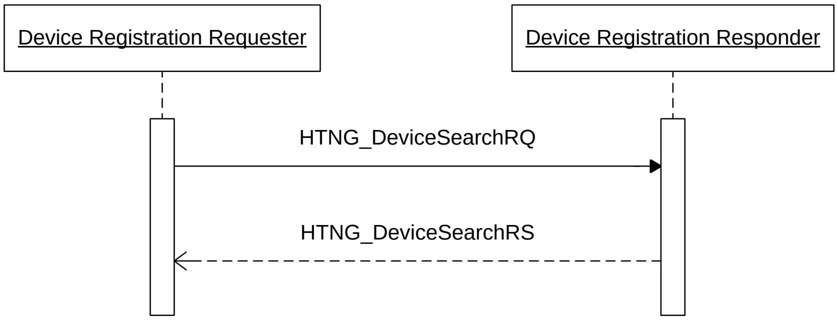
3.5.2 Roles

Role	Description	Examples
Device Discovery Requester	Any system interested in locating devices and their capabilities.	• Room Control System • Property Management System
Device Discovery Responder	A repository of references (for the purpose of Addressing) to servers, systems, services and devices and their capabilities.	• Registration Server

3.5.3 Use Case

Assumptions:	• The Device Discovery Responder is available. • The Device Discovery Requester knows the scope of search criteria for query.
Pre-condition:	None
Trigger:	The Device Discovery Requester needs to discover devices.
Basic Course of Events:	1) The Device Discovery Requester uses the Device Discovery Responder to query device information. a. The query can be filtered on device properties. 2) The Device Discovery Responder responds with the device(s) that satisfies the query.
Post-condition:	The Device Discovery Requester has the requested device information.
Exception Path:	None
Alternative Path:	None

3.5.4 Message Flows



3.5.5 Sample Request

```
<HTNG_DeviceSearchRQ EchoToken="SEARCH123" TimeStamp="2001-12-17T09:30:47.0Z" Version="1.0"
ReturnListIndicator="false">
        <Devices>
                <!-- SEARCH BY DEVICE & PROFILE TYPE -->
                <Device Type="LOCK" >
                        <DeviceLocation RoomID="1024"/>
                        <DeviceProfile ID="Lock"/>
                </Device>
                <!-- SEARCH BY DEVICE TYPE AND CAPABILITIES -->
                <Device Type="LAMP" >
                        <DeviceLocation RoomID="1024"/>
                        <DeviceProfile>
                                <Capabilities>
                                        <Capability Type="STATUS" Operation="getState"/>
                                </Capabilities>
                        </DeviceProfile>
                </Device>
        </Devices>
</HTNG_DeviceSearchRQ>
```

3.5.6 Sample Response

```
<HTNG_DeviceSearchRS EchoToken="TOKEN234" TimeStamp="2001-12-17T09:30:47.0Z" Version="1.0">
        <Success />
        <Devices>
                <Device Type="Lock" ID="1129087" ModelID="String" FriendlyName="LockCo Lock"
                        Description="IGR Lock" Class="Door Lock" ModelName="111"
endPoint="gateway/823-bc-igr111"
                        <DeviceProfile ID="Lock"/>
                </Device>
                <Device Type="Lamp" ID="1129087f" ModelID="String" FriendlyName="LampCo Lamp"
Description="IGR Lamp" Class="Lighting" ModelName="101" endPoint="gateway/823-bc-igr101"/>
                        <DeviceProfile>
                                <Capabilities>
                                        <Capability Type="STATUS" Operation="getState">
                                                <OutboundParameters>
                                                        <OutboundParameter DataType="xs:boolean"
Property="State">
                                                                <KeyValueItems>
                                                                        <KeyValueItem Key="On"
Value="true"/>
                                                                        <KeyValueItem Key="Off"
Value="false"/>
                                                                        </KeyValueItems>
                                                        </OutboundParameter>
                                                </OutboundParameters>
                                        </Capability>
                                </Ca[abilities>
                        </DeviceProfile>
                <Device/>
        </Devices>
 </HTNG_DeviceSearchRS>
```

4 Messages

4.1 Guest Room Status Check

4.1.1 Data Element Table – Request

Element | @Attribute	Num	Description/Contents
HTNG_HotelRoomStatusSearchRQ	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_HotelRoomStatusSearchRQ / POS / Source	1	This holds details regarding the requestor. It may be repeated to also accommodate the delivery systems.
HTNG_HotelRoomStatusSearchRQ / POS / Source / RequestorID	1	An identifier of the entity making the request (e.g., ATA/IATA/ID number, Electronic Reservation Service Provider (ERSP), Association of British Travel Agents (ABTA)).
@Type	0..1	A reference to the type of object defined by the UniqueID element. Refer to OpenTravel Code List Unique ID Type (UIT).
@ID_Context	0..1	Used to identify the source of the identifier (e.g., IATA, ABTA).
@ID	1	A unique identifying value assigned by the creating system. The ID attribute may be used to reference a primary-key value within a database or in a particular implementation.
HTNG_HotelRoomStatusSearchRQ / PropertyInfo	1	Identifies a specific hotel by using the Chain Code, the Brand Code, and the Hotel Code. The codes used are agreed upon by trading partners.

