



Kiosk Integration Specifications Release 2010B

**Version 2.0
22 October 2010**

About HTNG

Hotel Technology Next Generation ("HTNG") is a nonprofit organization with global scope, formed in 2002 to facilitate the development of next-generation, customer-centric technologies to better meet the needs of the global hotel community. HTNG's mission is to provide leadership that will facilitate the creation of one (or more) industry solution set(s) for the lodging industry that:

- Are modeled around the customer and allow for a rich definition and distribution of hotel products, beyond simply sleeping rooms;
- Comprise best-of-breed software components from existing vendors, and enable vendors to collaboratively produce world-class software products encompassing all major areas of technology spending: hotel operations, telecommunications, in-room entertainment, customer information systems, and electronic distribution;
- Properly exploit and leverage a base system architecture that provides integration and interoperability through messaging; and that provides security, redundancy, and high availability;
- Target the needs of hotel companies up to several hundred properties, that are too small to solve the issues themselves;
- Will reduce technology management cost and complexity while improving reliability and scalability; and
- Can be deployed globally, managed remotely, and outsourced to service providers where needed.

In June 2005, HTNG announced the first-ever "Branding and Certification Program" for hotel technology. This program will enable vendors to certify their products against open HTNG specifications, and to use the "HTNG Certified" logo in their advertising and collateral materials.

It will enable hotels to determine which vendors have completed certification of their products against which specific capabilities, and the environments in which performance is certified. HTNG's vision is to achieve a flexible technical environment that will allow multiple vendors' systems to interoperate and that will facilitate vendor alliances and the consolidation of applications, in order to provide hotels with easily managed, continually evolving, cost-effective solutions to meet their complete technology needs on a global basis.

Copyright 2010, Hotel Technology Next Generation

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the copyright owner.

For any software code contained within this specification, permission is hereby granted, free-of-charge, to any person obtaining a copy of this specification (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the above copyright notice and this permission notice being included in all copies or substantial portions of the Software.

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.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES, OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF, OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

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.

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	DOCUMENT HISTORY	5
1.1	FUNCTIONAL CHANGE LOG	5
2	ACKNOWLEDGEMENTS	6
3	DOCUMENT INFORMATION	7
3.1	DOCUMENT PURPOSE	7
3.2	SCOPE	7
3.3	AUDIENCE	7
3.4	OVERVIEW	7
3.5	DOCUMENT TERMS	8
3.6	REFERENCED DOCUMENTS	8
4	BUSINESS PROCESS	9
4.1	OVERVIEW	9
4.2	ROLES	9
4.2.1	Kiosk	9
4.2.2	Property Management System	9
4.3	BEHAVIOR	9
4.4	MESSAGE FLOWS	9
4.4.1	Walk-In Process	9
4.4.2	Check-In Process	9
4.4.3	In-House Process	10
4.4.4	Check-Out Process	10
5	USE CASES	12
5.1	SHOP	12
5.1.1	Messaging Use Case	12
5.2	BOOK	13
5.2.1	Messaging Use Case	13
5.3	UPSELL	14
5.3.1	Messaging Use Case	14
5.3.2	Data Element Table – Request	14
5.3.3	Sample Message – Request	15
5.3.4	Data Element Table – Response	15
5.3.5	Sample Message – Response	17
5.4	FIND BOOKING	18
5.4.1	Messaging Use Case	20
5.4.2	Data Element Table – Request	20
5.4.3	Sample Request Message	22
5.4.4	Data Element Table – Response	22
5.4.5	Sample Response Message	31
5.5	MODIFY BOOKING	33
5.5.1	Messaging Use Case	33
5.5.2	Data Element Table – Request	33
5.5.3	Modifying the Card on File	36
5.5.4	Assigning a Guest Room	36
5.5.5	Update Guest Name	37
5.6	CREATE SHAREWITH	38
5.6.1	Messaging Use Case	38
5.7	DEFINE MULTIPLE NAMES	39
5.7.1	Messaging Use Case	39
5.7.2	Data Element Table – Request	39
5.7.3	Sample Request Message	40
5.7.4	Data Element Table – Response	41
5.7.5	Sample Response Message	42
5.8	REQUEST ROOM LIST	42
5.8.1	Messaging Use Case	42
5.8.2	Data Element Table – Request	43
5.8.3	Sample Message – Request	45

5.8.4	Data Element Table – Response	45
5.8.5	Sample Message – Response.....	47
5.9	CHANGE/UPDATE ROOM.....	48
5.9.1	Messaging Use Case.....	48
5.10	AUTHORIZE CARD	49
5.10.1	Messaging Use Case.....	49
5.11	CHECK-IN.....	50
5.11.1	Messaging Use Case.....	50
5.11.2	Data Element Table – Request	51
5.11.3	Sample Request Message	51
5.11.4	Data Element Table – Response	52
5.11.5	Sample Response Message	52
5.12	CUT KEY REQUEST	53
5.12.1	Messaging Use Case.....	53
5.12.2	Data Element Table – Request	54
5.12.3	Sample Request Message – Encode Hotel Room Key	57
5.12.4	Sample Request Message – Request Hotel Room Key Encoding	57
5.12.5	Data Element Table – Response	57
5.12.6	Sample Response Message – Encode Hotel Room Key	60
5.12.7	Sample Response Message - Request Hotel Room Key Encoding.....	60
5.13	RETRIEVE FOLIO	60
5.13.1	Messaging Use Case.....	60
5.13.2	Data Element Table – Request	61
5.13.3	Sample Request Message	61
5.13.4	Data Element Table – Response	61
5.13.5	Sample Response Message	68
5.14	CHECK-OUT	69
5.14.1	Messaging Use Case.....	69
5.14.2	Data Element Table – Request	70
5.14.3	Sample Request Message	71
5.14.4	Data Element Table – Response	72
5.14.5	Sample Response Message	72
5.15	DISPLAY GUEST MESSAGES	73
5.15.1	Messaging Use Case.....	73
5.15.2	Data Element Table – Request	73
5.15.3	Sample Message – Request.....	74
5.15.4	Data Element Table – Response	74
5.15.5	Sample Message – Response.....	75
5.16	CHANGE GUEST MESSAGE STATUS	76
5.16.1	Messaging Use Case.....	76
5.16.2	Data Element Table – Request	76
5.16.3	Sample Message – Request.....	77
5.16.4	Data Element Table – Response	77
5.16.5	Sample Message – Response.....	78
6	APPENDIX 1	79
6.1	WEB SERVICE DESCRIPTION.....	79

1 Document History

1.1 Functional Change Log

Version	Date	Comments
1.0	23 April 2010	V1 functionality included: Find Booking, Modify Booking, Authorize Card, Check-In, Cut Key Request, Retrieve Folio, Check-Out.
2.0	22 October 2010	Additional functionality for V2: Shop, Book, Upsell, Create ShareWith, Define Multiple Names, Request Room List, Change/Update Room, Duplicate and Replacement Key Cutting, Display Guest Messages, Change Guest Message Status.

2 Acknowledgements

HTNG gratefully acknowledges the contributions of the following people in the development of this document:

Workgroup Member	Company
Gary Gage	Agilysys
Laurent Cardot	Ariane Systems
David Walpole	Ariane Systems
James Cornthwaite	It Just Works Software
Ryan de Laplante	It Just Works Software
Doug MacRae	It Just Works Software
Mary Turner	SoftBrands / Infor
Bill Rosenberg	Starwood Hotels & Resorts Worldwide

3 Document Information

3.1 Document Purpose

As self-service technology becomes more popular in everyday life, hotel check-in/check-out kiosks are increasingly being seen as a viable alternative in the hospitality industry. The interaction between Kiosk and Property Management Systems (PMS) is complex. There are many Kiosk and PMS providers, however, there are no interface standards implemented to facilitate the many aspects of the hospitality business processes, including reservations, room availability, payment processing and key encoding for hotel check-in/check-out kiosks.

The purpose of this Kiosk Integration document is to provide a specification for implementation of the HTNG open-standard solution for the interaction between Kiosk and PMS systems. By standardizing the data request and response messages, Kiosk and PMS vendors can share the same message set and hoteliers can also write to the specifications, thereby adding information from other disparate sources into a common framework. This document defines the standard messaging to resolve this problem. A first release was approved during the 2010A certification process with basic functionalities. This version adds a list of new enhanced functionalities for the 2010B certification process.

3.2 Scope

The scope of this document includes, directly or by reference, all information required to implement the integration between Kiosk and PMS systems for guest check-in/check-out processes. It does not include information needed to implement other self-service features such as spa reservations, additional service ordering, or specifications developed by other 3rd parties, however, note that these features may be incorporated in a future version.

3.3 Audience

The intended audience of this document is development teams and system designers of Kiosk or PMS seeking to implement these standardized interface specifications within their products. This document also provides Business Process Flows that may be used by Hotel Groups looking to standardize their interfaces within their hotel architectures.

3.4 Overview

By standardizing the data request and response messages, connecting a hotel check-in/check-out Kiosk to a PMS will be easier by utilizing the business processes, use cases and messaging included in this document. To ensure maximum compatibility with existing technologies, the workgroup has based its initial work on OpenTravel Alliance schemas and leveraged specifications developed by other HTNG workgroups.

The following information is included for each of the business processes noted below.

- MESSAGING USE CASE
- DATA ELEMENT TABLE – REQUEST
- SAMPLE REQUEST MESSAGE
- DATA ELEMENT TABLE – RESPONSE
- SAMPLE RESPONSE MESSAGE

Business Processes:

- FIND BOOKING
- MODIFY BOOKING
- AUTHORIZE CARD
- CHECK-IN
- CUT KEY REQUEST
- RETRIEVE FOLIO
- CHECK-OUT
- DEFINE MULTIPLE NAMES
- REQUEST ROOM LIST
- CHANGE/UPDATE ROOM
- SHOP
- BOOK
- UPSELL
- DISPLAY GUEST MESSAGES
- CHANGE GUEST MESSAGE STATUS
- CUT KEY REQUEST - DUPLICATE KEY
- CUT KEY REQUEST - REPLACEMENT KEY
- CREATE SHAREWITH

3.5 Document Terms

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

Term	Definition
PMS	Property Management System
OpenTravel or OTA	OpenTravel Alliance (http://www.opentravel.org)

3.6 Referenced Documents

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

Name	Location
HTNG Payment Systems & Data Security - Payment Processing Specification 2010A	http://www.opengroup.org/htng/specs/
HTNG Product Distribution Specification 2010A	http://www.opengroup.org/htng/specs/
OpenTravel Alliance Specifications	http://www.opentravel.org

4 Business Process

4.1 Overview

By creating a standard interface protocol between Kiosks and PMSs, this document gives flexibility to the hotels and lowers the overall costs of implementation by helping their vendors to develop a straightforward solution.

4.2 Roles

This specification defines the following roles:

4.2.1 Kiosk

The Kiosk is the system that will allow the guest, in a self-service manner, to access services otherwise performed by hotel staff.

4.2.2 Property Management System

The Property Management System provides a series of services for access by the Kiosk in response to the requests that are transmitted during the check-in/check-out process.

4.3 Behavior

Kiosks require specific information from PMSs in order to identify guests, retrieve their reservation details, input necessary information, add options and select preferences, confirm and pay, and cut a room key, in addition to printing documents.

4.4 Message Flows

For clarity, the use cases described hereafter are listed in chronological order.

4.4.1 Walk-In Process

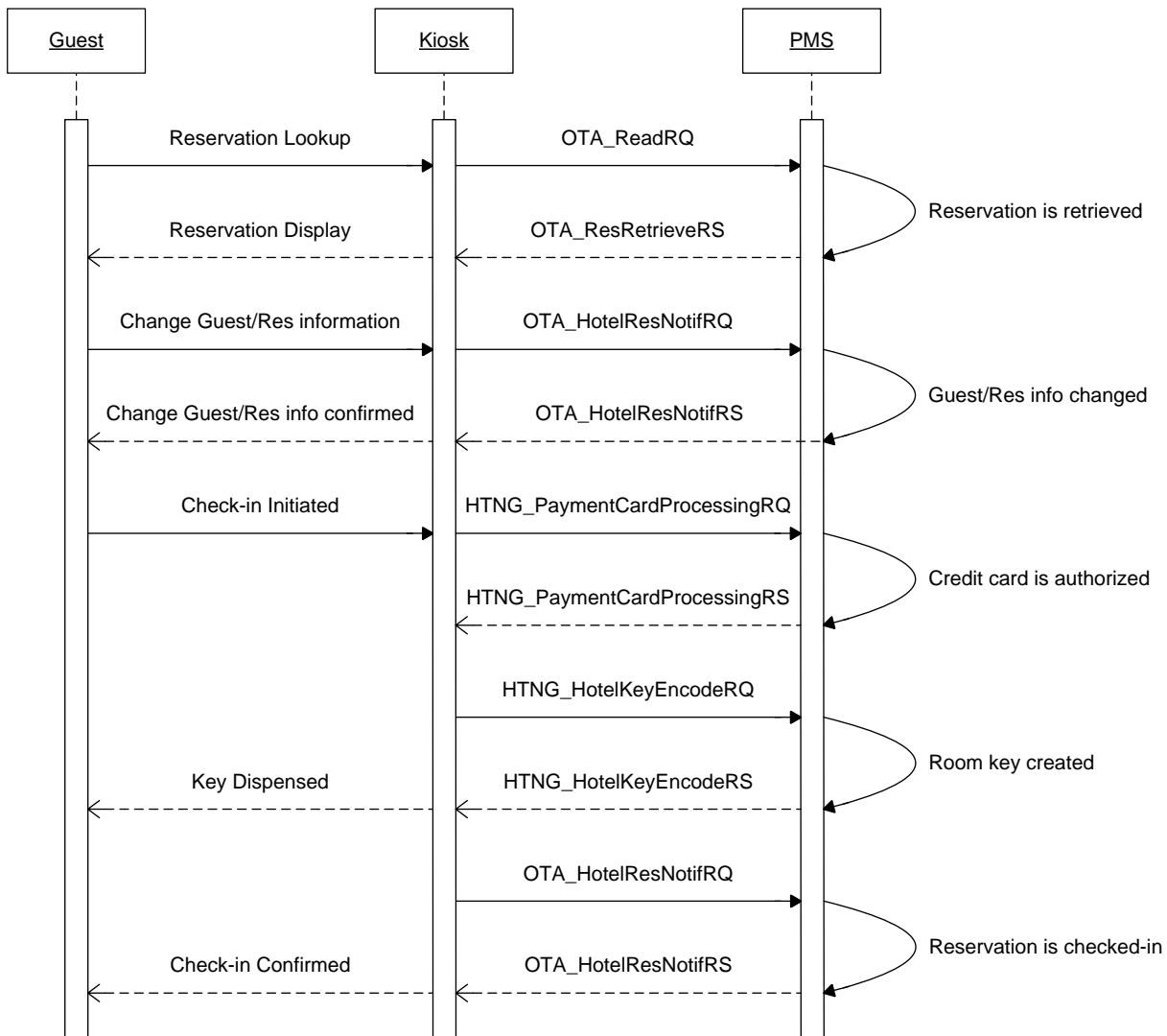
A guest begins the walk-in process by inputting his wishes: number of nights and number of people on the kiosk terminal. The kiosk will show a list of availabilities and confirm, after guest selection, reservation details that will apply to this guest. Guest will be asked to add information required by the hotel (name, ID...).

The scenario will then move to a classical check-in process as described hereafter.

4.4.2 Check-In Process

A guest begins the check-in process by verifying his identity on the kiosk terminal. The kiosk will request the reservation that applies to this guest. If multiple reservations are found, further refinement must be performed until a single reservation is located and presented on the kiosk screen.

The kiosk now needs to send the PMS the necessary credit card information for authorization, a request to encode a room key and to update the reservation's status to checked-in.

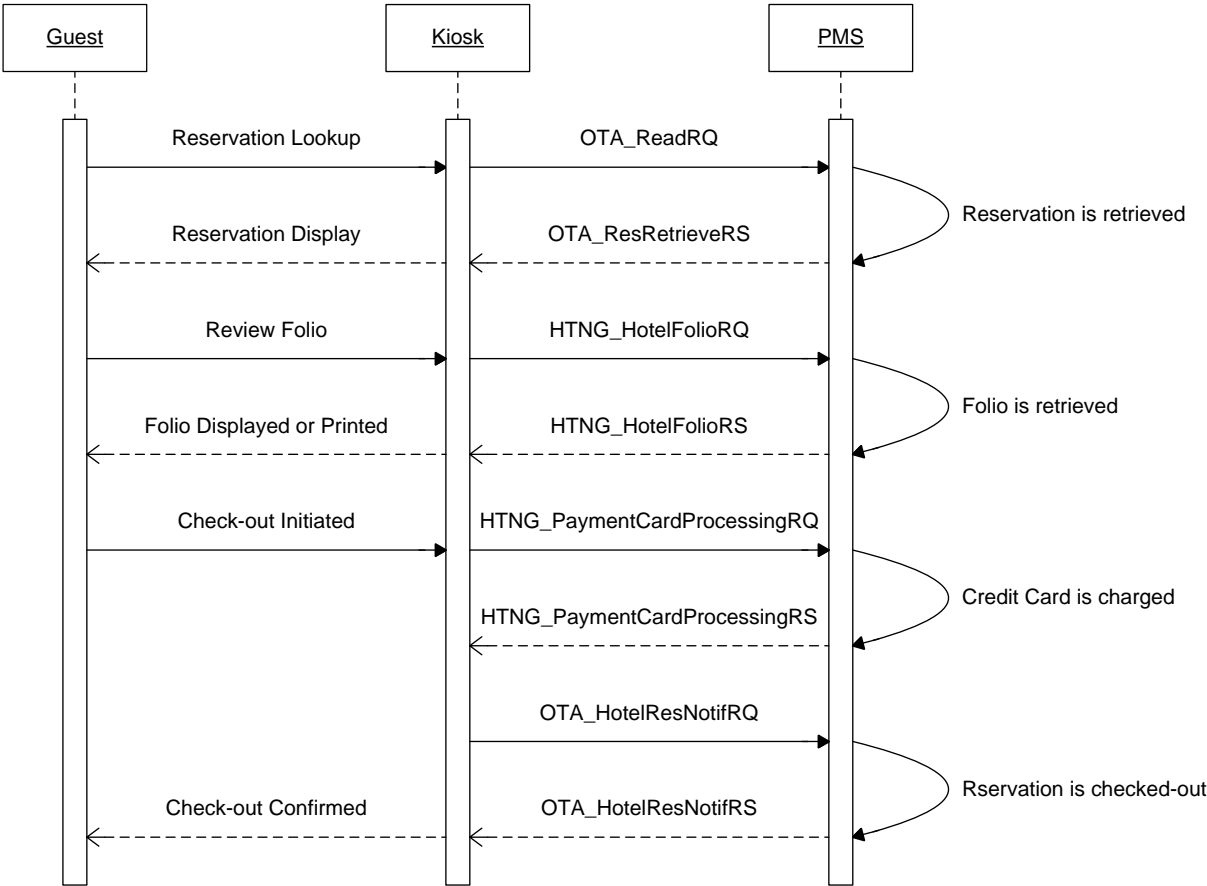


4.4.3 In-House Process

Guests will be able, during their stay, to go to the kiosk to ask for room key replacement (if they lose it) or a room key duplicate (if they need an additional key). They will be asked to identify themselves before the kiosk cuts the requested keys.