Element | @Attribute	Num	Description/Contents
@HotelCode	1	The code that uniquely identifies a single hotel property. The hotel code is decided between vendors.
@BrandCode	0..1	A code that identifies the brand or flag of a hotel, often used for independently-owned or franchised properties who are known by a specific brand.
@ChainCode	0..1	The code that identifies a hotel chain or management group. The hotel chain code is decided between vendors. This attribute is optional if the hotel is an independent property that can be identified by the HotelCode attribute.
HTNG_HotelRoomStatusSearchRQ / Room	1	Used to convey information about a single room or a suite comprised of room components. The populated values on this entity are used as the query parameters.
@RoomID	0..1	The room number.
HTNG_HotelRoomStatusSearchRQ / Room / RoomType	0..1	A container element which holds information describing the room.
@Building	0..1	The building in which the room is located.
@Floor	0..1	The floor on which the room is located.
@Wing	0..1	The wing on which the room is located.
@RoomLocationCode	0..1	Indicates the location of the room within the hotel structure. Typical values would be "Near Exit", "Close to elevator", "Low Floor" or "High Floor". Refer to OpenTravel Code List Room Location Type (RLT).
HTNG_HotelRoomStatusSearchRQ / Room / Devices	0..1	A collection of Devices.
HTNG_HotelRoomStatusSearchRQ / Room / Devices / Device	1..n	Container element which holds information describing the device.
@ID	0..1	An identifier that is unique within the network.
@FriendlyName	0..1	A name used to identify a device in a room.
@Class	0..1	General classification of the device, ie HVAC, TelevisionIntegration, etc.

Element | @Attribute	Num	Description/Contents
@Type	0..1	Specific classification of the device, ie Thermostat, SetTopBox, etc.
@Description	0..1	Text Description of the device.
HTNG_HotelRoomStatusSearchRQ / Room / TelephoneExtensions	0..1	A collection of telephone extensions.
HTNG_HotelRoomStatusSearchRQ / Room / TelephoneExtensions / TelephoneExtension	1..n	A telephone extension that is part of a single room or a suite of component rooms.

4.1.2 Data Element Table – Response

Element | @Attribute	Num	Description/Contents
HTNG_HotelRoomStatusSearchRS	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_HotelRoomStatusSearchRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_HotelRoomStatusSearchRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
HTNG_HotelRoomStatusSearchRS / Warnings / Warning	1..n	Used when a message has been successfully processed to report any warnings or business errors that occurred.

Element | @Attribute	Num	Description/Contents
@Type	1	The Warning element MUST contain the Type attribute that uses a recommended set of values to indicate the warning type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).
HTNG_HotelRoomStatusSearchRQ / Errors	0..1	A collection of errors that occurred during the processing of a message.
HTNG_HotelRoomStatusSearchRQ / Errors / Error	1..n	An error that occurred during the processing of a message.
@Type	1	The Error element MUST contain the Type attribute that uses a recommended set of values to indicate the error type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.

Element | @Attribute	Num	Description/Contents
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).
HTNG_HotelRoomStatusSearchRS / RoomInformationList	0..1	The result set generated by the query sent in the request.
HTNG_HotelRoomStatusSearchRS / RoomInformationList / RoomInformation	1..n	A container element used to hold room and reservation information.
HTNG_HotelRoomStatusSearchRS / RoomInformationList / RoomInformation / Room	1	Used to convey information about a single room or a suite comprised of room components.
@RoomID	1	The room number.
HTNG_HotelRoomStatusSearchRS / RoomInformationList / RoomInformation / Room / Devices	0..1	A collection of Devices.
HTNG_HotelRoomStatusSearchRS / RoomInformationList / RoomInformation / Room / Devices / Device	1..n	Container element which holds information describing the device.
@ID	1	An identifier that is unique within the network.
@FriendlyName	0..1	A name used to identify a device in a room.
@Class	1	General classification of the device, i.e. HVAC, TelevisionIntegration, etc.
@Type	0..1	Specific classification of the device, i.e. Thermostat, SetTopBox, etc.
@Description	1	Text Description of the device.
HTNG_HotelRoomStatusSearchRS / RoomInformationList / RoomInformation / Room / Devices / Device / CurrentHealthStatus	1	Container element that contains the current operational status of the device.

Element | @Attribute	Num	Description/Contents
@Value	1	The enumerated operational status of the device.
@Reason	0..1	Text description of the reason for the device operational status.

4.2 Guest Room Status Event Notification

4.2.1 Data Element Table – Request

Element | @Attribute	Num	Description/Contents
HTNG_HotelRoomStatusUpdateNotifRQ	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_HotelRoomStatusUpdateNotifRQ / POS / Source	1	This holds details regarding the requestor. It may be repeated to also accommodate the delivery systems.
HTNG_HotelRoomStatusUpdateNotifRQ / POS / Source / RequestorID	1	An identifier of the entity making the request (e.g., ATA/IATA/ID number, Electronic Reservation Service Provider (ERSP), Association of British Travel Agents (ABTA)).
@Type	0..1	A reference to the type of object defined by the UniqueID element. Refer to OpenTravel Code List Unique ID Type (UIT).
@ID_Context	0..1	Used to identify the source of the identifier (e.g., IATA, ABTA).
@ID	1	A unique identifying value assigned by the creating system. The ID attribute may be used to reference a primary-key value within a database or in a particular implementation.

Element | @Attribute	Num	Description/Contents
HotelRoomStatusUpdateNotifRQ / PropertyInfo	1	Identifies a specific hotel by using the Chain Code, the Brand Code, and the Hotel Code. The codes used are agreed upon by trading partners.
@HotelCode	1	The code that uniquely identifies a single hotel property. The hotel code is decided between vendors.
@BrandCode	0..1	A code that identifies the brand or flag of a hotel, often used for independently-owned or franchised properties who are known by a specific brand.
@ChainCode	0..1	The code that identifies a hotel chain or management group. The hotel chain code is decided between vendors. This attribute is optional if the hotel is an independent property that can be identified by the HotelCode attribute.
HotelRoomStatusUpdateNotifRQ / Room	1	Used to convey information about a single room or a suite comprised of room components.
@RoomID	1	The room number.
HotelRoomStatusUpdateNotifRQ / Room / Devices	1	A collection of Devices.
HotelRoomStatusUpdateNotifRQ / Room / Devices / Device	1..n	Container element which holds information describing the device.
@ID	0..1	An identifier that is unique within the network.
@FriendlyName	0..1	A name used to identify a device in a room.
@Description	1	Text Description of the device.
@Class	1	General classification of the device, ie HVAC, TelevisionIntegration, etc.
@Type	0..1	Specific classification of the device, ie Thermostat, SetTopBox, etc.
HotelRoomStatusUpdateNotifRQ / Room / Devices / Device / PriorHealthStatus	0..1	Container element that contains the previous operational status for the device.
@Value	1	The enumerated operational status of the device.
@Reason	0..1	Text description of the reason for the device operational status.

Element | @Attribute	Num	Description/Contents
HotelRoomStatusUpdateNotifRQ / Room / Devices / Device / CurrentHealthStatus	1	Container element that contains the current operational status of the device.
@Value	1	The enumerated operational status of the device.
@Reason	0..1	Text description of the reason for the device operational status.

4.2.2 Data Element Table – Response

Element | @Attribute	Num	Description/Contents
HTNG_HotelRoomStatusUpdateNotifRS	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_HotelRoomStatusUpdateNotifRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_HotelRoomStatusUpdateNotifRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
HTNG_HotelRoomStatusUpdateNotifRS / Warnings / Warning	1..n	Used when a message has been successfully processed to report any warnings or business errors that occurred.

Element | @Attribute	Num	Description/Contents
@Type	1	The Warning element MUST contain the Type attribute that uses a recommended set of values to indicate the warning type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).
HTNG_HotelRoomStatusUpdateNotifRS / Errors	0..1	A collection of errors that occurred during the processing of a message.
HTNG_HotelRoomStatusUpdateNotifRS / Errors / Error	1..n	An error that occurred during the processing of a message.
@Type	1	The Error element MUST contain the Type attribute that uses a recommended set of values to indicate the error type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.

Element | @Attribute	Num	Description/Contents
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).

4.3 Device Registration

4.3.1 Data Element Table – Request

Element | @Attribute	Num	Description/Contents
HTNG_DeviceRegistrationRQ	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_DeviceRegistrationRQ / POS	1	POS provides a mechanisim to indicate the source of the message.
HTNG_DeviceRegistrationRQ / POS / Source	1	This holds details regarding the requestor. It may be repeated to also accommodate the delivery systems.
HTNG_DeviceRegistrationRQ / POS / Source / RequestorID	1	An identifier of the entity making the request (e.g., ATA/IATA/ID number, Electronic Reservation Service Provider (ERSP), Association of British Travel Agents (ABTA)).
@Type	0..1	A reference to the type of object defined by the UniqueID element. Refer to OpenTravel Code List Unique ID Type (UIT).