4.4.4 Check-Out Process

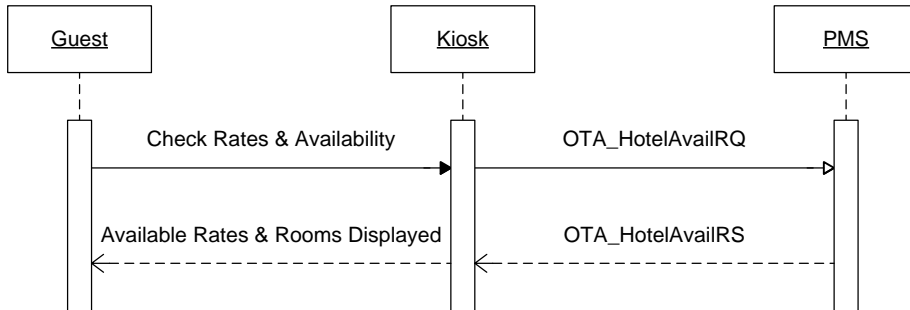
A guest begins the check-out process by verifying his identity on the kiosk terminal. Kiosk will retrieve the billing information and display the invoice. The guest will then be guided through the payment process and will receive payment confirmation.



5 Use Cases

5.1 Shop

5.1.1 Messaging Use Case

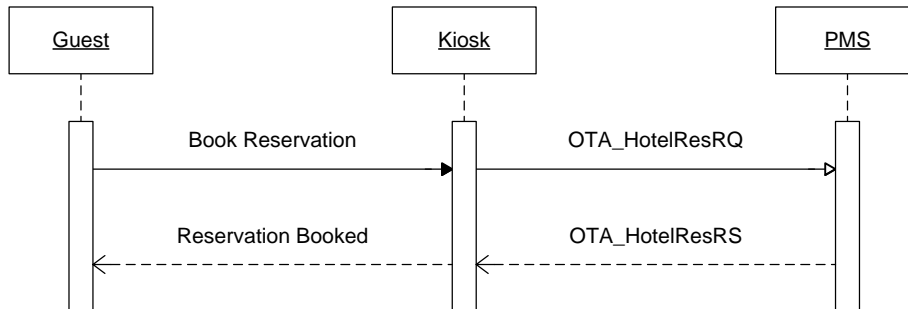


This use case is based upon the 2010A Product Distribution Workgroup specification. Please visit <http://collaboration.htng.org/specs/> for more information.

Use Case Name:	Shop
Summary:	Walk-in guest wishes to browse available rooms and rates.
Basic Course of Events:	<p>The use case begins when the guest starts to check in at the kiosk</p> <ol style="list-style-type: none"> 1. Guest chooses walk-in option 2. Kiosk collects stay information from guest (departure, #guests, and possibly desired room features) 3. The kiosk performs an availability lookup to the PMS 4. The PMS returns a list of room types (with rates) which match the entered criteria <p>The use case ends when the guest is presented with available room types and rates.</p>
Exception Path:	<p>In Step 1, business rules dictate that walk-ins are not permitted or are disabled; the use case does not apply.</p> <p>In Step 4, no rooms are available; The PMS sends an error message to the kiosk directing the guest to the front desk.</p>
Alternative Paths:	None.
Trigger:	The guest has indicated they are a walk-in rather than searching for an existing booking.
Assumptions:	The guest does not yet have a reservation booked. PMS may limit the rate codes and room types available to walk-in guests. PMS may not contain logic for room selection based on guest preferences.
Preconditions:	Guest has no reservation and no knowledge of availability.
Postconditions:	Guest is informed of available room types and available rates .

5.2 Book

5.2.1 Messaging Use Case

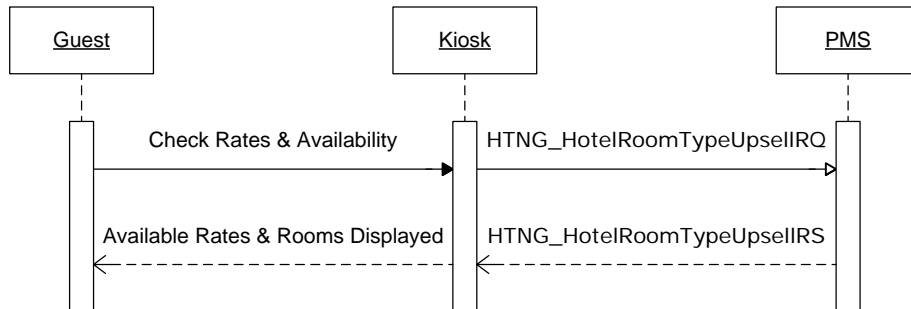


This use case is based upon the 2010A Product Distribution Workgroup specification. Please visit <http://collaboration.htng.org/specs/> for more information.

Use Case Name:	Book
Summary:	Walk-in guest has browsed the list of available room types/rates and wishes to stay at hotel. Kiosk must create a reservation. This is essentially a 'create reservation with requested room and rate code'.
Basic Course of Events:	<p>The use case begins after the guest has been presented with available room types and rates.</p> <ol style="list-style-type: none"> 1. The guest selects the room type and rate they desire 2. The kiosk collects any additional information necessary for creating a booking in the PMS (name, address, etc) 3. The kiosk requests a new reservation from the PMS using the specified room rate and room type, plus additional stay criteria provided in the SHOP use case (departure, number of guests) 4. The PMS returns a new reservation to the kiosk which is in an arriving state (not yet checked in). <p>The use case ends when the arriving reservation has been created.</p>
Exception Path:	<p>In step 3, The hotel has sold the last room of the selected room type</p> <ol style="list-style-type: none"> 1. The PMS sends an error message to the kiosk 2. The kiosk returns the guest to the room type/rate selection screen
Alternative Paths:	Depending on PMS requirements, steps 2 and 3 could be reversed, meaning the kiosk could create an arriving reservation containing no guest detail in order to guarantee a room can be allocated, since it can take some time for the guest to enter their stay details.
Trigger:	Guest selects their desired room type/rate from the list presented from the initial shop request.
Assumptions:	None.
Preconditions:	Guest has no reservation booked at the hotel.
Postconditions:	New reservation has been created in PMS.

5.3 Upsell

5.3.1 Messaging Use Case



Use Case Name:	Upsell
Summary:	The guest opts to upgrade their reservation room type.
Basic Course of Events:	<p>The use case begins after the guest has found their booking (see Find Booking)</p> <ol style="list-style-type: none"> 1. The kiosk performs an upsell search 2. The PMS returns a list of available room types, with price delta information (upsell fee / rate change) for each 3. The kiosk asks the guest if they would like to upgrade 4. The guest selects upgrade 5. The kiosk presents the upgrade options the the guest 6. The guest selects an upgrade option 7. The kiosk performs a reservation modification to select the upsold room type and offered rate.*** <p>The use case ends when the reservation has been modified.</p>
Exception Path:	In step 3, there are no room types available for upsell, or the Guest reservation cannot be modified (hard block or other reason), or Upsells are not configured in the PMS, in which case the guest is not offered the option to upgrade.
Alternative Paths:	Steps 3 and 4 could come first if the kiosk implementation is designed to seek the upgrade request from the guest prior to inquiring with the PMS for availability.
Trigger:	At check in, guest desires an upgraded room type.
Assumptions:	Guest assigned room type must not be blocked. Upsell rooms are configured in the PMS. Upsell charges can be handled by either a room rate change or by a recurring charge, but this is transparent to kiosk (only price delta is provided).
Preconditions:	Guest has an arriving reservation with or without a room assigned.
Postconditions:	Guest has an arriving reservation for which the room type and rate have changed; room may not be assigned.

*** The upsell information needs to be inserted in a OTA_HotelResNotifRQ message in the following xpath location:
 \\OTA_HotelResNotifRQ\HotelReservations\HotelReservation\RoomStays\RoomStay\RoomRates\RoomRate\
 Rates\Rate\Fees\Fee.

5.3.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
HTNG_HotelRoomTypeUpsellIRQ	1	A message used to determine if the reservation is eligible for an upsell from their currently reserved room tyoe to a different room type at an incremental charge.

Element @Attribute	Num	Description/Contents
@EchoToken	0..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	0..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 OTA versioned messages, the version of the message is indicated by a decimal value.
@Target	0..1	Used to indicate whether the request is for the Test or Production system. Value can be either 'Test' or 'Production'.
HTNG_HotelRoomTypeUpsellRQ / POS / Source	1	Provides information on the source of a request.
@TerminalID	0..1	The electronic address of the device from which information is entered.
HTNG_HotelRoomTypeUpsellRQ / POS / Source / RequestorID	0..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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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_HotelRoomTypeUpsellRQ / UniqueID	1	The unique identifier element allows the trading partners to uniquely identify each (i.e., the entire message) for transaction tracability.
@Type	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.

5.3.3 Sample Message – Request

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_HotelRoomTypeUpsellRQ EchoToken="a" TimeStamp="2010-12-17T09:30:47Z" Version="0.0" Target="Test"
xmlns="http://htng.org/2010B" xmlns:ota="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <ota:Source TerminalID="KioskABC">
      <ota:RequestorID Type="0" ID_Context="Kiosk" ID="123"/>
    </ota:Source>
  </POS>
  <UniqueID Type="14" ID_Context="PMS123" ID="RES123"/>
</HTNG_HotelRoomTypeUpsellRQ>
```

5.3.4 Data Element Table – Response

Element @Attribute	Num	Description/Contents
HTNG_HotelRoomTypeUpsellIRS	1	A message used to determine if the reservation is eligible for an upsell from their currently reserved room type to a different room type at an incremental charge.
@EchoToken	0..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

Element @Attribute	Num	Description/Contents
		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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
HTNG_HotelRoomTypeUpsellRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_HotelRoomTypeUpsellRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
HTNG_HotelRoomTypeUpsellRS / 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_HotelRoomTypeUpsellRS / GuestRoomTypes	1	A collection of GuestRoomType objects.
HTNG_HotelRoomTypeUpsellRS / GuestRoomTypes / GuestRoomType	0..n	An entity representing a possible upsell scenario.
HTNG_HotelRoomTypeUpsellRS / GuestRoomTypes / GuestRoomType / Room	1	Attributes to describe the room from room category to location to view to bed type.
@RoomCategory	0..1	Indicates the category of the room. Typical values would be Moderate, Standard, or Deluxe. Refer to OpenTravel Code List Segment Category Code (SEG).
@NonSmoking	0..1	Non-smoking indicator.
@RoomTypeCode	1	Specific system room type code, i.e.,: A1K, A1Q, etc.
@Composite	0..1	Indicates that the room (suite) is a composite of smaller units.
@RoomType	0..1	(Formerly, RoomInventoryCode) A code value that indicates the type of room for which this request is made, e.g., double, king, etc. Values may use the Hotel Descriptive Content table or a codes specific to the property or hotel brand.
@Floor	0..1	Floor on which the room is located.
@RoomViewCode	0..1	Indicates the view of the room. Typical values would be "Ocean view", "Pool view" or "Garden View". Refer to OpenTravel Code List Room View Type (RVT).
@BedTypeCode	0..1	Indicates the type of bed(s) found in the room. Typical values would be Double, Twin, Queen, or King. Refer to OpenTravel Code List Bed Type Code (BED).
@RoomLocateCode	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_HotelRoomTypeUpsellRS / GuestRoomTypes / GuestRoomType /	1	Collection of fees charged at the room level.

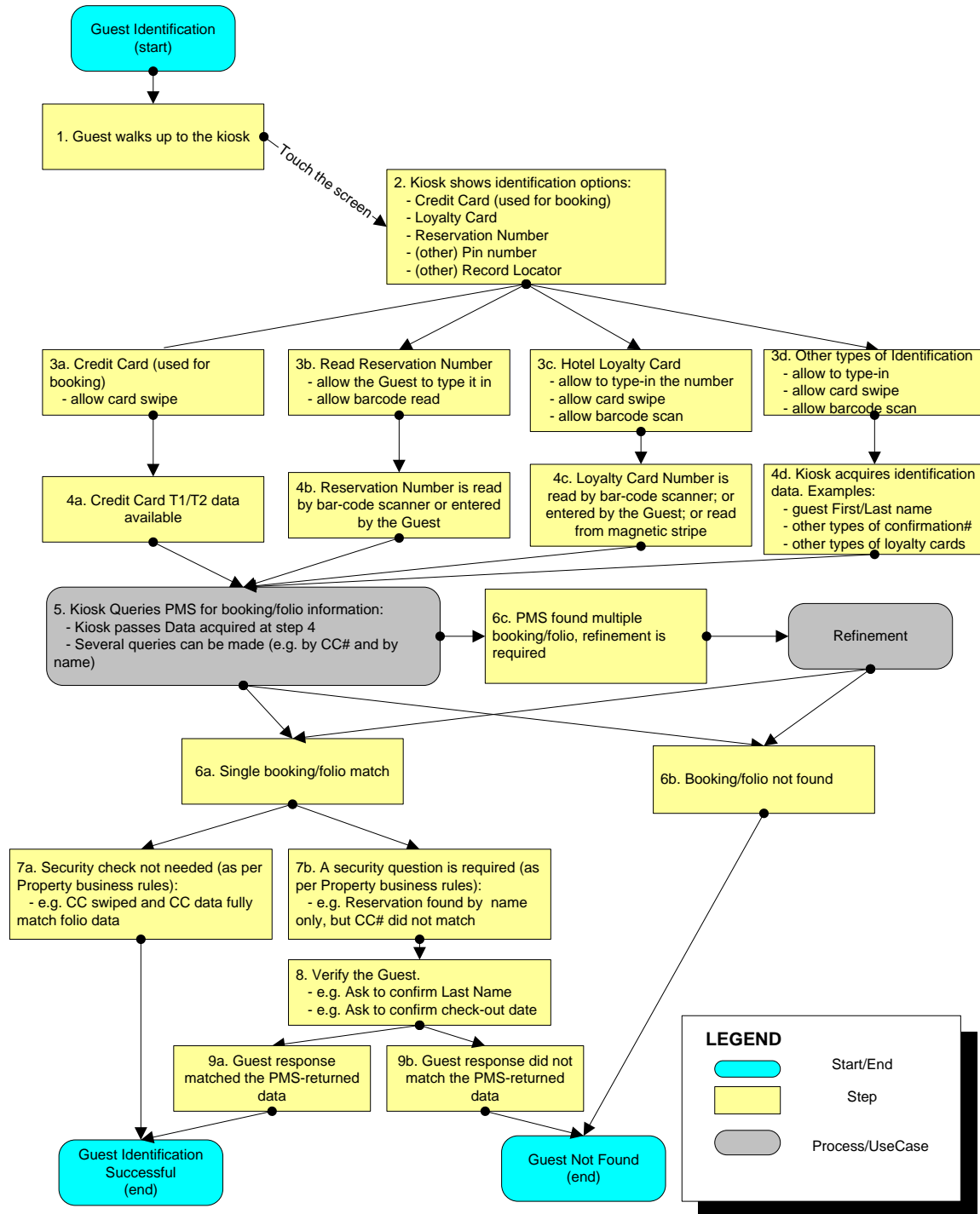
Element @Attribute	Num	Description/Contents
RoomLevelFees		
HTNG_HotelRoomTypeUpsellRS / GuestRoomTypes / GuestRoomType / RoomLevelFees / Fee	1	Fee Amount that is applied to the rate. Fees are used for non tax amounts like service charges.
@Type	1	Used to indicate if the amount is inclusive or exclusive of other charges, such as taxes, or is cumulative (amounts have been added to each other).
@MandatoryIndicator	1	When true, indicates the fee is mandatory. When false, the fee is not mandatory. In order for this message to have meaning, it should always be set to "true".
@TaxableIndicator	1	When true, indicates that the fee is subject to tax.
@ExpireDate	1	Indicates the ending date.
@EffectiveDate	1	Indicates the starting date.
@Amount	1	A monetary amount.
@DecimalPlaces	1	Indicates the number of decimal places for a particular currency. This is equivalent to the ISO 4217 standard "minor unit". Typically used when the amount provided includes the minor unit of currency without a decimal point (e.g., USD 8500 needs DecimalPlaces="2" to represent \$85).
@TaxInclusive	1	Indicates whether taxes are included when figuring the fees.
@ChargeUnit	1	The unit for which the charge applies (e.g., room, person, seat). Refer to OpenTravel Code List Charge Type (CHG).
@Code	1	Code identifying the fee (e.g., agency fee, municipality fee). Refer to OpenTravel Code List Fee Tax Type (FTT).
HTNG_HotelRoomTypeUpsellRS / GuestRoomTypes / GuestRoomType / Description	1	Description of the Room.
HTNG_HotelRoomTypeUpsellRS / GuestRoomTypes / GuestRoomType / Description / Text	1	Formatted text content.

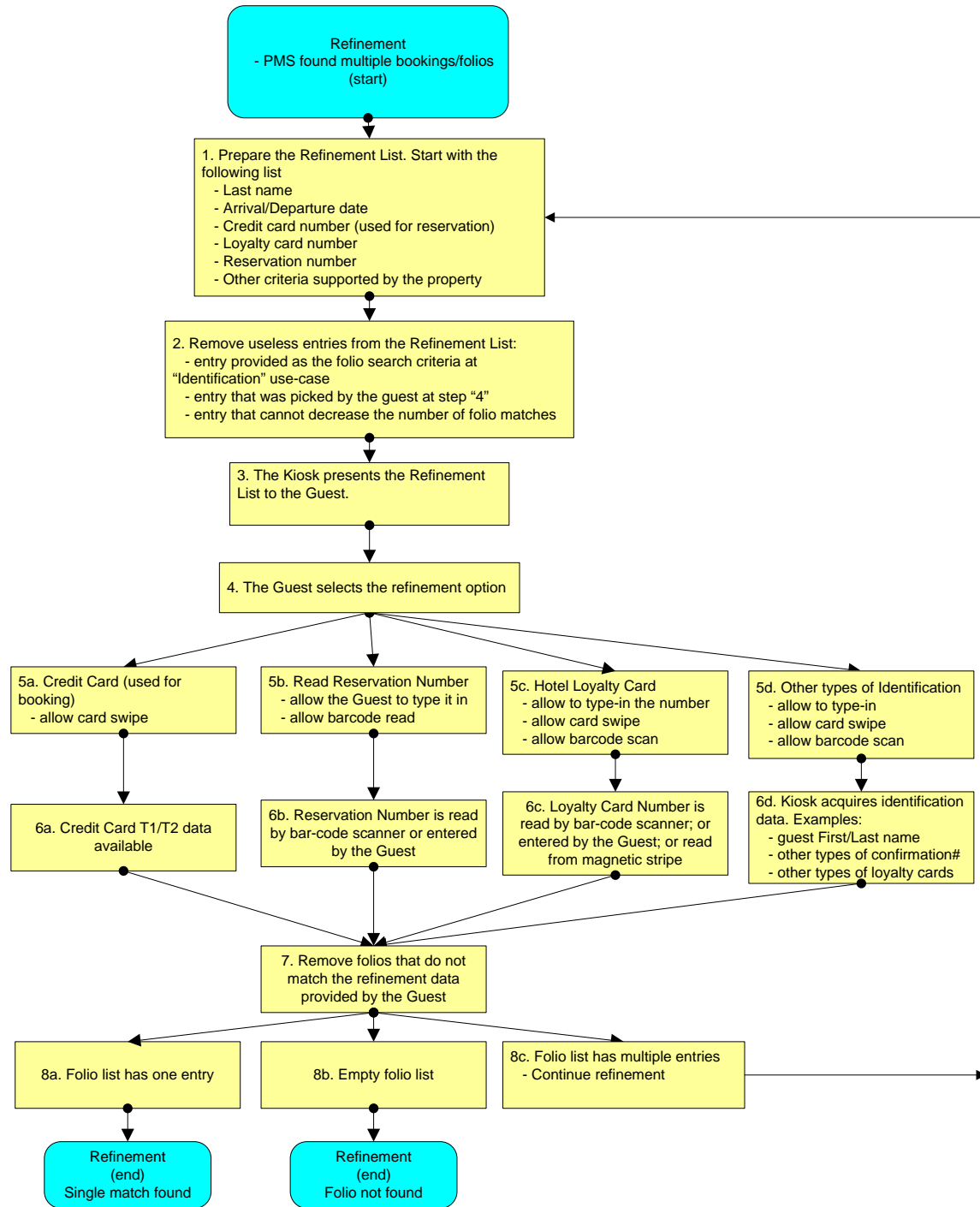
5.3.5 Sample Message – Response

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_HotelRoomTypeUpsellRS EchoToken="a" TimeStamp="2010-12-17T09:30:47Z" Version="0.0" Target="Test"
xmlns="http://htng.org/2010B" xmlns:ota="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Success/>
  <Warnings>
    <ota:Warning Type="0" Status="a" ShortText="a" Code="0">String</ota:Warning>
  </Warnings>
  <UniqueID Type="0" ID_Context="a" ID="a"/>
  <GuestRoomTypes>
    <GuestRoomType>
      <ota:Room RoomCategory="0" NonSmoking="true" RoomTypeCode="a" Composite="true"
RoomType="a" Floor="1" RoomViewCode="0" BedTypeCode="0" RoomLocationCode="0"/>
      <ota:RoomLevelFees>
        <ota:Fee Type="Inclusive" MandatoryIndicator="true" TaxableIndicator="true"
ExpireDate="1967-08-13" EffectiveDate="1967-08-13" Amount="1.123" DecimalPlaces="1" TaxInclusive="true"
ChargeUnit="1" Code="16">
          <ota:Taxes Amount="1.123" DecimalPlaces="1"/>
        </ota:Fee>
      </ota:RoomLevelFees>
      <ota:Description Name="a">
        <ota:Text>String</ota:Text>
      </ota:Description>
    </GuestRoomType>
  </GuestRoomTypes>
</HTNG_HotelRoomTypeUpsellRS>
```

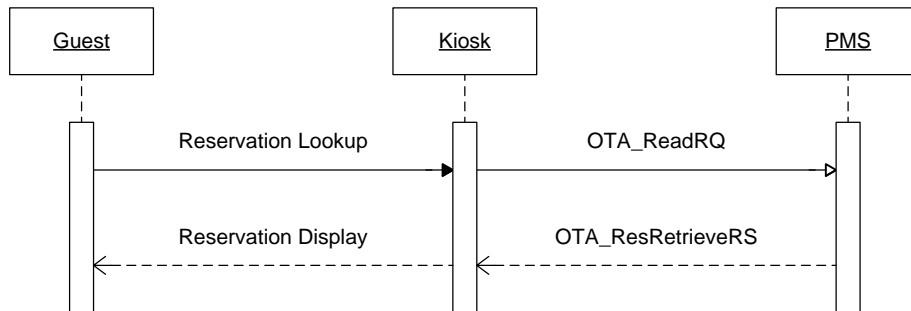
5.4 Find Booking

This is the process used to identify the guest, and locate the one reservation pertinent to this check in.