Element | @Attribute	Num	Description/Contents
@ID	1	A unique identifying value assigned by the creating system. The ID attribute may be used to reference a primary-key value within a database or in a particular implementation.
@ID_Context	0..1	Used to identify the source of the identifier (e.g., IATA, ABTA).
HTNG_DeviceRegistrationRQ / POS / Source / RequestorID / CompanyName	0..1	Identifies the company that is associated with the UniqueID.
@CompanyShortName	1	Used to provide the company common name.
HTNG_DeviceRegistrationRQ / PropertyInfo	1	Identifies a specific hotel by using the Chain Code, the Brand Code, and the Hotel Code. The codes used are agreed upon by trading partners.
@HotelCode	1	The code that uniquely identifies a single hotel property. The hotel code is decided between vendors.
@ChainCode	0..1	The code that identifies a hotel chain or management group. The hotel chain code is decided between vendors. This attribute is optional if the hotel is an independent property that can be identified by the HotelCode attribute.
@BrandCode	0..1	A code that identifies the brand or flag of a hotel, often used for independently-owned or franchised properties who are known by a specific brand.
@HotelCodeContext	0..1	A text field used to communicate the context (or source of - ex Sabre, Galileo, Worldspan, Amadeus) the HotelReferenceGroup codes.
HTNG_DeviceRegistrationRQ / Devices	1	A collection of Devices.
HTNG_DeviceRegistrationRQ / Devices / Device	1..n	Container element which holds information describing the device.
@Type	1	Specific classification of the device, ie Thermostat, SetTopBox, etc.
@ID	1	An identifier that is unique within the network.
@ModelID	0..1	The model of the device as provided by the manufacturer.

Element | @Attribute	Num	Description/Contents
@SerialNumber	0..1	The serial number of the device as provided by the manufacturer.
@MAC_Address	0..1	A unique identifier assigned to network devices.
@FriendlyName	1	A name used to identify a device in a room.
@Description	0..1	Text Description of the device.
@Class	1	General classification of the device, ie HVAC, TelevisionIntegration, etc.
@ModelName	0..1	The model of the device as provided by the manufacturer.
HTNG_DeviceRegistrationRQ / Devices / Device / EndpointAddress	1	Unique endpoint address of this device.
HTNG_DeviceRegistrationRQ / Devices / Device / Manufacturer	0..1	A container element containing information about the manufacturer of the device.
@Type	1	A reference to the type of object defined by the UniqueID element. Refer to OpenTravel Code List Unique ID Type (UIT).
@ID	1	A unique identifier assigned to a given manufacturer.
@ID_Context	0..1	Used to identify the source of the identifier (e.g., IATA, ABTA).
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceLocation	1	A container element containing information about where a device is located.
@RoomID	0..1*	Room Number where this particular device is located. (*Either RoomID of TelephoneExtensions MUST be used.)
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceLocation / RoomType	0..1	A container element used to describe certain facets of where a device is located.
@Building	0..1	The building in which the room is located.
@Floor	0..1	The floor on which the room is located.
@Wing	0..1	The wing on which the room is located.

Element | @Attribute	Num	Description/Contents
@RoomLocationCode	0..1	Indicates the location of the room within the hotel structure. Typical values would be "Near Exit", "Close to elevator", "Low Floor" or "High Floor". Refer to OpenTravel Code List Room Location Type (RLT).
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceLocation / TelephoneExtensions	0..1*	A collection of telephone extensions. (*Either RoomID of TelephoneExtensions MUST be used.)
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceLocation / TelephoneExtensions / TelephoneExtension	1..n	A telephone extension that is part of a single room or a suite of component rooms.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile	1	An element used to describe the set of available capabilities.
@ID	0..1*	A reference to an established Device Profile (See Device Capability Profile Management use case). If this value is populated, do not include the Capabilities collection.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile / Capabilities	0..1*	A collection of Capability elements used to convey device behaviors and interactions. This collection should only be present if not using Device Capability Profiles.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability	1..n	An individual behavior.
@Type	1	One of 4 values (Per the DMS specification): • STATUS – Obtains the status of a given property of a device. • COMMAND – Instruction to the device to perform an action. • EVENT – Callback to indicate a device property has changed. • VENDOR_SPECIFIC – Vendor specific implementation.
@Operation	1	The method/function that can/will called.