5.4.1 Messaging Use Case



Use Case Name:	Find Booking
Summary:	The guest goes to a kiosk to check in and provides one or more IDs to validate him or herself. The Kiosk searches for one booking that meets the criteria and returns booking summary data.
Basic Course of Events:	The Use Case Begins when the Guest starts to Check In at the Kiosk <ol style="list-style-type: none"> 1. The Guest enters some methods of Identification 2. The Kiosk performs a Booking Search for bookings for a guest arriving today that match the ID entered 3. The PMS finds a single booking that matches the entered criteria 4. The PMS returns a booking summary to the Kiosk 5. The Kiosk displays the booking summary and asks the guest to confirm. 6. The Guest Confirms that this is their booking The Use Case Ends when the Guest has found their booking.
Alternative Paths:	In Step 3 the PMS finds more than 1 matching booking <ol style="list-style-type: none"> 1. The PMS sends an error message to the Kiosk 2. The Kiosk ask the guest to enter additional ID 3. The Kiosk sends a new search to the PMS with more criteria 4. The PMS finds a single match. 5. The Use case returns to the Basic Flow In Step 3 the PMS finds zero bookings matching the selection <ol style="list-style-type: none"> 1. The PMS Sends an error message to the Kiosk 2. The Kiosk advises the guest that a booking cannot be found. 3. The Kiosk invites the guest to try again or go to the desk.
Trigger:	The Guest initiates a Check-In.
Assumptions:	The guest does have a booking at this hotel arriving today.
Preconditions:	The guest has not found his booking.
Postconditions:	The Guest Has found his booking.

5.4.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
OTA_ReadRQ	1	A generic message, available as an action on several OpenTravel services which requests a server to read and return the document type identified by the UniqueID element.
@EchoToken	0..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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
OTA_ReadRQ / 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	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	1	Used to identify the source of the identifier (e.g., IATA, ABTA).
OTA_ReadRQ / ReadRequests / ReadRequest / UniqueID	0..1	The booking reference for the reservation.
@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 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	1	Used to identify the source of the identifier (e.g., IATA, ABTA).
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / PersonName	0..1	Detailed name information
@PartialName	1	When true, the full name is not provided.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / PersonName / GivenName		Given name, first name or names.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / PersonName / SurName	0..1	Family name, last name.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / Email	0..1	Information on an email address.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / TelephoneInfo	0..1	Information on a telephone number.
@PhoneNumber	0..1	Telephone number assigned to a single location.
@CountryAccessCode	0..1	Code assigned by telecommunications authorities for international country access identifier.
@AreaCityCode	0..1	Code assigned for telephones in a specific region, city, or area.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / PaymentCard	0..1	Payment Card information.
@CardNumber	0..1	Credit card number embossed on the card.
@CardCode	0..1	The 2 character code of the credit card issuer.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / CustLoyalty	0..1	Program rewarding frequent use by accumulating credits for services provided by vendors.
@ExpireDate	0..1	Indicates the ending date.
@MembershipID	0..1	Unique identifier of the member in the program (membership number, account number, etc.).
@ProgramID	0..1	Identifier to indicate the company owner of the loyalty program.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / ReservationTimeSpan	0..1	The start and end date of the reservation.

Element @Attribute	Num	Description/Contents
@End	0..1	The ending value of the time span.
@Start	0..1	The starting value of the time span.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / TPA_Extensions / RoomType	0..1	Provides details regarding rooms, usually guest rooms.
@RoomID	0..1	A string value representing the unique identification of a room if the request is looking for a specific room.
OTA_ReadRQ / ReadRequests / ReadRequest / Verification / TPA_Extensions / Document	0..1	Provides information on a specific document.
@DocID	0..1	Unique number assigned by authorities to document.
@DocType	0..1	Indicates the type of document (e.g., Passport, Military ID, Drivers License, national ID, Vaccination Certificate). Refer to OpenTravel Code List Document Type (DOC).

5.4.3 Sample Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_ReadRQ EchoToken="6cfalea3-df0f-497f-9aaf-e927fc11affc" TimeStamp="2010-02-12T12:26:47" Version="1.0"
xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="TRM123">
      <RequestorID Type="18" ID="KSKSYS123"/>
    </Source>
  </POS>
  <ReadRequests>
    <ReadRequest>
      <UniqueID Type="14" ID="RES123456"/>
      <Verification>
        <PersonName PartialName="false">
          <GivenName>John</GivenName>
          <Surname>Smith</Surname>
        </PersonName>
        <Email>john.smith@gmail.com</Email>
        <TelephoneInfo PhoneNumber="555-1212" CountryAccessCode="1" AreaCityCode="800"/>
        <PaymentCard CardNumber="4444333322221111"/>
        <CustLoyalty ExpireDate="2010-08-31" MembershipID="MMB123456"
ProgramID="FRQTRV"/>
        <ReservationTimeSpan End="2010-08-15" Start="2010-08-13"/>
        <TPA_Extensions>
          <RoomType RoomID="1706"/>
          <Document DocID="6746783-463217643-64326" DocType="2"/>
        </TPA_Extensions>
      </Verification>
    </ReadRequest>
  </ReadRequests>
</OTA_ReadRQ>
```

5.4.4 Data Element Table – Response

Element @Attribute	Num	Description/Contents
OTA_ResRetrieveRS	1	This message returns a list of reservations when an exact match on a read request could not be made or the request was to return a list of reservations meeting specified criteria.
@EchoToken	0..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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
OTA_ResRetrieveRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
OTA_ResRetrieveRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
OTA_ResRetrieveRS / 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).
OTA_ResRetrieveRS / ReservationList / HotelReservation	0..n	One line in a list of hotel reservations. It may contain the reservation ID, booked date, arrival and departure dates, number of nights and rooms, hotel info, guest info, and room info.
OTA_ResRetrieveRS / ReservationList / HotelReservation / 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.
OTA_ResRetrieveRS / ReservationList / HotelReservation / UniqueID	1	Used to provide PMS and/or CRS identifiers.
@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 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	1	Used to identify the source of the identifier (e.g., IATA, ABTA).
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays	0..n	Collection of room stays.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay	0..n	Details on the Room Stay including Guest Counts, Time Span of this Room Stay, pointers to Res Guests, guest Memberships, Comments and Special Requests pertaining to this particular Room Stay and finally financial information related to the Room Stay, including Guarantee, Deposit and Payment and Cancellation Penalties.
@RoomStayStatus	1	Indicates the status of the reservation.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / RoomTypes	1	A collection of Room Types associated with a particular Room Stay.
OTA_ResRetrieveRS / ReservationList /	1	Provides details regarding rooms, usually guest rooms.

Element @Attribute	Num	Description/Contents
HotelReservation / RoomStays / RoomStay / RoomTypes / RoomType		
@NonSmoking	0..1	Non-smoking indicator.
@RoomTypeCode	1	Specific system room type code, i.e., A1K, A1Q, etc.
@Composite		Indicates that the room (suite) is a composite of smaller units.
@RoomID	0..1	A string value representing the unique identification of a room if the request is looking for a specific room.
@NumberOfUnits	1	The number of rooms.
@Floor	0..1	Floor on which the room is located.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / RoomTypes / RoomType / RoomDescription / Text	0..1	Textual information regarding the room.
@Language	0..1	Language identification.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / RoomTypes / RoomType / AdditionalDetails / AdditionalDetail / DetailDescription / Text	1	Used to define the type of information being sent (e.g., rate description, property description, room information). Refer to OpenTravel Code List Additional Detail Type (ADT).
@Language	0..1	Language identification.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / RoomTypes / RoomType / Occupancy	0..1	Provides parameters of occupancy limits.
@MaxOccupancy	1	Maximum number of persons allowed in a unit of accommodation or place.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / RatePlans / RatePlan	1	Defines the details of the rate plan as used in the booking process.
@PriceViewableInd	1	When true, the price for this rate plan can be viewed by the guest. When false, the guest must contact another entity to obtain price information
@RatePlanCode	1	A string value may be used to request a particular code or an ID if the guest qualifies for a specific rate, such as AARP, AAA, a corporate rate, etc., or to specify a negotiated code as a result of a negotiated rate.
@PrepaidIndicator	0..1	When true, indicates if the rate is a prepaid rate.
@RatePlanName	1	Provides the name of the rate plan or group. Typically used with RatePlanType to further describe the rate plan.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / GuestCounts	0..1	A collection of Guest Counts associated with Room Stay. A child Guest Count element is required for each distinct age group.
@IsPerRoom	0..1	The guests defined in the GuestCounts object apply to each room in the NumberOfRooms for the RoomStay. Value of "false" means that the guests defined in the GuestCounts object apply to all rooms combined in the NumberOfRooms for the RoomStay.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / GuestCounts /	0..n	A recurring element that identifies the number of guests and ages of the guests.

Element @Attribute	Num	Description/Contents
GuestCount		
@AgeQualifyingCode	1	A code representing a business rule that determines the charges for a guest based upon age range (e.g., Adult, Child, Senior, Child With Adult, Child Without Adult). This attribute allows for an increase in rate by occupant class. Refer to OpenTravel Code List Age Qualifying Code (AQC).
@Count	1	The number of guests in one AgeQualifyingCode or Count.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / TimeSpan	1	The Time Span which covers the Room Stay.
@End	1	The ending value of the time span.
@Start	1	The starting value of the time span.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / Guarantee	1	An enumerated type defining the guarantee to be applied to this reservation.
@GuaranteeCode	1	Guarantee Code.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / Guarantee / GuaranteesAccepted / PaymentCard	0..1	Identification about a specific credit card.
@MaskedCardNumber	1	May be used to send a concealed credit card number (e.g., xxxxxxxxxxxx9922).
@CardCode	0..1	The two character code of the credit card issuer.
@ExpireDate	0..1	Indicates the ending date.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / Total	1	The total amount charged for the Room Stay including additional occupant amounts and fees. If TaxInclusive is set to True, then taxes are included in the total amount.
@AmountBeforeTax	0..1	The total amount not including any associated tax (e.g., sales tax, VAT, GST or any associated tax).
@AmountAfterTax	0..1	The total amount including all associated taxes (e.g., sales tax, VAT, GST or any associated tax).
@CurrencyCode	0..1	The code specifying a monetary unit. Use ISO 4217, three alpha code.
@DecimalPlaces	1	Indicates the number of decimal places for a particular currency. This is equivalent to the ISO 4217 standard "minor unit". Typically used when the amount provided includes the minor unit of currency without a decimal point (e.g., USD 8500 needs DecimalPlaces="2" to represent \$85).
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / Memberships / Membership	0..n	The Membership object identifies the frequent customer reward program and (optionally) indicates points awarded for stay activity.
@AccountID	1	The account identification number for this particular member in this particular program.
@BonusCode	0..1	The code or name of the bonus program. BonusCode can be used to indicate the level of membership (Gold Club, Platinum member, etc.)
@TravelSector	0..1	Identifies the travel sector. Refer to OpenTravel Code List Travel Sector (TVS).
@ProgramCode	0..1	The code or name of the membership program ('Hertz', 'AAdvantage', etc.).
@PointsEarned	0..1	The total number of points earned through the selected membership.
OTA_ResRetrieveRS / ReservationList /	0..1	Comment details.

Element @Attribute	Num	Description/Contents
HotelReservation / RoomStays / RoomStay / Comments / Comment /		
@GuestViewable	1	When true, the comment may be shown to the consumer. When false, the comment may not be shown to the consumer.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay / Comments / Comment / Text	1	Textual information regarding the room stay.
@Language	0..1	Language identification.
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay /SpecialRequests / SpecialRequest /	0..1	The SpecialRequest object indicates special requests for a particular guest, service or reservation. Each of these may be independent of any that are tied to the profile (see Profile Synchronization standard).
@RequestCode	0..1	A special request for this reservation and is typically hotel-specific.
@Name	1	
OTA_ResRetrieveRS / ReservationList / HotelReservation / RoomStays / RoomStay /SpecialRequests / SpecialRequest / Text	0..1	
@Language	0..1	Language identification.
OTA_ResRetrieveRS / ReservationList / HotelReservation / Services / Service	0..1	A Service object represents a non-room product provided to guests. Service products may have associated inventory and charges.
@RatePlanCode	0..1	The representation of the rate plan under which this service was booked.
@ReservationStatusType	0..1	An enumerated type that defines the status of the reservation for this service.
@Inclusive	0..1	Whether the price for this service is included in the room rate.
@ServiceInventoryCode	0..1	The representation of the specific service being reserved.
@Quantity	0..1	The number of tickets, rounds of golf, etc. Also serves as the number of persons when pricing class is per person or per person per night.
@ServicePricingType	0..1	An enumerated type that defines how a service is priced. Values: Per stay, Per person, Per night, Per person per night, Per use.
OTA_ResRetrieveRS / ReservationList / HotelReservation / Services / Service / Price	0..1	The selling price of this service.
@ExpireDate	0..1	Indicates the ending date.
@EffectiveDate	0..1	Indicates the starting date.
OTA_ResRetrieveRS / ReservationList / HotelReservation / Services / Service / Price / Total	0..1	The total amount charged for the service including additional amounts and fees.
@AmountBeforeTax	0..1	The total amount not including any associated tax (e.g., sales tax, VAT, GST or any associated tax).
@AmountAfterTax	0..1	The total amount including all associated taxes (e.g., sales tax, VAT, GST or any associated tax).
@CurrencyCode	0..1	The code specifying a monetary unit. Use ISO 4217, three alpha code.
@DecimalPlaces	1	Indicates the number of decimal places for a particular currency. This is equivalent to the ISO 4217 standard "minor unit". Typically used when the amount provided includes the minor unit of currency without a decimal point

Element @Attribute	Num	Description/Contents
		(e.g., USD 8500 needs DecimalPlaces="2" to represent \$85).
OTA_ResRetrieveRS / ReservationList / HotelReservation / Services / Service / Price / Total / Taxes	0..1	A collection of taxes.
@CurrencyCode	0..1	The code specifying a monetary unit. Use ISO 4217, three alpha code.
@DecimalPlaces	1	Indicates the number of decimal places for a particular currency. This is equivalent to the ISO 4217 standard "minor unit". Typically used when the amount provided includes the minor unit of currency without a decimal point (e.g., USD 8500 needs DecimalPlaces="2" to represent \$85).
@Amount	1	
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / UniqueID	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	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.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / UniqueID	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	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.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer	1	Contains basic data on the customer's identity, location, relationships, finances, memberships, etc.
@BirthDate	0..1	Identifies the birth date of the customer.
@Gender	0..1	The gender of the document holder.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / PersonName / GivenName	1	Given name, first name or names.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / PersonName / MiddleName	0..1	The middle name of the person name.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer /	1	Family name, last name.

Element @Attribute	Num	Description/Contents
PersonName / Surname		
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / PersonName / NameSuffix	0..1	Hold various name suffixes and letters (e.g., Jr., Sr., III, Ret., Esq.).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / PersonName / NamePrefix	0..1	Salutation of honorific (e.g., Mr. Mrs., Ms., Miss, Dr.).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Telephone	0..1	Detailed telephone information.
@PhoneNumber	1	Information about a telephone number, including the actual number and its usage.
@CountryAccessCode	0..1	Code assigned by telecommunications authorities for international country access identifier.
@AreaCityCode	0..1	Code assigned for telephones in a specific region, city, or area.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Email	0..5	Electronic email addresses, in IETF specified format.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Address	0..99	Detailed information on an address for the customer.
@Type	0..1	Defines the type of address (e.g., home, business, other). Refer to OpenTravel Code List Communication Location Type (CLT).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Address / AddressLine	0..5	When the address is unformatted (FormattedInd="false") these lines will contain free form address details. When the address is formatted and street number and street name must be sent independently, the street number will be sent using StreetNmbr, and the street name will be sent in the first AddressLine occurrence.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Address / CityName	0..1	City (e.g., Dublin), town, or postal station (i.e., a postal service territory, often used in a military address).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Address / PostalCode	0..1	Post Office Code number.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Address / StateProv	0..1	State or Province name (e.g., Texas).

Element @Attribute	Num	Description/Contents
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Address / CountryName	0..1	Country name (e.g., Ireland).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Address / CompanyName	0..1	Identifies a company by name.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / CitizenCountryName	0..3	Name of the (self-professed) country that is claimed for citizenship.
@DefaultInd		When true, indicates a default value should be used.
@Code	0..1	A two character country code as defined in ISO3166.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / Document	0..1	Provides information on a specific document.
@DocIssueCountry	0..1	Country where the document was issued.
@DocHolderNationality	0..1	The country code of the nationality of the document holder.
@ExpireDate	0..1	Indicates the ending date.
@EffectiveDate	0..1	Indicates the starting date.
@BirthDate	0..1	Indicates the date of birth as indicated in the document, in ISO 8601 prescribed format.
@BirthPlace	0..1	Specifies the birth place of the document holder (e.g., city, state, county, province), when designating a country of birth, use BirthCountry.
@Gender	0..1	Identifies the gender.
@DocID	0..1	Unique number assigned by authorities to document.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / CustLoyalty	0..1	
@AllianceLoyaltyLevelName	0..1	Name of the alliance loyalty level (e.g., OneWorld uses Emerald, Ruby, etc. and SkyTeam uses Elite, etc.).
@ExpireDate	0..1	Indicates the ending date.
@EffectiveDate	0..1	Indicates the starting date.
@MembershipID	1	Unique identifier of the member in the program (membership number, account number, etc.).

Element @Attribute	Num	Description/Contents
@ProgramID	1	Identifier to indicate the company owner of the loyalty program.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / CompanyName	0..1	Identifies a company by name.
@CodeContext	0..1	Identifies the context of the identifying code, such as DUNS, IATA or internal code, etc.
@CompanyShortName	0..1	Used to provide the company common name.
@Code	0..1	Identifies a company by the company code.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / AddressInfo	0..1	Detailed information on an address for the company.
@Type	0..1	Defines the type of address (e.g., home, business, other). Refer to OpenTravel Code List Communication Location Type (CLT).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / AddressInfo	0..1	Detailed information on an address for the company.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / AddressInfo / AddressLine	0..1	When the address is unformatted (FormattedInd="false") these lines will contain free form address details. When the address is formatted and street number and street name must be sent independently, the street number will be sent using StreetNmbr, and the street name will be sent in the first AddressLine occurrence.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / AddressInfo / CityName	0..1	City (e.g., Dublin), town, or postal station (i.e., a postal service territory, often used in a military address).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / AddressInfo / PostalCode	0..1	Post Office Code number.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / AddressInfo / StateProv	0..1	State or Province name (e.g., Texas).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / AddressInfo / CountryName	0..1	The name or code of a country (e.g., as used in an address or to specify citizenship of a traveler).
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests /	0..1	Information on a telephone number for the company.

Element @Attribute	Num	Description/Contents
ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / TelephoneInfo		
@PhoneNumber	1	Telephone number assigned to a single location.
@CountryAccessCode	0..1	Code assigned by telecommunications authorities for international country access identifier.
@AreaCityCode	0..1	Code assigned for telephones in a specific region, city, or area.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / CompanyInfo / Email	0..1	Information on an email address for the company.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGlobalInfo / HotelReservationIDs	1	ResGlobalInfo is a container for various information that affects the Reservation as a whole. These include global comments, counts, reservation IDs, loyalty programs, and payment methods.
OTA_ResRetrieveRS / ReservationList / HotelReservation / ResGlobalInfo / HotelReservationIDs / HotelReservationID	1	The HotelReservationID object contains various unique (ReservationID) and non unique (ConfirmationID, CancellationID) identifiers that the trading partners associate with a given reservation.
@ResID_Type	0..1	Defines the type of Reservation ID (e.g., reservation number, cancellation number). Refer to OpenTravel Code List Unique ID Type (UIT).
@ResID_Value	1	This is the actual value associated with ResID_Type as generated by the system that is the source of the ResID_Type
@ResID_Source	0..1	A unique identifier to indicate the source system which generated the ResID_Value.

5.4.5 Sample Response Message

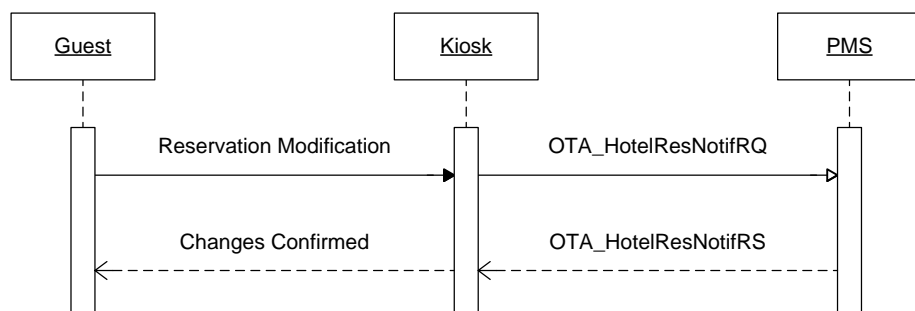
```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_ResRetrieveRS EchoToken="6cfalea3-df0f-497f-9aaf-e927fc11affc" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Success/>
  <ReservationsList>
    <HotelReservation>
      <UniqueID Type="14" ID="RES123456"/>
      <RoomStays>
        <RoomStay RoomStayStatus="Reserved">
          <RoomTypes>
            <RoomType NonSmoking="true" RoomTypeCode="KING" RoomID="1706">
              <RoomDescription>
                <Text Language="en-us">King Room</Text>
              </RoomDescription>
            </RoomType>
          </RoomTypes>
          <RatePlans>
            <RatePlan PriceViewableInd="true" RatePlanCode="RAT123"
PrepaidIndicator="false" RatePlanName="Standard Rate"/>
          </RatePlans>
          <GuestCounts>
            <GuestCount AgeQualifyingCode="10" Count="1"/>
          </GuestCounts>
          <TimeSpan End="2010-08-15" Start="2010-08-13"/>
          <Guarantee GuaranteeCode="C">
            <GuaranteesAccepted>
              <GuaranteeAccepted>

```