Element | @Attribute	Num	Description/Contents
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability / OutboundParameters	0..1	A collection of elements used to convey the parameters passed to the function call.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability / OutboundParameters / OutboundParameter	1..n	A single parameter.
@DataType	1	The XML schema data type of the parameter.
@Property	1	The name of the property.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability / OutboundParameters / OutboundProperty / KeyValueItems	0..1	A collection of enumerable items used to constrain the allowable values for the type referenced by the @Property.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability / OutboundParameters / OutboundProperty / KeyValueItems / KeyValueItem	1..n	An individual item in the enumeration.
@Key	1	The abstract display name corresponding to the Value.
@Value	1	Corresponds to the value of the OutboundParameter@DataType.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability / InboundParameters	0..1	A collection of elements used to convey the parameters returned by the function call.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability / InboundParameters / InboundParameter	1..n	A single parameter.

Element | @Attribute	**Num**	**Description/Contents**
@DataType	1	The XML schema data type of the parameter.
@Property	1	The name of the property.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability / InboundParameters / InboundProperty / KeyValueItems	0..1	A collection of enumerable items used to constrain the allowable values for the type referenced by the @Property.
HTNG_DeviceRegistrationRQ / Devices / Device / DeviceProfile Capabilities / Capability / InboundParameters / InboundProperty / KeyValueItems / KeyValueItem	1..n	An individual item in the enumeration.
@Key	1	The abstract display name corresponding to the Value.
@Value	1	Corresponds to the value of the OutboundParameter@DataType.
HTNG_DeviceRegistrationRQ / TPA_Extensions	0..1	This is used to exchange custom/implementation-specific values that are agreed upon between trading partners.

4.3.2 Data Element Table – Response

Element | @Attribute	**Num**	**Description/Contents**
HTNG_DeviceRegistrationRS	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).

Element | @Attribute	Num	Description/Contents
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_DeviceRegistrationRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_DeviceRegistrationRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
HTNG_DeviceRegistrationRS / Warnings / Warning	1..n	Used when a message has been successfully processed to report any warnings or business errors that occurred.
@Type	1	The Warning element MUST contain the Type attribute that uses a recommended set of values to indicate the warning type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).
HTNG_DeviceRegistrationRS / Errors	0..1	A collection of errors that occurred during the processing of a message.
HTNG_DeviceRegistrationRS / Errors / Error	1..n	An error that occurred during the processing of a message.

Element | @Attribute	Num	Description/Contents
@Type	1	The Error element MUST contain the Type attribute that uses a recommended set of values to indicate the error type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).

4.4 Device Capability Profile Management

4.4.1 Data Element Table – Request

Element | @Attribute	Num	Description/Contents
HTNG_DeviceCapabilityProfileRQ	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.

Element | @Attribute	Num	Description/Contents
HTNG_DeviceCapabilityProfileRQ / POS	1	POS provides a mechanisim to indicate the source of the message.
HTNG_DeviceCapabilityProfileRQ / POS / Source	1	This holds details regarding the requestor. It may be repeated to also accommodate the delivery systems.
HTNG_DeviceCapabilityProfileRQ / POS / Source / RequestorID	1	An identifier of the entity making the request (e.g., ATA/IATA/ID number, Electronic Reservation Service Provider (ERSP), Association of British Travel Agents (ABTA)).
@Type	0..1	A reference to the type of object defined by the UniqueID element. Refer to OpenTravel Code List Unique ID Type (UIT).
@ID	1	A unique identifying value assigned by the creating system. The ID attribute may be used to reference a primary-key value within a database or in a particular implementation.
@ID_Context	0..1	Used to identify the source of the identifier (e.g., IATA, ABTA).
HTNG_DeviceCapabilityProfileRQ / POS / Source / RequestorID / CompanyName	0..1	Identifies the company that is associated with the UniqueID.
@CompanyShortName	1	Used to provide the company common name.
HTNG_DeviceCapabilityProfileRQ / PropertyInfo	1	Identifies a specific hotel by using the Chain Code, the Brand Code, and the Hotel Code. The codes used are agreed upon by trading partners.
@HotelCode	1	The code that uniquely identifies a single hotel property. The hotel code is decided between vendors.
@ChainCode	0..1	A code that identifies the brand or flag of a hotel, often used for independently-owned or franchised properties who are known by a specific brand.
@BrandCode	0..1	The code that identifies a hotel chain or management group. The hotel chain code is decided between vendors. This attribute is optional if the hotel is an independent property that can be identified by the HotelCode attribute.