```
CardCode="VI" ExpireDate="0113"/>
    <PaymentCard MaskedCardNumber="XXXXXXXXXXXX1111"
    </GuaranteeAccepted>
    </GuaranteesAccepted>
    </Guarantee>
    <Total AmountBeforeTax="0"/>
    <Memberships>
        <Membership AccountID="MMB123456" ProgramCode="FRQTRV"/>
    </Memberships>
    </RoomStay>
</RoomStays>
<ResGuests>
    <ResGuest>
        <Profiles>
            <ProfileInfo>
                <UniqueID Type="1" ID="PRF1234456"/>
                <Profile>
                    <Customer BirthDate="1967-08-13" Gender="Male">
                        <PersonName>
                            <NamePrefix>Mr.</NamePrefix>
                            <GivenName>John</GivenName>
                            <MiddleName>A</MiddleName>
                            <Surname>Smith</Surname>
                        </PersonName>
                        <Telephone PhoneNumber="555-1212"
                        <Email>john.smith@gmail.com</Email>
                        <Address Type="2">
                            <AddressLine>101 Main
                            <CityName>Anytown</CityName>
                            <PostalCode>PA</PostalCode>
                            <StateProv>01234</StateProv>
                            <CountryName>US</CountryName>
                        </Address>
                        <Document DocIssueCountry="US"
ExpireDate="2013-08-31" EffectiveDate="2003-09-01" BirthDate="1967-08-13" BirthPlace="NY, NY" Gender="Male"
DocID=" 6746783-463217643-64326"/>
                        <CustLoyalty ExpireDate="2010-08-31"
MembershipID="MMB123456" ProgramID="FRQTRV"/>
                    </Customer>
                </Profile>
            </ProfileInfo>
        </Profiles>
    </ResGuest>
</ResGuests>
<ResGlobalInfo>
    <HotelReservationIDs>
        <HotelReservationID ResID_Type="14" ResID_Value="RES123456"/>
    </HotelReservationIDs>
</ResGlobalInfo>
</HotelReservation>
</ReservationsList>
</OTA_ResRetrieveRS>
```


5.5 Modify Booking

5.5.1 Messaging Use Case



Use Case Name:	Modify Booking
Summary:	The guest changes some elements of his booking and saves the changes: name, change departure date, request alternate room and additional services, select pay method and attached profile.
Basic Course of Events:	<p>The Use Case Begins when the Guest elects to Change elements of the booking</p> <ol style="list-style-type: none"> 1) The Guest navigates through the Kiosk User Interface to change various parts of his Booking <ul style="list-style-type: none"> - Name - select pay method 2) Based on the Booking Rules the PMS may prevent the Kiosk from changing some parts of the booking 3) When the Guest has made all changes to the booking the User accepts the changes 4) The Kiosk sends all changes to the PMS 5) The PMS commits the Changes <p>The Use Case Ends when the Guest has made the changes he needs to finalize its booking.</p>
Exception Path:	<p>In Step 5, The PMS detects that the bookings have been updated since the Kiosk retrieved the information.</p> <ol style="list-style-type: none"> 1. The PMS sends an error message to the Kiosk 2. The Kiosk retrieves the booking again and updates with the user selection and re submits, or asks the Guest to remake the changes. <p>In Step 5, the PMS cannot confirm the changes requested by the Guest:</p> <ol style="list-style-type: none"> 3. The PMS sends an error message to the Kiosk 4. The Kiosk retrieves the booking again and updates with the user selection and re submits, or asks the Guest to remake the changes
Trigger:	The guest elects to change his booking.
Assumptions:	The Kiosk has retrieved the full booking.
Preconditions:	The Booking has not been updated.
Postconditions:	The Booking has been updated.

5.5.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRQ	1	Hotel Reservation Notif Request supports the functionality of updating other systems with reservation data. The message assumes a push model, with the originating system pushing the data to another system. The originating system would usually be a booking source, such as a Global Distribution System (GDS), a Central Reservation System (CRS) or some other agent of the hotel.

Element @Attribute	Num	Description/Contents
@EchoToken	0..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.
@ResStatus	1	To specify the type of action requested when more than one function could be handled by the message. The value should be 'Modify' .
@TimeStamp	0..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 OTA versioned messages, the version of the message is indicated by a decimal value.
@Target	0..1	Used to indicate whether the request is for the Test or Production system. Value can be either 'Test' or 'Production'.
OTA_HotelResNotifRQ / POS / Source	1	Provides information on the source of a request.
@TerminalID	0..1	This is the electronic address of the device from which information is entered.
OTA_HotelResNotifRQ / POS / Source / RequestorID	0..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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA Code List Unique ID Type (UIT).
@ID		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).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation	1	The Reservation class contains the current reservation being created or altered.
@RoomStayReservation	0..1	Boolean True if this reservation is reserving rooms and false if it is only reserving services.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / UniqueID		The booking reference for the reservation.
@Type	1	A reference to the type of object defined by the UniqueID element. Refer to OTA Code List Unique ID Type (UIT).
@ID		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).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / RoomStays / RoomStay / Guarantee / GuaranteesAccepted / GuaranteeAccepted / PaymentCard	0..1	Payment Card information.
@CardNumber	0..1	Credit card number embossed on the card.
@CardType	0..1	Indicates the type of magnetic striped card. Refer to OTA Code ListCard Type (CDT).
@CardCode	0..1	The 2 character code of the credit card issuer.
@ExpiryDate	0..1	Indicates the ending date.

Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / RoomStays / RoomStay / Guarantee / GuaranteesAccepted / GuaranteeAccepted / PaymentCard / CardHolderName	0..1	The name embossed on the card.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo	0..1	Root element for profile content.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / UniqueID	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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA Code List Unique ID Type (UIT).
@ID	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).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile	1	Root element for profile content.
@ProfileType	0..n	Code to specify a profile such as Customer, Tour Operator, Corporation, etc. Refer to OTA Code List Profile Type (PRT).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Customer / PersonName / NamePrefix	0..1	Salutation of honorific (e.g., Mr. Mrs. Ms. Miss, Dr.).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Customer / PersonName / GivenName	0..1	Given name, first name or names.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Customer / PersonName / MiddleName	0..1	The middle name of the person name.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Customer / PersonName / SurnamePrefix	0..1	e.g., "van der", "von", "de".
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Customer / PersonName / Surname	0..1	Family name, last name.

Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Customer / PersonName / NameSuffix	0..1	Hold various name suffixes and letters (e.g., Jr., Sr. III, Ret., Esq.).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Customer / PersonName / NameTitle	0..1	Degree or honors (e.g., Ph.D., M.D.)

5.5.3 Modifying the Card on File

Sample Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRQ EchoToken="5a7ae570-93fe-46b1-abd5-3ca882982a02" ResStatus="Modify" TimeStamp="2010-02-12T12:26:47" Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="TRM123">
      <RequestorID Type="18" ID="KSKSYS123"/>
    </Source>
  </POS>
  <HotelReservations>
    <HotelReservation>
      <UniqueID Type="14" ID="RES123456"/>
      <RoomStays>
        <RoomStay>
          <Guarantee>
            <GuaranteesAccepted>
              <GuaranteeAccepted>
                <PaymentCard CardNumber="4444333322221111"
CardCode="VI" ExpireDate="0113">
                  <CardHolderName>a</CardHolderName>
                </PaymentCard>
              </GuaranteeAccepted>
            </GuaranteesAccepted>
          </Guarantee>
        </RoomStay>
      </RoomStays>
    </HotelReservation>
  </HotelReservations>
</OTA_HotelResNotifRQ>
```

Sample Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRS EchoToken="5a7ae570-93fe-46b1-abd5-3ca882982a02" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Success/>
</OTA_HotelResNotifRS>
```

5.5.4 Assigning a Guest Room

Sample Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRQ EchoToken="94124645-94e5-4605-9256-9b833a534ef7" ResStatus="Modify" TimeStamp="2010-02-12T12:26:47" Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

```
<POS>
  <Source TerminalID="a">
    <RequestorID Type="18" ID="KSKSYS123"/>
  </Source>
</POS>
<HotelReservations>
  <HotelReservation>
    <UniqueID Type="14" ID="RES123456"/>
    <RoomStays>
      <RoomStay>
        <RoomType RoomID="1701"/>
      </RoomStay>
    </RoomStays>
  </HotelReservation>
</HotelReservations>
</OTA_HotelResNotifRQ>
```

Sample Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRS EchoToken="94124645-94e5-4605-9256-9b833a534ef7" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Success/>
</OTA_HotelResNotifRS>
```

5.5.5 Update Guest Name

Sample Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRQ EchoToken="b349af06-fb40-491e-86fc-657c0477d6f8" ResStatus="Modify" TimeStamp="2010-02-
12T12:26:47" Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="TRM123">
      <RequestorID Type="18" ID="KSKSYS123"/>
    </Source>
  </POS>
  <HotelReservations>
    <HotelReservation>
      <UniqueID Type="14" ID="RES123456"/>
      <ResGuests>
        <ResGuest Action="Add-Update">
          <Profiles>
            <ProfileInfo>
              <UniqueID Type="1" ID="PRF123456"/>
              <Profile>
                <Customer>
                  <PersonName>
                    <GivenName>John</GivenName>
                    <MiddleName>A</MiddleName>
                    <Surname>Smith</Surname>
                  </PersonName>
                </Customer>
              </Profile>
            </ProfileInfo>
          </Profiles>
        </ResGuest>
      </ResGuests>
    </HotelReservation>
  </HotelReservations>
</OTA_HotelResNotifRQ>
```

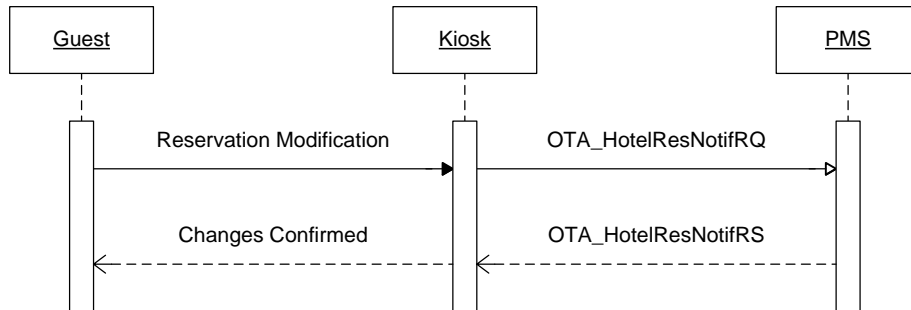
Sample Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<OTA_HotelResNotifRS EchoToken="b349af06-fb40-491e-86fc-657c0477d6f8" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Success/>
</OTA_HotelResNotifRS>
```

5.6 Create Sharewith

5.6.1 Messaging Use Case

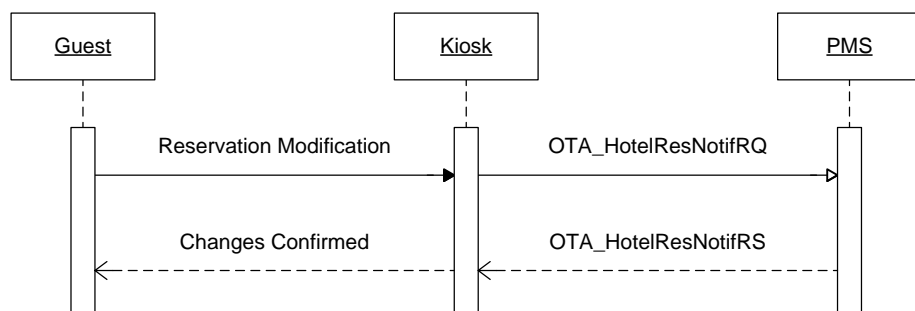


This use case is based upon the 2010A Product Distribution Workgroup specification. Please visit <http://collaboration.htng.org/specs/> for more information.

Use Case Name:	Create Sharewith
Summary:	Guest has checked in their reservation and wishes to allow a traveling companion to be able to have room charging privileges (and possibly split the room rate as well.)
Basic Course of Events:	See use case diagram.
Exception Path:	None.
Alternative Paths:	None.
Trigger:	Guest selects option on Kiosk screen to define additional travelers.
Assumptions:	None.
Preconditions:	None.
Postconditions:	None.
Business Rules:	None.

5.7 Define Multiple Names

5.7.1 Messaging Use Case



Use Case Name:	Define Multiple Names
Summary:	During the check in process, the guest can add to their reservation the names of accompanying guests, for the purpose of identifying them to the hotel staff for the purpose of receiving hotel services.
Basic Course of Events:	<p>The use case begins after an in-house or arriving guest has found their booking (See Find Booking)</p> <ol style="list-style-type: none"> 1. The guest selects an option to add an accompanying guest to the reservation 2. The kiosk collects a guest first name and last name 3. The kiosk performs a reservation modification to add the guest name to the reservation <p>The use case ends after the reservation has been modified.</p>
Exception Paths:	<p>In Step 3, the kiosk may not be permitted to add any more accompanying guests due to maximum occupancy or other business rules.</p> <ol style="list-style-type: none"> 1. The PMS sends an error message to the kiosk 2. The kiosk indicates to the guest that no more names can be added
Alternative Paths:	<p>Dependent on kiosk implementation, step 1 or 2 could be preceded by kiosk presenting a list of current names on the reservation.</p> <ol style="list-style-type: none"> 1. Kiosk presents a list of current accompanying guests on the reservation 2. Guest can opt to add a new name or abandon the add process
Trigger:	Guest indicates a desire to add an additional guest name to the existing reservation by selecting the appropriate option on the Kiosk screen.
Assumptions:	None.
Preconditions:	An accompanying guest is not listed on an existing reservation.
Postconditions:	The accompanying guest's name appears as such on the reservation.

5.7.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRQ	1	Hotel Reservation Notif Request supports the functionality of updating other systems with reservation data. The message assumes a push model, with the originating system pushing the data to another system. The originating system would usually be a booking source, such as a Global Distribution System (GDS), a Central Reservation System (CRS) or some other agent of the hotel.
@EchoToken	0..1	A reference for additional message identification, assigned by the requesting host system. When a request message includes an echo, the corresponding response message MUST include an echo token with an identical value.
@ResStatus	1	To specify the type of action requested when more than one function could be handled by the message. The value should be ' Modify '.
@TimeStamp	0..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

Element @Attribute	Num	Description/Contents
		the 24-hour clock (e.g., 20 November 2003, 1:59:38 pm UTC becomes 2003-11-20T13:59:38Z).
@Version	1	For all OTA versioned messages, the version of the message is indicated by a decimal value.
@Target	0..1	Used to indicate whether the request is for the Test or Production system. Value can be either 'Test' or 'Production'.
OTA_HotelResNotifRQ / POS / Source	1	Provides information on the source of a request.
@TerminalID	0..1	This is the electronic address of the device from which information is entered.
OTA_HotelResNotifRQ / POS / Source / RequestorID	0..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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation	1	The Reservation class contains the current reservation being created or altered.
@ResStatus	1	The status of the reservation. Should be "Modify" when defining multiple names.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest	1	The ResGuest object contains the information about a guest associated with a reservation.
@Action	1	The status of the ResGuest. Should be "Add-Update" when defining multiple names.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / RelatedTraveler	0..99	Identifies a traveler associated with the customer.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / RelatedTraveler / PersonName / GivenName	1	Given name, first name or names.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / ResGuests / ResGuest / Profiles / ProfileInfo / Profile / Customer / RelatedTraveler / PersonName / SurName	1	Family name, last name.

5.7.3 Sample Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRQ EchoToken="a" ResStatus="Modify" TimeStamp="2010-12-17T09:30:47Z" Version="0.0"
Target="Test" xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <POS>
    <Source TerminalID="Kiosk123">
      <RequestorID Type="0" ID_Context="Kiosk" ID="123"/>
    </Source>
  </POS>
</OTA_HotelResNotifRQ>
```



```

        </Source>
    </POS>
    <UniqueID Type="14" ID_Context="PMS123" ID="RES123" />
    <HotelReservations>
        <HotelReservation ResStatus="Modify">
            <ResGuests>
                <ResGuest Action="Add-Update">
                    <Profiles>
                        <ProfileInfo>
                            <Profile>
                                <Customer>
                                    <RelatedTraveler>
                                        <PersonName>
                                            <GivenName>a</GivenName>
                                            <Surname>a</Surname>
                                        </PersonName>
                                    </RelatedTraveler>
                                </Customer>
                            </Profile>
                        </ProfileInfo>
                    </Profiles>
                </ResGuest>
            </ResGuests>
        </HotelReservation>
    </HotelReservations>
</OTA_HotelResNotifRQ>

```

5.7.4 Data Element Table – Response

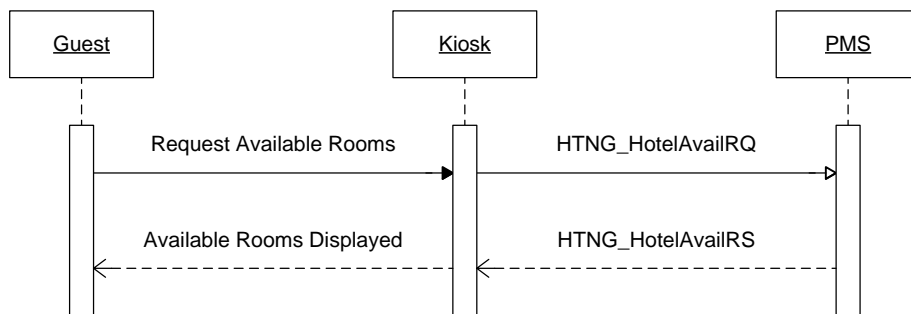
Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRS	1	This message returns a list of reservations when an exact match on a read request could not be made or the request was to return a list of reservations meeting specified criteria.
@EchoToken	0..1	A reference for additional message identification, 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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
OTA_HotelResNotifRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
OTA_HotelResNotifRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
OTA_HotelResNotifRS / 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).

5.7.5 Sample Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRS EchoToken="f4a2d799-0824-4801-9264-1f8554f8d427" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Success/>
</OTA_HotelResNotifRS>
```

5.8 Request Room List

5.8.1 Messaging Use Case



Use Case Name:	Request Room List
Summary:	The kiosk obtains a list of rooms that can be assigned to an arriving reservation (rooms can be clean or dirty), based on guest preferences.
Basic Course of Events:	<p>The use case begins after an arriving guest has found their booking (see Find Booking)</p> <ol style="list-style-type: none"> 1. The guest chooses an option to select a room 2. The kiosk collects the desired room preferences from the guest 3. The kiosk requests the list of available rooms from the PMS which match the criteria specified by the guest, plus an indicator of 'maximum number of responses' desired by the kiosk 4. The PMS returns a list of available rooms which can be assigned; the response can contain both rooms available for check-in (i.e., Vacant/clean) and rooms available for pre-assignment (departing, dirty, etc.) immediately assigned, sorted by status (vacant/clean first) and PMS ranking scheme. The list contains no more than the maximum responses requested by the kiosk.
Exception Path:	<p>In step 4, there may be no rooms (dirty or clean) that can be assigned to the reservation.</p> <ol style="list-style-type: none"> 1. The PMS sends an error message to the kiosk 2. The kiosk indicates to the guest that no rooms are available and follows its business logic for this exception (usually guest is sent to front desk)
Alternative Paths:	Dependent on kiosk implementation, Steps 2 and (3, 4) could be reversed, in that kiosk first requests rooms for assignment, then presents the guest with preference selections that match only the available rooms presented by the PMS.
Trigger:	Guest opts to select a room prior to checking in.
Assumptions:	The PMS will return all available rooms prioritized by the PMS ranking scheme.
Preconditions:	Guest has an arriving reservation which may or may not have a room assigned.
Postconditions:	No change.

The response data utilizes the AvailableIndicator flag to indicate which rooms can be checked into immediately (@AvailableIndicator="true") and which can only be pre-assigned (@AvailableIndicator="false"). The AvailableIndicator="true" bucket will be filled with records up to the @MaxResponses value in the request, then the AvailableIndicator="false" bucket will be populated. The sum of the elements across both buckets must not exceed @MaxResponses. Within each bucket, the rooms may be sorted by PMS ranking scheme.

5.8.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
OTA_HotelAvailRQ	1	Requests availability of hotel properties by specific criteria that may include: dates, date ranges, price range, room types, regular and qualifying rates, and/or services and amenities. The availability message can be used to get an initial availability or to get availability for the purpose of modifying an existing reservation.
@EchoToken	0..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.
@HotelStayOnly	1	A boolean flag that indicates the response message should return only HotelStay information. If true, no RoomStay information will be returned.
@ExactMatchOnly	1	A boolean flag that indicates the response message should send back only those rates that are an exact match to the requested criteria.
@TimeStamp	0..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 OTA versioned messages, the version of the message is indicated by a decimal value.
@Target	0..1	Used to indicate whether the request is for the Test or Production system. Value can be either 'Test' or 'Production'.
@MaxResponses	1	A positive integer value that indicates the maximum number of responses desired in the return.
OTA_HotelAvailRQ / POS / Source	1	Provides information on the source of a request.
@TerminalID	0..1	The electronic address of the device from which information is entered.
OTA_HotelAvailRQ / POS / Source / RequestorID	0..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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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).
OTA_HotelAvailRQ / AvailRequestSegments / AvailRequestSegment	1	A collection of criteria that requests a bookable entity (in this case, as listing of physical rooms available for occupancy for a given date range).
@AvailReqType	1	An enumerated type that indicates whether the availability request is for a room or non-room product, such as amenities or other activities (e.g., banquet, conference room, etc.). Values: Room, NonRoom, Both. For the purposes of this use case, the value must be set to "Room".
OTA_HotelAvailRQ / AvailRequestSegments / AvailRequestSegment / HotelSearchCriteria / Criterion	1	Child elements that identify a single search criterion by criteria type. Because many of the types include partial matches to string values such as partial addresses (street names without a number) or partial telephone numbers (area code or three-digit prefix area, etc.) an ExactMatch attribute indicates whether the match to the string value must be exact.
OTA_HotelAvailRQ / AvailRequestSegments /	1	Range of dates or fixed set of dates for Availability Request. Date range can also be specified by using start dates and number of nights (minimum,

Element @Attribute	Num	Description/Contents
AvailRequestSegment / HotelSearchCriteria / Criterion / StayDateRange		maximum or fixed).
@End	1	The departure date for the reservation.
@Start	1	The arrival date for the reservation.
OTA_HotelAvailRQ / AvailRequestSegments / AvailRequestSegment / HotelSearchCriteria / Criterion / RoomStayCandidate / RoomStayCandidate	1	Element used to identify available room products.
@NonSmoking	0..1	Non-smoking indicator.
@RoomTypeCode	0..1	Specific system room type code, i.e., A1K, A1Q etc.
OTA_HotelAvailRQ / AvailRequestSegments / AvailRequestSegment / HotelSearchCriteria / Criterion / RoomStayCandidate / RoomStayCandidate / GuestCounts	1	A collection of GuestCount by age group.
OTA_HotelAvailRQ / AvailRequestSegments / AvailRequestSegment / HotelSearchCriteria / Criterion / RoomStayCandidate / RoomStayCandidate / GuestCounts / GuestCount	1..n	A recurring element that identifies the number of guests and ages of the guests.
@AgeQualifyingCode	1	A code representing a business rule that determines the charges for a guest based upon age range (e.g., Adult, Child, Senior, Child With Adult, Child Without Adult). This attribute allows for an increase in rate by occupant class. Refer to OpenTravel Code List Age Qualifying Code (AQC).
@Count	1	The number of guests in one AgeQualifyingCode or Count.
OTA_HotelAvailRQ / AvailRequestSegments / AvailRequestSegment / HotelSearchCriteria / Criterion / RoomStayCandidate / RoomStayCandidate / RoomAmenity	0..5	Used to pass room attributes. Typical attributes are smoking, non-smoking, wheelchair access, room location (low floor, high floor, close to elevator, etc) and view (garden, pool, ocean, etc).
@RoomAmenity	1	Identifies the types of room amenities offered by the property. Refer to OpenTravel Code List Room Amenity Type (RMA).
OTA_HotelAvailRQ / HotelReservationIDs	1	A Collection of HotelReservationID objects for a given reservation. The collection of all ReservationIDs can include Passenger Name Record (PNR), Guest Name Record (GNR) and Guest Folio numbers. Associated with each can be a Confirmation number which is usually given to the guest.
OTA_HotelAvailRQ / HotelReservationIDs / HotelReservationID	0..1	The HotelReservationID object contains various unique (ReservationID) and non-unique (ConfirmationID, CancellationID) identifiers that the trading partners associate with a given reservation.
@ResID_Type	1	Defines the type of Reservation ID (e.g., reservation number, cancellation number). Refer to OpenTravel Code List Unique ID Type (UIT).
@ResID_Value	1	This is the actual value associated with ResID_Type as generated by the system that is the source of the ResID_Type.
@ResID_Source	0..1	A unique identifier to indicate the source system which generated the ResID_Value.

5.8.3 Sample Message – Request

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelAvailRQ EchoToken="a" HotelStayOnly="true" ExactMatchOnly="true" TimeStamp="2010-08-13T09:30:47Z"
Version="0.0" Target="Test" MaxResponses="2" xmlns="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="Kiosk123">
      <RequestorID Type="0" ID_Context="Kiosk" ID="123"/>
    </Source>
  </POS>
  <AvailRequestSegments>
    <AvailRequestSegment AvailReqType="Room">
      <HotelSearchCriteria>
        <Criterion>
          <StayDateRange End="2010-08-13" Start="2010-08-13"/>
          <RoomStayCandidates>
            <RoomStayCandidate NonSmoking="true" RoomTypeCode="A1K">
              <GuestCounts>
                <GuestCount AgeQualifyingCode="10" Count="1"/>
              </GuestCounts>
              <RoomAmenity RoomAmenity="0"></RoomAmenity>
            </RoomStayCandidate>
          </RoomStayCandidates>
        </Criterion>
      </HotelSearchCriteria>
    </AvailRequestSegment>
  </AvailRequestSegments>
  <HotelReservationIDs>
    <HotelReservationID ResID_Type="14" ResID_Value="PMSRES123" ResID_Source="a"/>
  </HotelReservationIDs>
</OTA_HotelAvailRQ>
```

5.8.4 Data Element Table – Response

Element @Attribute	Num	Description/Contents
OTA_HotelAvailRS	1	Returns information about hotel properties that meet the requested criteria, indicating whether the requested service, rate, room stay, etc. is available within the date(s) specified. The response message may include Warnings from business processing rules or Errors if the request did not succeed.
@EchoToken	0..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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
OTA_ HotelAvailRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
OTA_ HotelAvailRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
OTA_ HotelAvailRS / 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.

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).
OTA_HotelAvailRS / HotelStays	1	A collection of HotelStay objects.
OTA_HotelAvailRS / HotelStays / HotelStay	1..n	A quick view of the requested hotels' general availability for each day in the requested range.
OTA_HotelAvailRS / HotelStays / HotelStay / Availability	1	Details on the hotel stay including the type of availability and the date range for which it applies.
@End	1	The reservation departure date.
@Start	1	The reservation arrival date.
OTA_HotelAvailRS / RoomStays	0..1	A collection of RoomStay objects.
OTA_HotelAvailRS / RoomStays / RoomStay	1..n	Details on the Room Stay including Guest Counts, Time Span of this Room Stay, and financial information related to the Room Stay, including Guarantee, Deposit and Payment and Cancellation Penalties.
@AvailabilityIndicator	1	May be used as a simple true/false to indicate availability or used in conjunction with @AvailabilityStatus to indicate if restricted rate qualifications have been met, e.g., Exclusive and Available (true) vs. Exclusive but not Available (false). For the purposes of this message, this flag indicates whether the room is available for immediate occupancy (e.g., the room is vacant and clean.)
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes	1	A collection of RoomType objects.
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType	1..n	Element used to identify available room products.
@RoomCategory	0..1	Indicates the category of the room. Typical values would be Moderate, Standard, or Deluxe. Refer to OpenTravel Code List Segment Category Code (SEG).
@NonSmoking	0..1	Non-smoking indicator.
@RoomTypeCode	1	Specific system room type code, i.e., A1K, A1Q etc.
@Composite	0..1	Indicates that the room (suite) is a composite of smaller units.
@RoomID	1	A string value representing the unique identification of a room if the request is looking for a specific room.
@RoomType	0..1	(Formerly, RoomInventoryCode) A code value that indicates the type of room for which this request is made, e.g., double, king, etc. Values may use the Hotel Descriptive Content table or a codes specific to the property or hotel brand.
@Floor	0..1	Floor on which the room is located.
@RoomViewCode	0..1	Indicates the view of the room. Typical values would be "Ocean view", "Pool view" or "Garden View". Refer to OpenTravel Code List Room View Type (RVT).
@BedTypeCode	0..1	Indicates the type of bed(s) found in the room. Typical values would be Double, Twin, Queen, or King. Refer to OpenTravel Code List Bed Type Code (BED).
@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).
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / RoomDescription	0..1	Textual information regarding the room.
@Name	1	For the purposed of this message, this should be set to "Short Description".
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / RoomDescription / Text	1	A short description of the room.

Element @Attribute	Num	Description/Contents
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / AdditionalDetails	0..1	A collection of AdditionalDetails objects.
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / AdditionalDetails / AdditionalDetail	1..n	Used to send additional information.
@Type	1	Used to define the type of information being sent (e.g., rate description, property description, room information). Refer to OpenTravel Code List Additional Detail Type (ADT).
@Code	1	Trading partner code associated to AdditionalDetailType.
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / AdditionalDetails / AdditionalDetail / DetailDescription	0..1	The base element containing detail information about the description of the room.
@Name	1	For the purposes of this message, this should be set to "Long Description".
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / AdditionalDetails / AdditionalDetail / DetailDescription / Text	1	A detailed description of the room.
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / Amenities	0..1	A collection of Amenity objects.
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / Amenities / Amenity	0..5	This element is used to pass room attributes. Typical attributes are smoking, non-smoking, wheelchair access, room location (low floor, high floor, close to elevator, etc.) and view (garden, pool, ocean, etc.).
@RoomAmenity	1	Identifies the types of room amenities offered by the property. Refer to OpenTravel Code List Room Amenity Type (RMA).
OTA_HotelAvailRS / RoomStays / RoomStay / RoomTypes / RoomType / Occupancy	0..5	Provides parameters of occupancy limits
@MinOccupancy	0..1	Minimum number of persons allowed in a unit of accommodation or place.
@MaxOccupancy	1	Maximum number of persons allowed in a unit of accommodation or place.
@AgeQualifyingCode	0..1	A code representing a business rule that determines the charges for a guest based upon age range (e.g., Adult, Child, Senior, Child With Adult, Child Without Adult). This attribute allows for an increase in rate by occupant class. Refer to OpenTravel Code List Age Qualifying Code (AQC).

5.8.5 Sample Message – Response

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelAvailRS EchoToken="a" TimeStamp="2010-08-13T09:30:47Z" Version="0.0" Target="Test"
xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Success/>
  <Warnings>
    <Warning Type="0" Status="a" ShortText="a" Code="0">String</Warning>
  </Warnings>
  <HotelStays>
    <HotelStay>
      <Availability End="2010-08-14" Start="2010-08-13"/>
    </HotelStay>
  </HotelStays>
  <RoomStays>
    <RoomStay AvailableIndicator="true">
      <RoomTypes>
        <RoomType NonSmoking="true" RoomTypeCode="A1k" RoomID="100" Floor="1">
          <RoomDescription Name="Short Description">
```

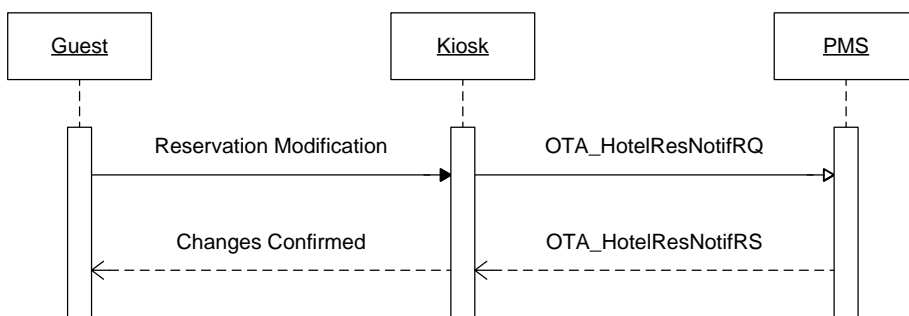
```

        <Text>Standard King</Text>
      </RoomDescription>
      <Occupancy MinOccupancy="1" MaxOccupancy="2" />
    </RoomType>
    <RoomType NonSmoking="true" RoomTypeCode="AlK" RoomID="101" Floor="1">
      <RoomDescription Name="Short Description">
        <Text>Standard King</Text>
      </RoomDescription>
      <Occupancy MinOccupancy="1" MaxOccupancy="2" />
    </RoomType>
    <RoomType NonSmoking="true" RoomTypeCode="AlK" RoomID="115" Floor="1">
      <RoomDescription Name="Short Description">
        <Text>Standard King</Text>
      </RoomDescription>
      <Occupancy MinOccupancy="1" MaxOccupancy="2" />
    </RoomType>
  </RoomTypes>
</RoomStay>
<RoomStay AvailableIndicator="false">
  <RoomTypes>
    <RoomType NonSmoking="true" RoomTypeCode="AlK" RoomID="102" Floor="1">
      <RoomDescription Name="Short Description">
        <Text>Standard King</Text>
      </RoomDescription>
      <Occupancy MinOccupancy="1" MaxOccupancy="2" />
    </RoomType>
    <RoomType NonSmoking="true" RoomTypeCode="AlK" RoomID="219" Floor="2">
      <RoomDescription Name="Short Description">
        <Text>Standard King</Text>
      </RoomDescription>
      <Occupancy MinOccupancy="1" MaxOccupancy="2" />
    </RoomType>
  </RoomTypes>
</RoomStay>
</RoomStays>
</OTA_HotelAvailRS>

```

5.9 Change/Update Room

5.9.1 Messaging Use Case



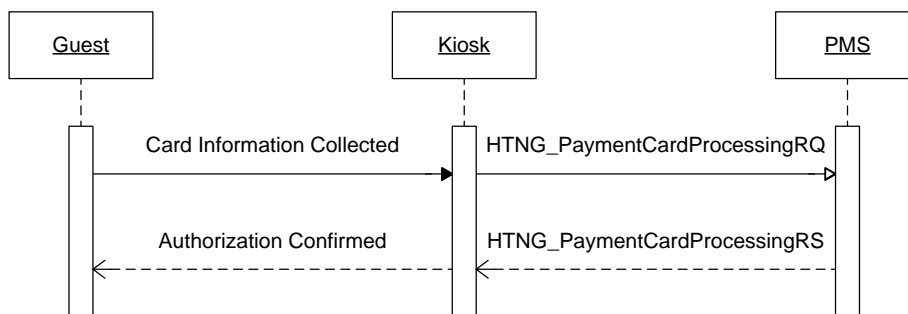
This use case is based upon the 2010A Product Distribution Workgroup specification. Please visit <http://collaboration.htng.org/specs/> for more information.