Element | @Attribute	Num	Description/Contents
@HotelCodeContext	0..1	A text field used to communicate the context (or source of – ex Sabre, Galileo, Worldspan, Amadeus) the HotelReferenceGroup codes.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile	1	An element used to describe the set of available capabilities.
@ID	0..1*	A reference to an established Device Profile (See Device Capability Profile Management use case). If this value is populated, do not include the Capabilities collection.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities	0..1*	A collection of Capability elements used to convey device behaviors and interactions. This collection should only be present if not using Device Capability Profiles.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability	1..n	An individual behavior.
@Type	1	One of 4 values (Per the DMS specification): • STATUS – Obtains the status of a given property of a device. • COMMAND – Instruction to the device to perform an action. • EVENT – Callback to indicate a device property has changed. • VENDOR_SPECIFIC – Vendor specific implementation.
@Operation	1	The method/function that can/will called.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability / OutboundParameters	0..1	A collection of elements used to convey the parameters passed to the function call.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability / OutboundParameters / OutboundParameter	1..n	A single parameter.
@DataType	1	The XML schema data type of the parameter.
@Property	1	The name of the property.

Element | @Attribute	Num	Description/Contents
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability / OutboundParameters / OutboundProperty / KeyValueItems	0..1	A collection of enumerable items used to constrain the allowable values for the type referenced by the @Property.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability / OutboundParameters / OutboundProperty / KeyValueItems / KeyValueItem	1..n	An individual item in the enumeration.
@Key	1	The abstract display name corresponding to the Value.
@Value	1	Corresponds to the value of the OutboundParameter@DataType.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability / InboundParameters	0..1	A collection of elements used to convey the parameters returned by the function call.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability / InboundParameters / InboundParameter	1..n	A single parameter.
@DataType	1	The XML schema data type of the parameter.
@Property	1	The name of the property.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability / InboundParameters / InboundProperty / KeyValueItems	0..1	A collection of enumerable items used to constrain the allowable values for the type referenced by the @Property.
HTNG_DeviceCapabilityProfileRQ / DeviceProfile / Capabilities / Capability / InboundParameters / InboundProperty / KeyValueItems / KeyValueItem	1..n	An individual item in the enumeration.
@Key	1	The abstract display name corresponding to the Value.

Element | @Attribute	Num	Description/Contents
@Value	1	Corresponds to the value of the OutboundParameter@DataType.
HTNG_DeviceCapabilityProfileRQ / TPA_Extensions	0..1	This is used to exchange custom/implementation-specific values that are agreed upon between trading partners.

4.4.2 Data Element Table – Response

Element | @Attribute	Num	Description/Contents
HTNG_DeviceCapabilityProfileRS	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_DeviceCapabilityProfileRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_DeviceCapabilityProfileRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
HTNG_DeviceCapabilityProfileRS / Warnings / Warning	1..n	Used when a message has been successfully processed to report any warnings or business errors that occurred.
@Type	1	The Warning element MUST contain the Type attribute that uses a recommended set of values to indicate the warning type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).

Element | @Attribute	Num	Description/Contents
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).
HTNG_DeviceCapabilityProfileRS / Errors	0..1	A collection of errors that occurred during the processing of a message.
HTNG_DeviceCapabilityProfileRS / Errors / Error	1..n	An error that occurred during the processing of a message.
@Type	1	The Error element MUST contain the Type attribute that uses a recommended set of values to indicate the error type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).

4.5 Device Endpoint Discovery

4.5.1 Data Element Table – Request

Element | @Attribute	Num	Description/Contents
HTNG_DeviceSearchRQ	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_DeviceSearchRQ / POS	1	POS provides a mechanisim to indicate the source of the message.
HTNG_DeviceSearchRQ / POS / Source	1	This holds details regarding the requestor. It may be repeated to also accommodate the delivery systems.
HTNG_DeviceSearchRQ / POS / Source / RequestorID	1	An identifier of the entity making the request (e.g., ATA/IATA/ID number, Electronic Reservation Service Provider (ERSP), Association of British Travel Agents (ABTA)).
@Type	0..1	A reference to the type of object defined by the UniqueID element. Refer to OpenTravel Code List Unique ID Type (UIT).
@ID	1	A unique identifying value assigned by the creating system. The ID attribute may be used to reference a primary-key value within a database or in a particular implementation.
@ID_Context	0..1	Used to identify the source of the identifier (e.g., IATA, ABTA).
HTNG_DeviceSearchRQ / POS / Source / RequestorID / CompanyName	0..1	Identifies the company that is associated with the UniqueID.
@CompanyShortName	1	Used to provide the company common name.