Use Case Name:	Change/Update Room
Summary:	In the case of a pre-assigned room, allow the guest to pick another room. In the case where the guest has selected a room, allow the guest to change the selection.
Basic Course of Events:	The use case begins after the kiosk has obtained a list of rooms which can be assigned to the reservation and the guest has made their selection (see

	Request Room List). 1. The kiosk performs a reservation modification with the selected room number 2. The PMS updates the reservation The use case ends after the reservation has been modified with the selected room number.
Exception Path:	In step 1, the selected room may have already been assigned to another reservation, or the guest is hard blocked, in which case the PMS sends an error message to the kiosk.
Alternative Paths:	None.
Trigger:	Guest has selected their desired room.
Assumptions:	Additional rooms are available. Assigned room is not blocked. Reservation rates will not change.
Preconditions:	Guest has an arriving reservation with or without a room assigned.
Postconditions:	Guest has an arriving reservation with a room assigned.

5.10 Authorize Card

5.10.1 Messaging Use Case



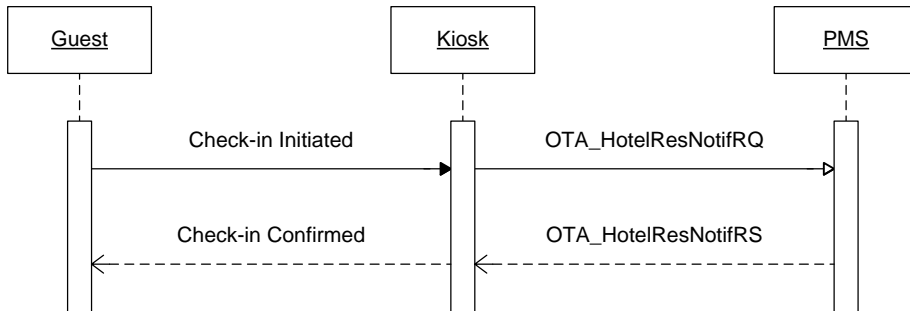
This use case is based upon the 2010A Payment Systems & Data Security, Payment Processing Specification. Please visit <http://collaboration.htng.org/specs/> for more information.

Use Case Name:	Authorize Card - PMS
Summary:	After John has confirmed his booking and preferences, he is required to insert his credit card in the kiosk. A payment transaction then happens and allows the kiosk to move on to finalize the check-in.
Basic Course of Events:	The Use Case begins after John has confirmed his booking. 1. The PMS requires a preauthorization for the stay 2. John is asked to insert his credit card 3. The kiosk sends the payment information to the PMS (including credit card number) and waits for payment confirmation 4. The PMS confirms preauthorization and send authorization number to the kiosk The Use Case ends when the PMS confirms check-in is complete.
Exception Path:	In Step 5, Payment is refused. PMS sends info to the kiosk which directs John to the desk. In Step 1, No payment is needed. The whole Use-Case is cancelled.
Alternative Paths:	In Step 1, the PMS requires a prepayment instead of a preauthorization.
Assumptions:	PMS is linked to a payment gateway.
Preconditions:	John has confirmed his booking.

Postconditions: Required payment has been accepted.

5.11 Check-In

5.11.1 Messaging Use Case



Use Case Name:	Check In
Summary:	<p>The Guest checks in through the Kiosk. The Guest finds his booking by entering some ID criteria. The Kiosk validates that the combination of criteria is enough ID and requests a booking from the PMS that matches the ID criteria and is checking in today.</p> <p>The Guest has the ability to manage the non-paying shares associated with his booking. He can change his departure date where the PMS allows him to. If he wishes he can request a list of rooms and select which room he wishes to be allocated. The list of rooms provided by the PMS may optionally include rooms of an alternate type either at the same rate or increased rate. If settling to a Payment Card, the Guest may change his Payment card to an alternate one. The Guest may search for his profile to attach to the booking. Proceeding with the check-in, the Kiosk authorizes the guest's Payment Card and cuts one or more keys.</p>
Basic Course of Events:	<p>The Use Case Begins when the Guest Elects to Check in using the Self Service Kiosk.</p> <ol style="list-style-type: none"> 1. The Guest provides sufficient ID to finds his booking 2. The Kiosk sends a booking search message to the PMS 3. The PMS finds the booking and returns a booking summary 4. The Kiosk displays the Booking summary and asks the Guest to confirm it 5. The Guest optionally changes some parts of the booking as allowed by the PMS and Kiosk <p>The Use Case Ends when the Guest has Checked In using the self service Kiosk.</p>
Extension Points:	<ul style="list-style-type: none"> • Change Guest Names • Changing Booking Departure Date • Requesting a specific Room • Selecting to change Payment Card Guaranteeing the booking • Searching for a Guest Profile • Authorizing Payment Card • Cutting a Key
Assumptions:	The Guest has a booking at the hotel and is due to arrive today.
Preconditions:	The Guest is not Checked In.
Postconditions:	The Guest is Checked In.

5.11.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRQ	1	Hotel Reservation Notif Request supports the functionality of updating other systems with reservation data. The message assumes a push model, with the originating system pushing the data to another system. The originating system would usually be a booking source, such as a Global Distribution System (GDS), a Central Reservation System (CRS) or some other agent of the hotel.
@EchoToken	0..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.
@ResStatus	1	To specify the type of action requested when more than one function could be handled by the message. The value should be 'Modify' .
@TimeStamp	0..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 OTA versioned messages, the version of the message is indicated by a decimal value.
@Target	0..1	Used to indicate whether the request is for the Test or Production system. Value can be either 'Test' or 'Production'.
OTA_HotelResNotifRQ / POS / Source	1	Provides information on the source of a request.
@TerminalID	0..1	The electronic address of the device from which information is entered.
OTA_HotelResNotifRQ / POS / Source / RequestorID	0..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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation	1	The Reservation class contains the current reservation being created or altered.
@ResStatus	1	The status of the reservation. Should be "In-house" when checking-in a reservation.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / UniqueID	1	The booking reference for the reservation.
@Type	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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).

5.11.3 Sample Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRQ EchoToken="2c37c576-f725-4609-bfa4-e30bd43460ca" ResStatus="Modify" TimeStamp="2010-02-12T12:26:47" Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="TRM123">
      <RequestorID Type="18" ID="KSKSYS123"/>
    </Source>
  </POS>
</OTA_HotelResNotifRQ>
```

```

</POS>
<HotelReservations>
  <HotelReservation ResStatus="In-house" >
    <UniqueID Type="14" ID="RES123456" />
  </HotelReservation>
</HotelReservations>
</OTA_HotelResNotifRQ>

```

5.11.4 Data Element Table – Response

Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRS	1	Hotel Reservation Notif Request supports the functionality of updating other systems with reservation data. The message assumes a push model, with the originating system pushing the data to another system. The originating system would usually be a booking source, such as a Global Distribution System (GDS), a Central Reservation System (CRS) or some other agent of the hotel.
@EchoToken	0..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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
OTA_HotelResNotifRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
OTA_HotelResNotifRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
OTA_HotelResNotifRS / 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).

5.11.5 Sample Response Message

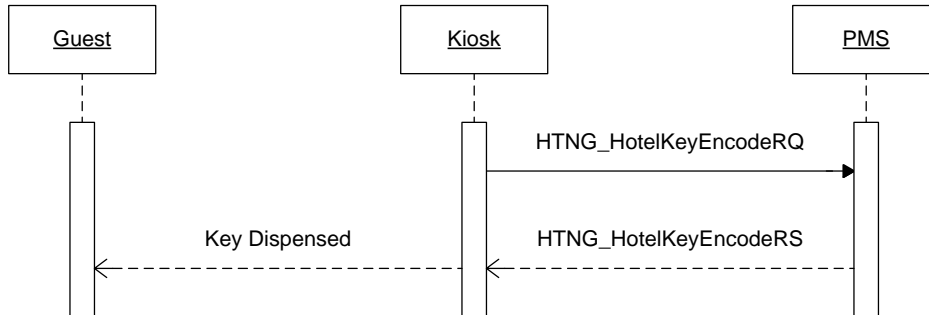
```

<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRS EchoToken="2c37c576-f725-4609-bfa4-e30bd43460ca" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Success/>
</OTA_HotelResNotifRS>

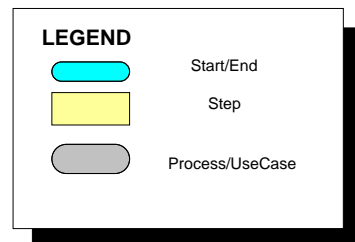
```

5.12 Cut Key Request

5.12.1 Messaging Use Case



Use Case Name:	Cut Key Request – New Room Key / Duplicate Key / Replacement Key
Summary:	Kiosk requests the Key Encoding Data from PMS or from the Key System. Kiosk sends Encoding data to the card writer. One or more room keys are encoded, duplicated, or replaced.
Basic Course of Events:	See the use-case diagram above.
Exception Path:	<ul style="list-style-type: none"> - Attempt to write a room key can end up with a hardware failure. In this case, if no keys are made, the use-case should fail. If less than requested keys are made, the completion is "partial success". - Attempt to query PMS or the Key System can end-up with a failure. In this case, no keys are made.
Alternative Paths:	See the use-case diagram (below): <ul style="list-style-type: none"> - 2a/2b – PMS or KeySystem request for Encoding Data
Extension Points:	See the use-case diagram (below): <ul style="list-style-type: none"> - 2a/2b - PMS or KeySystem request for Encoding Data - 4 – Encode one room key
Trigger:	Kiosk application is ready to encode the room keys.
Preconditions:	See the use-case diagram (below): <ul style="list-style-type: none"> - 1 – pre-conditions
Postconditions:	Guest has the room keys.



Element @Attribute	Num	Description/Contents
HTNG_HotelKeyEncodeRQ	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

Element @Attribute	Num	Description/Contents
		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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
HTNG_HotelKeyEncodeRQ / POS / Source	0..1	Holds details regarding the requestor. It may be repeated to also accommodate the delivery systems.
@TerminalID	1	The electronic address of the device from which information is entered.
HTNG_HotelKeyEncodeRQ / 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_HotelKeyEncodeRQ / UniqueID	1	An identifier used to uniquely reference an object in a system (e.g., an airline reservation reference, customer profile reference, booking confirmation number, or a reference to a previous availability quote).
@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_HotelKeyEncodeRQ / HotelKeyEncode	1	
@KeyType	1	Indicates the type and format of key to be created. Possible values are: <ul style="list-style-type: none"> • "Add" for new key • "Add-Update" for duplicate key • "Replace" for replacement key
@Quantity	0..1	Used to define the quantity for an associated element or attribute.
@ReturnTrackData	0..1	If "true", indicates the track data to be encoded should be returned by the key system so a key can be encoded by hardware controlled by the sender. A value of "false" indicates the key systems should encode the key.
HTNG_HotelKeyEncodeRQ / HotelKeyEncode / Encoder	0..1	Information regarding the encoder to be used to create the key. Is likely only used when ReturnTrackData = "false".
@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_HotelKeyEncodeRQ / HotelKeyEncode / MagneticData	0..1	Card Magnetic Stripe Data as defined by ISO 7813 for banking cards.

Element @Attribute	Num	Description/Contents
@Track1	0..1	The binary magnetic stripe data for track 1.
@Track2	0..1	The binary magnetic stripe data for track 2.
@Track3	0..1	The binary magnetic stripe data for track 3.
HTNG_HotelKeyEncodeRQ / HotelKeyEncode / SmartCardData	0..1	If the means of entry is a smartcard (versus a magnetic key card) this element must be present.
@CardType	0..1	Indicates the type and format of key to be created.
@CardID	1	Unique identifier for the specific smartcard to be encoded.
HTNG_HotelKeyEncodeRQ / HotelKeyEncode / SmartCardData / ReadOnlyData	0..1	Data that should be written to the smart card and not allowed to be modified in subsequent requests.
HTNG_HotelKeyEncodeRQ / HotelKeyEncode / TimeSpan	1	The attributes of the OTA DateTimeSpan data type are based on the W3C base data types of <code>timeInstant</code> and <code>timeDuration</code> . The lexical representation for <code>timeDuration</code> is the [ISO 8601] extended format <code>PnYn MnDTnH nMnS</code> , where <code>nY</code> represents the number of years, <code>nM</code> the number of months, <code>nD</code> the number of days, 'T' is the date/time separator, <code>nH</code> the number of hours, <code>nM</code> the number of minutes and <code>nS</code> the number of seconds. The number of seconds can include decimal digits to arbitrary precision. As an example, 7 months, 2 days, 2hours and 30 minutes would be expressed as <code>P0Y7M2DT2H30M0S</code> . Truncated representations are allowed provided they conform to ISO 8601 format. Time periods, i.e., specific durations of time, can be represented by supplying two items of information: a start instant and a duration or a start instant and an end instant or an end instant and a duration. The OTA standards use the XML mapping that provides for two elements to represent the specific period of time; a <code>startInstant</code> and a <code>duration</code> .
@End	1	The ending value of the time span.
@Start	0..1	The starting value of the time span.
HTNG_HotelKeyEncodeRQ / HotelKeyEncode / AccessAreas	0..1	A collection of additional restricted areas the guest will be able to access using their room key.
HTNG_HotelKeyEncodeRQ / HotelKeyEncode / AccessAreas / AccessArea	1..n	An individual access area.
@Type	0..1	A reference to the type of object defined by the <code>UniqueID</code> 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_HotelKeyEncodeRQ / HotelKeyEncode / RoomType	1	Provides details regarding rooms, usually guest rooms.
@Composite	0..1	Indicates that the room (suite) is a composite of smaller units.
@RoomID	1	A string value representing the unique identification of a room if the request is looking for a specific room.
HTNG_HotelKeyEncodeRQ / HotelKeyEncode / RoomType / ComponentRooms	0..1	A collection of connecting rooms or suite components
HTNG_HotelKeyEncodeRQ / HotelKeyEncode / RoomType / ComponentRooms /	1..n	An individual room that connects to another room, or is part of a suite.

Element @Attribute	Num	Description/Contents
ComponentRoom		
@RoomID	1	A string value representing the unique identification of a room if the request is looking for a specific room.

5.12.3 Sample Request Message – Encode Hotel Room Key

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_HotelKeyEncodeRQ EchoToken="96fd8433-1935-491e-9a02-97d1c5a20352" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://htng.org/2010A" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="TRM123">
      <RequestorID Type="18" ID="KSKSYS123"/>
    </Source>
  </POS>
  <UniqueID Type="14" ID="RES123456"/>
  <HotelKeyEncode KeyType="Add-Update" Quantity="1" ReturnTrackData="false">
    <Encoder Type="0" ID="ENC123456"/>
    <MagneticData Track2="MTIzNDU2"></MagneticData>
    <TimeSpan End="2010-08-15T12:00:00" Start="2010-08-13T16:37:24"/>
    <AccessAreas>
      <AccessArea Type="27" ID="POOL"/>
    </AccessAreas>
    <RoomType Composite="false" RoomID="1706"/>
  </HotelKeyEncode>
</HTNG_HotelKeyEncodeRQ>
```

5.12.4 Sample Request Message – Request Hotel Room Key Encoding

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_HotelKeyEncodeRQ EchoToken="96fd8433-1935-491e-9a02-97d1c5a20352" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://htng.org/2010A" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="TRM123">
      <RequestorID Type="18" ID="KSKSYS123"/>
    </Source>
  </POS>
  <UniqueID Type="14" ID="RES123456"/>
  <HotelKeyEncode KeyType="Add-Update" ReturnTrackData="true">
    <MagneticData Track2="MTIzNDU2"></MagneticData>
    <TimeSpan End="2010-08-15T12:00:00" Start="2010-08-13T16:37:24"/>
    <AccessAreas>
      <AccessArea Type="27" ID="POOL"/>
    </AccessAreas>
    <RoomType Composite="false" RoomID="1706"/>
  </HotelKeyEncode>
</HTNG_HotelKeyEncodeRQ>
```

5.12.5 Data Element Table – Response

Element @Attribute	Num	Description/Contents
HTNG_HotelKeyEncoderRS	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
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
HTNG_HotelKeyEncodeRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_HotelKeyEncodeRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
HTNG_HotelKeyEncodeRS / 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_HotelKeyEncodeRS / UniqueID	1	An identifier used to uniquely reference an object in a system (e.g., an airline reservation reference, customer profile reference, booking confirmation number, or a reference to a previous availability quote).
@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_HotelKeyEncodeRS / HotelKeyEncode	0..1	The base element containing all of the necessary information to perform a local or remote key cut. It is important to note that many of the child elements or attributes may or may not be echoed-back in the response message for the convenience of the sender.
@KeyType	0..1	Indicates the type and format of key created. Possible values are: <ul style="list-style-type: none"> • "Add" for new key • "Add-Update" for duplicate key • "Replace" for replacement key
@Quantity	1	Used to define the quantity for an associated element or attribute.
@ReturnTrackData	0..1	Indicates whether the track data to be encoded should be returned so it can be created locally (versus the receiver encoding the key directly).
HTNG_HotelKeyEncodeRS / HotelKeyEncode / Encoder	1	Information regarding the encoder to be used to create the key.
@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.
OTA_HotelKeyEncodeRS / HotelKeyEncode / MagneticData	0..1	Card Magnetic Stripe Data as defined by ISO 7813 for banking cards.
@Track1	0..1	The binary magnetic stripe data for track 1.

Element @Attribute	Num	Description/Contents
@Track2	0..1	The binary magnetic stripe data for track 2.
@Track3	0..1	The binary magnetic stripe data for track 3.
OTA_HotelKeyEncoderRS / HotelKeyEncode / SmartCardData	0..1	If the means of entry is a smartcard (versus a magnetic key card) this element must be present.
@CardType	1	Indicates the type and format of key to be created.
OTA_HotelKeyEncoderRS / HotelKeyEncode / SmartCardData / ReadOnlyData	1	Data that should be written to the smart card and not allowed to be modified in subsequent requests.
OTA_HotelKeyEncoderRS / HotelKeyEncode / TimeSpan	1	The attributes of the OTA DateTimeSpan data type are based on the W3C base data types of timeInstant and timeDuration. The lexical representation for timeDuration is the [ISO 8601] extended format PnYn MnDTnH nMnS, where nY represents the number of years, nM the number of months, nD the number of days, 'T' is the date/time separator, nH the number of hours, nM the number of minutes and nS the number of seconds. The number of seconds can include decimal digits to arbitrary precision. As an example, 7 months, 2 days, 2hours and 30 minutes would be expressed as P0Y7M2DT2H30M0S. Truncated representations are allowed provided they conform to ISO 8601 format. Time periods, i.e., specific durations of time, can be represented by supplying two items of information: a start instant and a duration or a start instant and an end instant or an end instant and a duration. The OTA standards use the XML mapping that provides for two elements to represent the specific period of time; a startInstant and a duration.
@End	1	The ending value of the time span.
@Start	1	The starting value of the time span.
OTA_HotelKeyEncoderRS / HotelKeyEncode / AccessAreas	0..1	A collection of additional restricted areas the guest will be able to access using their room key.
OTA_HotelKeyEncoderRS / HotelKeyEncode / AccessAreas / AccessArea	1..n	An individual access area.
@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.
OTA_HotelKeyEncoderRS / HotelKeyEncode / RoomType	1	Provides details regarding rooms, usually guest rooms.
@Composite	0..1	Indicates that the room (suite) is a composite of smaller units.
@RoomID	1	A string value representing the unique identification of a room if the request is looking for a specific room.
OTA_HotelKeyEncoderRS / HotelKeyEncode / ComponentRooms	0..1	A collection of connecting rooms or suite components.
OTA_HotelKeyEncoderRS / HotelKeyEncode / ComponentRooms / ComponentRoom	1..n	An individual room that connects to another room, or is part of a suite.
@RoomID	1	A string value representing the unique identification of a room if the request is looking for a specific room.

5.12.6 Sample Response Message – Encode Hotel Room Key

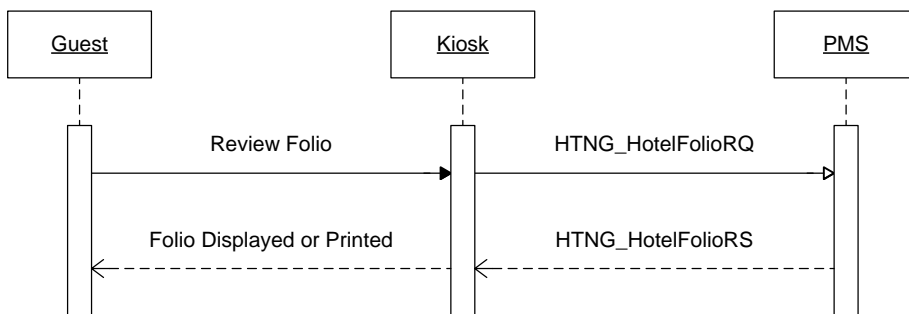
```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_HotelKeyEncoderS EchoToken="96fd8433-1935-491e-9a02-97d1c5a20352" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://htng.org/2010A" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Success/>
  <UniqueID Type="14" ID="RES123456"/>
  <HotelKeyEncode KeyType="Add-Update" Quantity="1" ReturnTrackData="false">
    <AccessAreas>
      <AccessArea Type="27" ID="POOL"/>
      <AccessArea Type="27" ID="EASTENTRANCE"/>
    </AccessAreas>
  </HotelKeyEncode>
</HTNG_HotelKeyEncoderS>
```

5.12.7 Sample Response Message - Request Hotel Room Key Encoding

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_HotelKeyEncoderS EchoToken="96fd8433-1935-491e-9a02-97d1c5a20352" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://htng.org/2010A" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Success/>
  <UniqueID Type="14" ID="RES123456"/>
  <HotelKeyEncode KeyType="Add-Update" Quantity="1" ReturnTrackData="false">
    <MagneticData Track2="MTIzNDU2" Track3="MTIzNDU2"></MagneticData>
    <AccessAreas>
      <AccessArea Type="27" ID="POOL"/>
      <AccessArea Type="27" ID="EASTENTRANCE"/>
    </AccessAreas>
  </HotelKeyEncode>
</HTNG_HotelKeyEncoderS>
```

5.13 Retrieve Folio

5.13.1 Messaging Use Case



Use Case Name:	Folio Retrieval
Summary:	Kiosk requests the folios for an In-house guest. The PMS uses the reservation number to enumerate the appropriate folios. The PMS then returns each folio attached to the reservation.
Basic Course of Events:	The Use Case Begins when the guest requests to view their folio. The Use Case Ends when they are presented with their folio.
Exception Path:	-Reservation not found. -Folio not found.
Trigger:	Guest wants to review the charged incurred during their stay.
Assumptions:	Guest is in-house.
Preconditions:	Reservation is checked-in at the hotel.
Postconditions:	Guest receives a copy of their folio(s).

Business Rules:	The ability for a guest to view their folio, and when, is driven by the business rules of the Hotel.
-----------------	--

5.13.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
HTNG_HotelFolioRQ	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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
HTNG_HotelFolioRQ / POS / Source	1	This holds details regarding the requestor. It may be repeated to also accommodate the delivery systems.
@TerminalID	0..1	This is the electronic address of the device from which information is entered.
HTNG_HotelFolioRQ / 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_HotelFolioRQ / UniqueID	1	An identifier used to uniquely reference an object in a system (e.g., an airline reservation reference, customer profile reference, booking confirmation number, or a reference to a previous availability quote).
@Type	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.

5.13.3 Sample Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_HotelFolioRQ EchoToken="e23a0dab-9a03-4ab6-8c8e-7b9cb68fdf2d" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://htng.org/2010A" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="TRM123">
      <RequestorID Type="18" ID="KSKSYS123"/>
    </Source>
  </POS>
  <UniqueID Type="14" ID="RES123456"/>
</HTNG_HotelFolioRQ>
```

5.13.4 Data Element Table – Response

Element @Attribute	Num	Description/Contents
----------------------	-----	----------------------

Element @Attribute	Num	Description/Contents
HTNG_HotelFolioRS	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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
HTNG_HotelFolioRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_HotelFolioRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
HTNG_HotelFolioRS / 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_HotelFolioRS / UniqueID	1	An identifier used to uniquely reference an object in a system (e.g., an airline reservation reference, customer profile reference, booking confirmation number, or a reference to a previous availability quote).
@Type	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_HotelFolioRS / Folios	0..1	A collection of folios.
HTNG_HotelFolioRS / Folios / Folio	1..n	An individual folio of charges.
@GuestPayable	1	When true, the guest may provide payment for the folio. When false, the guest may not provide payment for the folio.
@FolioID	1	Unique PMS identifier if a given folio.
@InvoiceID	0..1	
@FolioGroupingID	0..1	For folios with a FolioType of "Guest" this differentiates them. This provides flexibility to group incidental charges, room charges, or any other logical grouping.

Element @Attribute	Num	Description/Contents
@GuestViewable	1	When true, the comment may be shown to the consumer. When false, the comment may not be shown to the consumer.
@FolioType	1	An enumeration of the following values: Group, Corporate, Wholesaler, Comp Accounting, Guest, Package, Other.
HTNG_HotelFolioRS / Folios / Folio / BasicPropertyInfo	0..1	An abbreviated short summary of hotel descriptive information.
@HotelName	0..1	A text field used to communicate the proper name of the hotel.
@HotelCode	0..1	The code that uniquely identifies a single hotel property. The hotel code is decided between vendors.
@HotelCodeContext	0..1	A text field used to communicate the context (or source of – i.e., Sabre, Galileo, Worldspan, Amadeus) the HotelReferenceGroup codes.
HTNG_HotelFolioRS / Folios / Folio / BasicPropertyInfo / Address / AddressLine	0..5	When the address is unformatted (FormattedInd="false") these lines will contain free form address details. When the address is formatted and street number and street name must be sent independently, the street number will be sent using StreetNmbr, and the street name will be sent in the first AddressLine occurrence.
HTNG_HotelFolioRS / Folios / Folio / BasicPropertyInfo / Address / CityName	0..1	City (e.g., Dublin), town, or postal station (i.e., a postal service territory, often used in a military address).
HTNG_HotelFolioRS / Folios / Folio / BasicPropertyInfo / Address / PostalCode	0..1	Post Office Code number.
HTNG_HotelFolioRS / Folios / Folio / BasicPropertyInfo / Address / StateProv	0..1	State or Province name (e.g., Texas).
@StateCode	0..1	The standard code or abbreviation for the state, province, or region.
HTNG_HotelFolioRS / Folios / Folio / BasicPropertyInfo / Address / CountryName	0..1	Country name (e.g., Ireland).
@Code	0..1	ISO 3166 code for a country.
HTNG_HotelFolioRS / Folios / Folio / BasicPropertyInfo / ContactNumbers / ContactNumber	0..1	Contact numbers of the hotel property. Examples are telephone and fax numbers.
@PhoneNumber	0..1	Telephone number assigned to a single location.
@CountryAccessCode	0..1	Code assigned by telecommunications authorities for international country access identifier.
@AreaCityCode	0..1	Code assigned for telephones in a specific region, city, or area.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile	1	The profile for the customer incurring the charges.
@ProfileType	1	Code to specify a profile such as Customer, Tour Operator, Corporation, etc. Refer to OpenTravel Code List Profile Type (PRT).
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / PersonName / GivenName	0..1	Given name, first name or names.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / PersonName / MiddleName	0..1	The middle name of the person name
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / PersonName /	0..1	Family name, last name.

Element @Attribute	Num	Description/Contents
Surname		
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / PersonName / NameSuffix	0..1	Hold various name suffixes and letters (e.g., Jr., Sr., III, Ret., Esq.).
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / PersonName / NameTitle	0..1	Degree or honors (e.g., Ph.D., M.D.).
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / Address / AddressLine	0..1	When the address is unformatted (FormattedInd="false") these lines will contain free form address details. When the address is formatted and street number and street name must be sent independently, the street number will be sent using StreetNmbr, and the street name will be sent in the first AddressLine occurrence.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / Address / CityName	0..1	City (e.g., Dublin), town, or postal station (i.e., a postal service territory, often used in a military address).
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / Address / PostalCode	0..1	Post Office Code number.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / Address / StateProv	0..1	State or Province name (e.g., Texas).
@StateCode	0..1	The standard code or abbreviation for the state, province, or region.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / Customer / Address / CountryName	0..1	Country name (e.g., Ireland).
@Code	0..1	A two character country code as defined in ISO3166.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / CompanyInfo / CompanyName	1	Identifies a company by name.
@CodeContext	0..1	Identifies the context of the identifying code, such as DUNS, IATA or internal code, etc.
@CompanyShortName	0..1	Used to provide the company common name.
@Code	1	Identifies a company by the company code.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / CompanyInfo / AddressInfo / AddressLine	0..5	When the address is unformatted (FormattedInd="false") these lines will contain free form address details. When the address is formatted and street number and street name must be sent independently, the street number will be sent using StreetNmbr, and the street name will be sent in the first AddressLine occurrence.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / CompanyInfo / AddressInfo / CityName	0..1	City (e.g., Dublin), town, or postal station (i.e., a postal service territory, often used in a military address).
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / CompanyInfo / AddressInfo / PostalCode	0..1	Post Office Code number.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / CompanyInfo / AddressInfo / StateProv	0..1	State or Province name (e.g., Texas).

Element @Attribute	Num	Description/Contents
@StateCode	0..1	The standard code or abbreviation for the state, province, or region.
HTNG_HotelFolioRS / Folios / Folio / CustomerProfile / CompanyInfo / AddressInfo / CountryName	0..1	The name or code of a country (e.g., as used in an address or to specify citizenship of a traveler).
@Code	0..1	A two character country code as defined in ISO3166.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile	0..1	The profile for the entity with financial responsibility for the charges incurred.
@ProfileType	1	Code to specify a profile such as Customer, Tour Operator, Corporation, etc. Refer to OpenTravel Code List Profile Type (PRT).
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / PersonName / GivenName	0..1	Given name, first name or names.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / PersonName / MiddleName	0..1	The middle name of the person name
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / PersonName / Surname	0..1	Family name, last name.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / PersonName / NameSuffix	0..1	Hold various name suffixes and letters (e.g., Jr., Sr., III, Ret., Esq.).
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / PersonName / NameTitle	0..1	Degree or honors (e.g., Ph.D., M.D.).
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / Address / AddressLine	0..5	When the address is unformatted (FormattedInd="false") these lines will contain free form address details. When the address is formatted and street number and street name must be sent independently, the street number will be sent using StreetNmbr, and the street name will be sent in the first AddressLine occurrence.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / Address / CityName	0..1	City (e.g., Dublin), town, or postal station (i.e., a postal service territory, often used in a military address).
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / Address / PostalCode	0..1	Post Office Code number.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / Address / StateProv	0..1	State or Province name (e.g., Texas).
@StateCode	0..1	The standard code or abbreviation for the state, province, or region.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / Customer / Address CountryName	0..1	Country name (e.g., Ireland).
@Code	0..1	A two character country code as defined in ISO3166.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / CompanyInfo / CompanyName	0..1	Identifies a company by name.
@CodeContext	0..1	Identifies the context of the identifying code, such as DUNS, IATA or internal code, etc.
@CompanyShortName	0..1	Used to provide the company common name.
@Code	0..1	Identifies a company by the company code.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile /	0..5	When the address is unformatted (FormattedInd="false") these lines will contain free form address details. When the address is formatted and street

Element @Attribute	Num	Description/Contents
CompanyInfo / AddressInfo / AddressLine		number and street name must be sent independently, the street number will be sent using StreetNmbr, and the street name will be sent in the first AddressLine occurrence.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / CompanyInfo / AddressInfo / CityName	0..1	City (e.g., Dublin), town, or postal station (i.e., a postal service territory, often used in a military address).
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / CompanyInfo / AddressInfo / PostalCode	0..1	Post Office Code number.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / CompanyInfo / AddressInfo / StateProv	0..1	State or Province name (e.g., Texas).
@StateCode	0..1	The standard code or abbreviation for the state, province, or region.
HTNG_HotelFolioRS / Folios / Folio / PayerProfile / CompanyInfo / AddressInfo / CountryName	0..1	The name or code of a country (e.g., as used in an address or to specify citizenship of a traveller).
@Code	0..1	A 2 character country code as defined in ISO3166.
HTNG_HotelFolioRS / Folios / Folio / RevenueSummary / GrossAmount	0..1	The total booking cost to the customer.
@Amount	1	A monetary amount.
HTNG_HotelFolioRS / Folios / Folio / RevenueSummary / TaxItems	0..1	The total taxes charged to the customer.
@Amount	1	A monetary amount.
HTNG_HotelFolioRS / Folios / Folio / RevenueSummary / BalanceDueAmount	1	The amount remaining to be paid by the customer, i.e., GrossAmount less the greater of DepositAmount and Amount Received.
@Amount	1	A monetary amount.
HTNG_HotelFolioRS / Folios / Folio / RevenueSummary / AmountReceived	0..1	The payment amount received against the reservation.
@Amount	1	A monetary amount.
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail	1..n	The line item detail of specific revenue transactions.
@ReferenceID	0..1	The unique transaction identifier for this posting.
@TransactionDate	1	The date the transaction was posted.
@PMSRevenueCode	0..1	The transaction code assigned by the PMS.
@CurrencyCode	0..1	The code specifying a monetary unit. Use ISO 4217, three alpha code.
@Amount	1	A monetary amount.
@Description	1	The line item detail description for this posting.

Element @Attribute	Num	Description/Contents
@DecimalPlaces	1	Indicates the number of decimal places for a particular currency. This is equivalent to the ISO 4217 standard "minor unit". Typically used when the amount provided includes the minor unit of currency without a decimal point (e.g., USD 8500 needs DecimalPlaces="2" to represent \$85).
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / FolioIDs	1	This element is no implemented in the Kiosk messaging, but the OpenTravel schema requires its use. Only a single child FolioID is needed in order fulfill the schema requirement.
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / FolioIDs / FolioID	1	Even though this value is not used in the Kiosk messaging, it's value should match the value of \\HTNG_HotelFolioRS\Folios\Folio@FolioID.
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / UnitPrice	0..1	The unit amount charged for the service including additional amounts and fees.
@AmountBeforeTax	0..1	The unit amount not including any associated tax (e.g., sales tax, VAT, GST or any associated tax).
@AmountAfterTax	0..1	The unit amount including all associated taxes (e.g., sales tax, VAT, GST or any associated tax).
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / UnitPrice / Taxes	0..1	A collection of taxes.
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / UnitPrice / Taxes / Tax	1..n	Applicable tax element. This element allows for both percentages and flat amounts. If one field is used, the other should be zero since logically, taxes should be calculated in only one of the two ways.
@Type	0..1	Used to indicate if the amount is inclusive or exclusive of other charges, such as taxes, or is cumulative (amounts have been added to each other).
@Percent	0..1	Fee percentage; if zero, assume use of the Amount attribute (Amount or Percent must be a zero value).
@Amount	1	A monetary amount.
@Code	0..1	Code identifying the fee (e.g., agency fee, municipality fee). Refer to OpenTravel Code List Fee Tax Type (FTT).
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / UnitPrice / Taxes / Tax / TaxDescription	1	The tax description item.
@Name	1	The line item detail description for this tax posting.
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / ExtendedPrice	0..1	The total amount charged for the service including additional amounts and fees.
@AmountBeforeTax	0..1	The unit amount not including any associated tax (e.g., sales tax, VAT, GST or any associated tax).
@AmountAfterTax	0..1	The unit amount including all associated taxes (e.g., sales tax, VAT, GST or any associated tax).
@Quantity	1	The number of items purchased.
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / ExtendedPrice / Taxes	0..1	A collection of taxes.
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / ExtendedPrice / Taxes / Tax	1..n	Applicable tax element. This element allows for both percentages and flat amounts. If one field is used, the other should be zero since logically, taxes should be calculated in only one of the two ways.
@Type	0..1	Used to indicate if the amount is inclusive or exclusive of other charges, such as taxes, or is cumulative (amounts have been added to each other).

Element @Attribute	Num	Description/Contents
@Percent	0..1	Fee percentage. If zero, assume use of the Amount attribute (Amount or Percent must be a zero value).
@Amount	1	A monetary amount.
@Code	0..1	Code identifying the fee (e.g., agency fee, municipality fee). Refer to OpenTravel Code List Fee Tax Type (FTT).
HTNG_HotelFolioRS / Folios / Folio / RevenueDetails / RevenueDetail / ExtendedPrice / Taxes / Tax / TaxDescription	1	The tax description item.
@Name	1	The line item detail description for this tax posting.

5.13.5 Sample Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_HotelFolioRS EchoToken="e23a0dab-9a03-4ab6-8c8e-7b9cb68fdf2d" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://htng.org/2010A" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Success/>
  <UniqueID Type="14" ID="RES123456"/>
  <Folios>
    <Folio GuestPayable="true" FolioID="74328654" InvoiceID="7897435" FolioGroupingID="G"
GuestViewable="true" FolioType="Incidental">
      <BasicPropertyInfo HotelName="Sunnyview Hotel" HotelCode="SNYVW">
        <Address>
          <AddressLine>10 E Sunnyview Circle</AddressLine>
          <CityName>Sunnytown</CityName>
          <PostalCode>65432</PostalCode>
          <StateProv StateCode="CA">California</StateProv>
          <CountryName Code="US">United States of America</CountryName>
        </Address>
      </BasicPropertyInfo>
      <CustomerProfile ProfileType="1">
        <Customer>
          <PersonName>
            <NamePrefix>Mr.</NamePrefix>
            <GivenName>John</GivenName>
            <MiddleName>A</MiddleName>
            <Surname>Smith</Surname>
          </PersonName>
          <Address>
            <AddressLine>101 Main Street</AddressLine>
            <CityName>Anytown</CityName>
            <PostalCode>012345</PostalCode>
            <StateProv StateCode="PA">Pennsylvania</StateProv>
            <CountryName Code="US">United States of America</CountryName>
          </Address>
        </Customer>
      </CustomerProfile>
      <PayerProfile>
        <Customer>
          <PersonName>
            <NamePrefix>Mr.</NamePrefix>
            <GivenName>John</GivenName>
            <MiddleName>A</MiddleName>
            <Surname>Smith</Surname>
          </PersonName>
          <Address>
            <AddressLine>101 Main Street</AddressLine>
            <CityName>Anytown</CityName>
            <PostalCode>012345</PostalCode>
            <StateProv StateCode="PA">Pennsylvania</StateProv>
            <CountryName Code="US">United States of America</CountryName>
          </Address>
        </Customer>
      </PayerProfile>
    </Folio>
  </Folios>
</HTNG_HotelFolioRS>
```

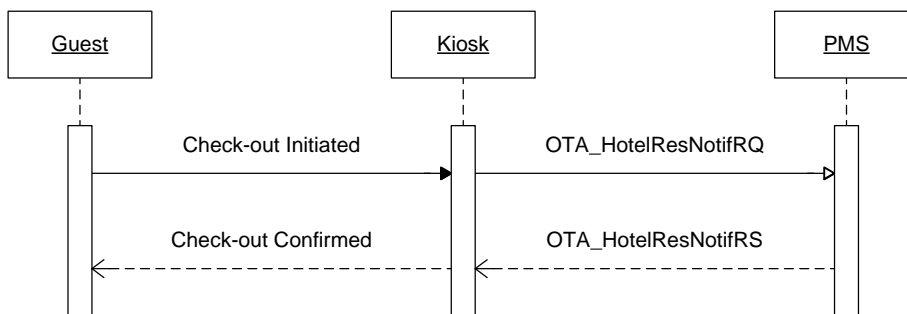
```

        <RevenueSummary>
            <GrossAmount Amount="500.00" />
            <TaxItems Amount="50.00" />
            <BalanceDueAmount Amount="550.00" />
            <AmountReceived Amount="0.00" />
        </RevenueSummary>
        <RevenueDetails>
            <RevenueDetail ReferenceID="REF12345" TransactionDate="2010-08-13"
PMSRevenueCode="ROOM" Amount="275.00" Description="Room Revenue">
                <FolioIDs>
                    <FolioID>74328654</FolioID>
                </FolioIDs>
                <ExtendedPrice AmountBeforeTax="250.00" Quantity="1">
                    <Taxes>
                        <Tax Type="Exclusive" Percent="10.00" Amount="25.00">
                            <TaxDescription Name="State Rooms and Meals Tax"/>
                        </Tax>
                    </Taxes>
                </ExtendedPrice>
            </RevenueDetail>
            <RevenueDetail ReferenceID="REF12346" TransactionDate="2010-08-14"
PMSRevenueCode="ROOM" Amount="275.00" Description="Room Revenue">
                <FolioIDs>
                    <FolioID>74328654</FolioID>
                </FolioIDs>
                <ExtendedPrice AmountBeforeTax="250.00" Quantity="1">
                    <Taxes>
                        <Tax Type="Exclusive" Percent="10.00" Amount="25.00">
                            <TaxDescription Name="State Rooms and Meals Tax"/>
                        </Tax>
                    </Taxes>
                </ExtendedPrice>
            </RevenueDetail>
        </RevenueDetails>
    </Folio>
</Folios>
</HTNG_HotelFolioRS>

```

5.14 Check-Out

5.14.1 Messaging Use Case



Use Case Name:	Check Out
Summary:	The Guest elects to Check out of the hotel using the Kiosk. The Guest identifies himself, reviews his folio and adds a mini bar charge, pays with his Credit Card, prints his folio and is checked out of the hotel.