Element | @Attribute	Num	Description/Contents
HTNG_DeviceSearchRQ / PropertyInfo	1	Identifies a specific hotel by using the Chain Code, the Brand Code, and the Hotel Code. The codes used are agreed upon by trading partners.
@HotelCode	1	The code that uniquely identifies a single hotel property. The hotel code is decided between vendors.
@ChainCode	0..1	A code that identifies the brand or flag of a hotel, often used for independently-owned or franchised properties who are known by a specific brand.
@BrandCode	0..1	The code that identifies a hotel chain or management group. The hotel chain code is decided between vendors. This attribute is optional if the hotel is an independent property that can be identified by the HotelCode attribute.
@HotelCodeContext	0..1	A text field used to communicate the context (or source of - ex Sabre, Galileo, Worldspan, Amadeus) the HotelReferenceGroup codes.
HTNG_DeviceSearchRQ / Devices	1	A collection of Devices.
HTNG_DeviceSearchRQ / Devices / Device	0..n	Container element which holds information describing the device.
@Type	0..1	Specific classification of the device, ie Thermostat, SetTopBox, etc.
HTNG_DeviceSearchRQ / Devices / Device / DeviceLocation	0..1	
@RoomID	1	Room Number where this particular device is located.
HTNG_DeviceSearchRQ / Devices / Device / DeviceProfile	1	An element used to describe the set of available capabilities.
@ID	0..1*	A reference to an established Device Profile (See Device Capability Profile Management use case). If this value is populated, do not include the Capabilities collection.
HTNG_DeviceSearchRQ / Devices / Device / DeviceProfile Capabilities	0..1*	A collection of Capability elements used to convey device behaviors and interactions. This collection should only be present if not using Device Capability Profiles.

Element | @Attribute	**Num**	**Description/Contents**
HTNG_DeviceSearchRQ / Devices / Device / DeviceProfile Capabilities / Capability	1..n	An individual behavior.
@Type	1	One of 4 values (Per the DMS specification): • STATUS – Obtains the status of a given property of a device. • COMMAND – Instruction to the device to perform an action. • EVENT – Callback to indicate a device property has changed. • VENDOR_SPECIFIC – Vendor specific implementation.
@Operation	1	The method/function that can/will called.

4.5.2 Data Element Table – Response

Element | @Attribute	**Num**	**Description/Contents**
HTNG_DeviceSearchRS	1	Root element of the message.
@EchoToken	1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo token, the corresponding response message MUST include an echo token with an identical value.
@TimeStamp	1	Indicates the creation date and time of the message in UTC using the following format specified by ISO 8601; YYYY-MM-DDThh:mm:ssZ with time values using the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OpenTravel versioned messages, the version of the message is indicated by a decimal value.
HTNG_DeviceSearchRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_DeviceSearchRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.

Element | @Attribute	Num	Description/Contents
HTNG_DeviceSearchRS / Warnings / Warning	1..n	Used when a message has been successfully processed to report any warnings or business errors that occurred.
@Type	1	The Warning element MUST contain the Type attribute that uses a recommended set of values to indicate the warning type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).
HTNG_DeviceSearchRS / Errors	0..1	A collection of errors that occurred during the processing of a message.
HTNG_DeviceSearchRS / Errors / Error	1..n	An error that occurred during the processing of a message.
@Type	1	The Error element MUST contain the Type attribute that uses a recommended set of values to indicate the error type. The validating XSD can expect to accept values that it has NOT been explicitly coded for and process them by using Type ="Unknown". Refer to OpenTravel Code List Error Warning Type (EWT).

Element | @Attribute	Num	Description/Contents
@Status	0..1	If present, recommended values are those enumerated in the OTA_ErrorRS, (NotProcessed | Incomplete | Complete | Unknown) however, the data type is designated as string data, recognizing that trading partners may identify additional status conditions not included in the enumeration.
@ShortText	1	An abbreviated version of the error in textual format.
@Code	0..1	If present, this refers to a table of coded values exchanged between applications to identify errors or warnings. Refer to OpenTravel Code List Error Codes (ERR).
HTNG_DeviceSearchRS / Devices	1	A collection of Devices.
HTNG_DeviceSearchRS / Devices / Device	1..n	Container element which holds information describing the device.
@Type	1	Specific classification of the device, ie Thermostat, SetTopBox, etc.
@ID	1	An identifier that is unique within the network.
@ModelID	0..1	The model of the device as provided by the manufacturer.
@SerialNumber	0..1	The serial number of the device as provided by the manufacturer.
@MAC_Address	0..1	A unique identifier assigned to network devices.
@FriendlyName	1	A name used to identify a device in a room.
@Description	0..1	Text Description of the device.
@Class	1	General classification of the device, ie HVAC, TelevisionIntegration, etc.
@ModelName	0..1	The model of the device as provided by the manufacturer.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile	0..1	An element used to describe the set of available capabilities.
@ID	0..1*	A reference to an established Device Profile (See Device Capability Profile Management use case). If this value is populated, do not include the Capabilities collection.

Element | @Attribute	**Num**	**Description/Contents**
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities	0..1*	A collection of Capability elements used to convey device behaviors and interactions. This collection should only be present if not using Device Capability Profiles.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability	1..n	An individual behavior.
@Type	1	One of 4 values (Per the DMS specification): • STATUS – Obtains the status of a given property of a device. • COMMAND – Instruction to the device to perform an action. • EVENT – Callback to indicate a device property has changed. • VENDOR_SPECIFIC – Vendor specific implementation.
@Operation	1	The method/function that can/will called.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability / OutboundParameters	0..1	A collection of elements used to convey the parameters passed to the function call.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability / OutboundParameters / OutboundParameter	1..n	A single parameter.
@DataType	1	The XML schema data type of the parameter.
@Property	1	The name of the property.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability / OutboundParameters / OutboundParameter / KeyValueItems	0..1	A collection of enumerable items used to constrain the allowable values for the type referenced by the @Property.

Element | @Attribute	Num	Description/Contents
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability / OutboundParameters / OutboundParameter / KeyValueItems / KeyValueItem	1..n	An individual item in the enumeration.
@Key	1	The abstract display name corresponding to the Value.
@Value	1	Corresponds to the value of the OutboundParameter@DataType.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability / InboundParameters	0..1	A collection of elements used to convey the parameters returned by the function call.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability / InboundParameters / InboundParameter	1..n	A single parameter.
@DataType	1	The XML schema data type of the parameter.
@Property	1	The name of the property.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability / InboundParameters / InboundParameter / KeyValueItems	0..1	A collection of enumerable items used to constrain the allowable values for the type referenced by the @Property.
HTNG_DeviceSearchRS / Devices / Device / DeviceProfile Capabilities / Capability / InboundParameters / InboundParameter / KeyValueItems / KeyValueItem	1..n	An individual item in the enumeration.
@Key	1	The abstract display name corresponding to the Value.
@Value	1	Corresponds to the value of the OutboundParameter@DataType.

Element | @Attribute	Num	Description/Contents
HTNG_DeviceSearchRS / TPA_Extensions	0..1	This is used to exchange custom/implementation-specific values that are agreed upon between trading partners.

5 Appendices

5.1 Glossary of Terms

For the purpose of this document the following terms have been defined as follows:

Term	Definition
Device Capability Profile	A device profile defines a common set of capabilities supported by a particular class of device. A device profile makes it possible to maintain a single source of capability data that can be used to describe a large number of identical devices.
Device Discovery Requester	Any system interested in locating devices and their capabilities.
Device Discovery Responder	A repository of references (for the purpose of Addressing) to servers, systems, services and devices and their capabilities.
Device Registration Requester	A system or device capable of publishing device information to be made available to other systems or devices.
Device Registration Responder	A repository of references (for the purpose of Addressing) to servers, systems, services and devices and their capabilities.
Device Registration System	A repository of references (for the purpose of Addressing) to servers, systems, services and devices and their capabilities.
Discovery	A system that interacts with the Registration role to provide references to servers, systems, services and devices. Device Discovery utilities allow remote identification of the devices and create lists of the devices with relevant information. Its ultimate goal is to standardize and streamline the process of integration between applications and device services.
Endpoint	The IP network address of a device service. This is not to be confused with the physical address of a device.
Event Subscriber	A system interested in subscribing to device information update events.
Gateway	A system that acts as a proxy to other gateways, bridges or devices.
Hotel Information System	Any hotel software application or service containing features sets based on the status of guest room devices.
Intelligent Device	A piece of hardware that can be queried or changed, or report its state.
Device Registration Server	Stores information about devices and allows for the

	querying of device capabilities and how to address them.
Intelligent Guest Room (IGR) System	A system that provides the ability to query the status of guest room devices.
Registration	A process used to add a device to a UDDD registration system. The registration process involves defining the details (including endpoint) of all devices within the control span of the system/gateway/controller.
Room Status Publisher	A system that provides the ability to register for and publish events relating to the guest room devices.
Room Status Subscriber	A system interested in receiving real-time updates to the health and status of guest room devices.
Status Server	A type of gateway, capable of caching room device status/state and capable of monitoring and alerting.
Web Service	The W3C definition is “a software system designed to support interoperable machine-to-machine interaction over a network”.

5.2 Referenced Documents

The following table shows the documents upon which this document depends:

Document Title	**Location/URL**
HTNG Device Messaging Structure (DMS) Specification v1.0	http://collaboration.htng.org/inroom.ici/protected/documents.php?action=show&dcat=10&gdid=23121
HTNG Guest & Room Status Messaging (GRSM) Specification v2.0	http://collaboration.htng.org/specs/documents.php?action=show&dcat=42&gdid=23699
HTNG Protocol & Message Transport (PMT) Event Notification Specification v1.0	http://collaboration.htng.org/specs/documents.php?action=show&dcat=35&gdid=23182
OpenTravel Alliance Specifications	http://opentravel.org/Specifications/Default.aspx