Basic Course of Events:	<p>This Use Case Begins when the Guest, John, goes to the Kiosk to Check Out.</p> <ol style="list-style-type: none"> 1. John enters a method of identification 2. The Kiosk performs a Booking Search for the single in house booking that matches the identification presented 3. The PMS finds a single booking that matches the entered criteria 4. The PMS returns the selected booking to the Kiosk 5. The Kiosk then requests the Folio Display 6. The Kiosk displays the Folio for John to Review 7. The Kiosk prompts John to enter any Mini Bar Charges 8. John enters Mini Bar Charges. The Kiosk Sends a Post Charge message to the PMS 9. The PMS updates the Guest Folio 10. The Guest settles their folio using their Credit Card 11. The Kiosk prints a zero value invoice 12. The Kiosk sends the PMS a Check Out Message 13. The PMS checks John out of the room and hotel 14. The PMS sends the kiosk a Check Out Complete Message <p>This Use Case ends when the Guest has Checked Out.</p>
Exception Path:	<p>In Step 3, in the case where a booking is not found for the ID provided by John.</p> <ol style="list-style-type: none"> 1. The PMS returns a booking not found message 2. The kiosk asks John to try again with different details <p>In the Case where multiple bookings are found that match,</p> <ol style="list-style-type: none"> 1. The PMS returns a more than one match found message 2. The Kiosk asks for an additional ID element to confirm the booking 3. John Supplies an ID Element and the Kiosk Refines the Booking Search message 4. The PMS finds a single Match and returns the booking for display.
Alternative Paths:	<p>In Step 4 the Kiosk alerts John that he has unread Messages</p> <ol style="list-style-type: none"> 1. John reads his message 2. The Kiosk sends a Message Read notification to the PMS 3. The PMS marks the message as read 4. John repeats for each unread message
Extension Points:	<p>In Step 8 John decides to separately settle his Mini Bar Charges as he does not want the Mini Bar Charges on the invoice he will file for expense reimbursement.</p> <ol style="list-style-type: none"> 1. John selects the three Mini Bar Charges on his Folio 2. John settles these to his Credit Card 3. The Kiosk Prints a zero balance Invoice including the Mini Bard Charges and payment only 4. John then settles the remaining transactions and receives a 2nd invoice with the other charges and payment
Assumptions:	John is a Checked In Guest.
Preconditions:	John has not Checked Out.
Postconditions:	John has Checked Out.

5.14.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRQ	1	Hotel Reservation Notif Request supports the functionality of updating other systems with reservation data. The message assumes a push model, with the originating system pushing the data to another system. The originating system would usually be a booking source, such as a Global Distribution System (GDS), a Central Reservation System (CRS) or some other agent of the hotel.
@EchoToken	0..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.
@ResStatus	1	To specify the type of action requested when more than one function could be handled by the message. The value should be 'Modify' .

Element @Attribute	Num	Description/Contents
@TimeStamp	0..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 OTA versioned messages, the version of the message is indicated by a decimal value.
@Target	0..1	Used to indicate whether the request is for the Test or Production system. Value can be either 'Test' or 'Production'.
OTA_HotelResNotifRQ / POS / Source	1	Provides information on the source of a request.
@TerminalID	0..1	This is the electronic address of the device from which information is entered.
OTA_HotelResNotifRQ / POS / Source / RequestorID	0..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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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).
OTA_HotelResNotifRQ / HotelReservations / HotelReservation	1	The Reservation class contains the current reservation being created or altered.
@ResStatus	1	The status of the reservation. Should be "Checked out" when checking-out a reservation.
OTA_HotelResNotifRQ / HotelReservations / HotelReservation / UniqueID	1	The booking reference for the reservation.
@Type	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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).

5.14.3 Sample Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRQ EchoToken="f4a2d799-0824-4801-9264-1f8554f8d427" ResStatus="Modify" TimeStamp="2010-02-
12T12:26:47" Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <Source TerminalID="TRM123">
      <RequestorID Type="18" ID="KSKSYS123"/>
    </Source>
  </POS>
  <HotelReservations>
    <HotelReservation ResStatus="Checked out">
      <UniqueID Type="14" ID="RES123456"/>
    </HotelReservation>
  </HotelReservations>
</OTA_HotelResNotifRQ>
```

5.14.4 Data Element Table – Response

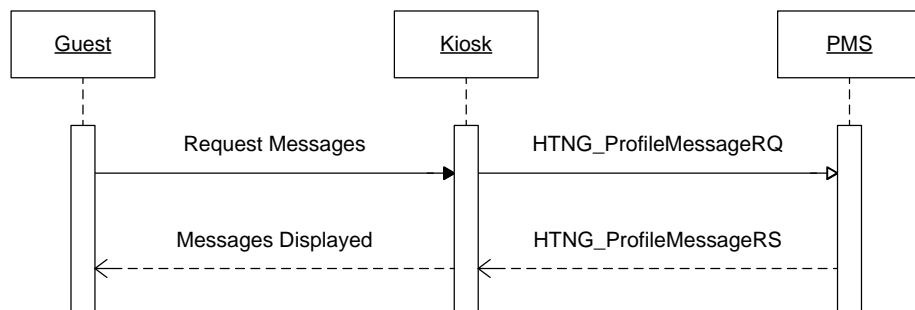
Element @Attribute	Num	Description/Contents
OTA_HotelResNotifRS	1	This message returns a list of reservations when an exact match on a read request could not be made or the request was to return a list of reservations meeting specified criteria.
@EchoToken	0..1	A reference for additional message identification, 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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
OTA_HotelResNotifRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
OTA_HotelResNotifRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
OTA_HotelResNotifRS / 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).

5.14.5 Sample Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA_HotelResNotifRS EchoToken="f4a2d799-0824-4801-9264-1f8554f8d427" TimeStamp="2010-02-12T12:26:47"
Version="1.0" xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Success/>
</OTA_HotelResNotifRS>
```


5.15 Display Guest Messages

5.15.1 Messaging Use Case



Use Case Name:	Display Guest Messages
Summary:	Kiosk displays a listing of messages left for the guest.
Basic Course of Events:	<p>The use case begins after the guest has found their booking.</p> <ol style="list-style-type: none"> 1. Guest selects the option to view their messages 2. Kiosk requests messages to display to the guest 3. PMS sends all guest messages to the kiosk 4. For each message, the kiosk displays the message to the guest 5. For each message displayed and acknowledged by the guest, the kiosk will send a change message status request to the PMS (see separate use case for Change Message Status) <p>The use case ends when the guest exits the message viewing loop or when all available messages have been displayed.</p>
Exception Path:	In step 3, there may be no messages available, the kiosk should indicate to the guest that there are no messages.
Alternative Paths:	In step 4, the kiosk could display messages of different status (read, unread, deleted) or only of one default status, based on guest input, kiosk implementation, or business rules.
Trigger:	Guest selects option to view messages.
Assumptions:	Only unread and read messages will be returned.
Preconditions:	Guest has read and unread messages attached to their reservation.
Postconditions:	Guest messages will be marked as read.

5.15.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
HTNG_ProfileMessageRQ	1	Message used to retrieve messages left for a guest staying at the hotel.
@EchoToken	0..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	0..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 OTA versioned messages, the version of the message is indicated by a decimal value.

Element @Attribute	Num	Description/Contents
@Target	0..1	Used to indicate whether the request is for the Test or Production system. Value can be either 'Test' or 'Production'.
HTNG_ProfileMessageRQ / POS / Source	1	Provides information on the source of a request.
@TerminalID	0..1	The electronic address of the device from which information is entered.
HTNG_ProfileMessageRQ / POS / Source / RequestorID	0..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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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_ProfileMessageRQ / UniqueID	1	The unique identifier element allows the trading partners to uniquely identify each (i.e., the entire message) for transaction tracability.
@Type	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.

5.15.3 Sample Message – Request

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_ProfileMessageRQ EchoToken="a" Timestamp="2010-12-17T09:30:47Z" Version="0.0" Target="Test"
xmlns="http://htng.org/2010B" xmlns:ota="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <ota:Source TerminalID="KIOSK123">
      <ota:RequestorID Type="0" ID_Context="Kiosk" ID="123"/>
    </ota:Source>
  </POS> <UniqueID Type="14" ID_Context="PMS" ID="RES134"/>
</HTNG_ProfileMessageRQ>
```

5.15.4 Data Element Table – Response

Element @Attribute	Num	Description/Contents
HTNG_ProfileMessageRS	1	Message used to retrieve messages left for a guest staying at the hotel.
@EchoToken	0..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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
HTNG_ProfileMessageRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_ProfileMessageRS /	0..1	Used in conjunction with the Success element to define one or more business

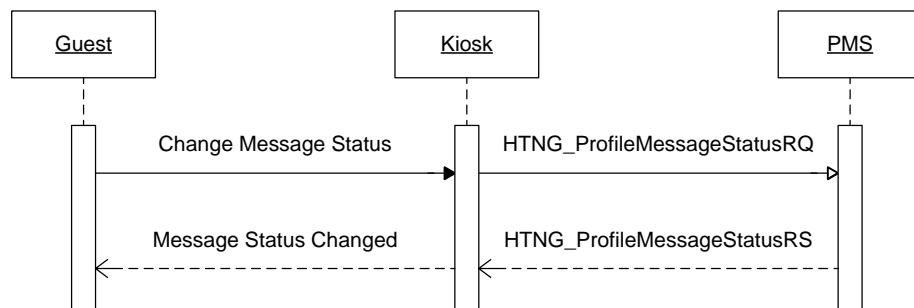
Element @Attribute	Num	Description/Contents
Warnings		errors.
HTNG_ProfileMessageRS / Warnings / Warning	1..n	Used when a message has been successfully processed to report any warnings or business errors that occurred.
@Type	1	The Warning element MUST contain the Type attribute that uses a recommended 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_ProfileMessageRS / UniqueID	1	The unique identifier element allows the trading partners to uniquely identify each (i.e., the entire message) for transaction tracability.
@Type	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_ProfileMessageRS / ProfileMessages	0..1	A collection of ProfileMessage objects.
HTNG_ProfileMessageRS / ProfileMessages / ProfileMessage	1..n	A message let for a guest staying at the hotel.
@MessageID	1	The primary key of the message record in the database.
@Status	1	The status of the message. Possible values are Read, Unread, and Deleted.
CreateDateTime	1	Time stamp of the creation of the message.
HTNG_ProfileMessageRS / ProfileMessages / Text	1	The text (body) of the message.

5.15.5 Sample Message – Response

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_ProfileMessageRS EchoToken="a" TimeStamp="2010-12-17T09:30:47Z" Version="0.0" Target="Test"
xmlns="http://htng.org/2010B" xmlns:ota="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Success/>
  <Warnings>
    <ota:Warning Type="0" Status="a" ShortText="a" Code="0">String</ota:Warning>
  </Warnings>
  <UniqueID Type="0" ID_Context="a" ID="a"/>
  <ProfileMessages>
    <ProfileMessage MessageID="123456" Status="Unread" CreateDateTime="2010-12-17T09:30:47Z">
      <ota:Text>Your boss called to say that he need for you to ensure you ...</ota:Text>
    </ProfileMessage>
    <ProfileMessage MessageID="123457" Language="en-us" Status="Unread" CreateDateTime="2010-12-
17T09:30:47Z">
      <ota:Text>Call Home</ota:Text>
    </ProfileMessage>
  </ProfileMessages>
</HTNG_ProfileMessageRS>
```

5.16 Change Guest Message Status

5.16.1 Messaging Use Case



Use Case Name:	Change Guest Message Status
Summary:	The Kiosk marks an unread message as read or the guest would like to delete a message they have already read.
Basic Course of Events:	<p>The use case begins when a guest views a message (see also separate use case Display Guest Messages).</p> <ol style="list-style-type: none"> 1. Guest Views message 2. Kiosk Sends a Change Message Status 'Read' request to the PMS 3. PMS marks the message as read <p>The use case ends when the message status has been changed.</p>
Exception Path:	None.
Alternative Paths:	<p>IN step 1, the guest may opt to delete a message</p> <ol style="list-style-type: none"> 1. Guest selects option to delete message 2. Kiosk sends a change message status 'deleted' request to the PMS 3. PMS marks the message as deleted
Trigger:	None.
Assumptions:	When marking a message as read, kiosk has ensured that the guest actually saw the message (e.g., on screen for x seconds) or that the message was printed.
Preconditions:	Guest message exists with specific status.
Postconditions:	Guest message status has changed.

5.16.2 Data Element Table – Request

Element @Attribute	Num	Description/Contents
HTNG_ProfileMessageStatusRQ	1	Message used to update the status of a message retrieved by a guest staying at the hotel.
@EchoToken	0..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	0..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 OTA versioned messages, the version of the message is indicated by a decimal value.
@Target	0..1	Used to indicate whether the request is for the Test or Production system. Value can be either 'Test' or 'Production'.

Element @Attribute	Num	Description/Contents
HTNG_ProfileMessageStatusRQ / POS / Source	1	Provides information on the source of a request.
@TerminalID	0..1	This is the electronic address of the device from which information is entered.
HTNG_ProfileMessageStatusRQ / POS / Source / RequestorID	0..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	1	A reference to the type of object defined by the UniqueID element. Refer to OTA 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_ProfileMessageStatusRQ / UniqueID	1	The unique identifier element allows the trading partners to uniquely identify each (i.e., the entire message) for transaction tracability.
@Type	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_ProfileMessageStatusRQ / ProfileMessages	1	A collection of ProfileMessage objects.
HTNG_ProfileMessageStatusRQ / ProfileMessages / ProfileMessage	1..n	A message let for a guest staying at the hotel.
@MessageID	1	The primary key of the message record in the database.
@Status	1	The status of the message. Possible values are Read, Unread, and Deleted.

5.16.3 Sample Message – Request

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_ProfileMessageStatusRQ EchoToken="a" TimeStamp="2010-12-17T09:30:47Z" Version="0.0" Target="Test"
xmlns="http://htng.org/2010B" xmlns:ota="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <POS>
    <ota:Source TerminalID="KIOSK123">
      <ota:RequestorID Type="0" ID_Context="Kiosk" ID="123"/>
    </ota:Source>
  </POS>
  <UniqueID Type="14" ID_Context="PMS" ID="RES123"/>
  <ProfileMessages>
    <ProfileMessage MessageID="1236456" Status="Read"/>
  </ProfileMessages>
</HTNG_ProfileMessageStatusRQ>
```

5.16.4 Data Element Table – Response

Element @Attribute	Num	Description/Contents
HTNG_ProfileMessageStatusRS	1	Message used to update the status of a message retrieved by a guest staying at the hotel.
@EchoToken	0..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.

Element @Attribute	Num	Description/Contents
@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.
@Target	0..1	Used to indicate whether the request is for the Test or Production system.
HTNG_ProfileMessageStatusRS / Success	0..1	The presence of the empty Success element explicitly indicates that the OpenTravel versioned message succeeded.
HTNG_ProfileMessageStatusRS / Warnings	0..1	Used in conjunction with the Success element to define one or more business errors.
HTNG_ProfileMessageStatusRS / 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).

5.16.5 Sample Message – Response

```
<?xml version="1.0" encoding="UTF-8"?>
<HTNG_ProfileMessageStatusRS EchoToken="a" TimeStamp="2001-12-17T09:30:47Z" Version="0.0" Target="Test"
xmlns="http://htng.org/2010B" xmlns:ota="http://www.opentravel.org/OTA/2003/05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Success/>
  <Warnings>
    <ota:Warning Type="0" Status="a" ShortText="a" Code="0">String</ota:Warning>
  </Warnings>
</HTNG_ProfileMessageStatusRS>
```

6 Appendix 1

6.1 Web Service Description

Web service descriptions are exposed by use of *HTNG_KioskService.wsdl*. Additional required files are imported automatically.

portType name	operation name	Request message	Response message
ARIAndReservationPush	OTA_HotelResNotifRQ	OTA_HotelResNotifRQ	OTA_HotelResNotifRS
FolioManagement	RetrieveFolio	HTNG_HotelFolioRQ	HTNG_HotelFolioRS
GuestCommunications	RetrieveMessages	HTNG_ProfileMessageRQ	HTNG_ProfileMessageRS
GuestCommunications	UpdateMessageStatus	HTNG_ProfileMessageStatusRQ	HTNG_ProfileMessageStatusRS
ReservationStayStatus	ProcessReservationCheckIn	OTA_HotelResNotifRQ	OTA_HotelResNotifRS
ReservationStayStatus	ProcessReservationCheckOut	OTA_HotelResNotifRQ	OTA_HotelResNotifRS
SeamlessShopAndBook	CheckAvailability	OTA_HotelAvailRQ	OTA_HotelAvailRS
SeamlessShopAndBook	ProcessReservationRequest	OTA_HotelResRQ	OTA_HotelResRS
SeamlessShopAndBook	RetrieveReservations	OTA_ReadRQ	OTA_ResRetrieveRS
SeamlessShopAndBook	UpsellRequest	HTNG_HotelRoomTypeUpsellRQ	HTNG_HotelRoomTypeUpsellRS
SecureAreaAccess	EncodeHotelRoomKey	HTNG_HotelKeyEncodeRQ	HTNG_HotelKeyEncodeRS
SecureAreaAccess	RequestHotelRoomKeyEncoding	HTNG_HotelKeyEncodeRQ	HTNG_HotelKeyEncodeRS
PaymentCardProcessing	ProcessPaymentCard	HTNG_PaymentCardProcessingRQ	HTNG_PaymentCardProcessingRS