Sabre Hospitality

Generic Push ARI Specification

Release 10.31.0

Copyright

Table of Contents

Revision History

Date	Rev Version	Description	Author Initials
10/27/23	v10.31.0	• Added the ability to communicate which meal plan is included in a rate in the OTA_HotelRatePlanNotifRQ message. • Added the ability to communicate in the OTA_HotelInvCountNotifRQ message the maximum occupancy values and whether an extra bed is required. • Added the ability to communicate a breakdown of taxes and fees for the adult pricing and for the child offset pricing in the OTA_HotelRatePlanNotifRQ message. • Added the ability to communicate cancel policy details when communicating product details in the OTA_HotelRatePlanNotifRQ message. • Updated the description of the Use Case: Include Hurdle Value.	JR
6/27/23	v10.29.0 Rev A	• Updated @AmountAfterTax description in the OTA_HotelRatePlanNotifRQ message.	JR
3/18/23	v10.29.0	• Added the ability when ARI hurdle values are captured during the availability check to include those values within the OTA_HotelAvailNotifRQ message. • Added the missing UniqueID element to the OTA_HotelInvCountNotifRQ message.	JR
2/7/22	v10.17.0 Rev D	• Removed @RelatesToCorrelationID from the HTNG Header Attribute Descriptions table. • Removed invalid namespace references in the HTNG Header Use Cases section.	JR
10/29/21	v10.17.0 Rev C	• Updated @RatePlanNotifType values in the OTA_HotelRatePlanNotifRQ message. • Updated OTA_HotelInvCountNotifRQ Message Description.	JR
7/8/21	v10.17.0 Rev B	• Added notes to @MinAdvancedBookingOffset and @MaxAdvancedBookingOffset in the OTA_HotelAvailNotifRQ message. • Quality review updates.	JR
5/15/21	v10.17.0 Rev A	• Updated Use Case: Synchronous Request and Use Case: Synchronous Response.	JR
7/16/20	v10.17.0	• Added the ability to flag and suspend products where setup issues result in failure messages.	JR
4/23/20	v10.16.0	• Added the ability to communicate child pricing by child age ranges in the OTA_HotelRatePlanNotifRQ message. • Added the ability to communicate child pricing by occupancy in the OTA_HotelRatePlanNotifRQ message. • Added the ability to communicate child pricing by occupancy and age ranges in the OTA_HotelRatePlanNotifRQ message.	JR
2/6/20	v10.13.0 Rev A	• Updated OTA_HotelAvailNotifRQ Message Description.	JR

Date	Rev Version	Description	Author Initials
11/9/19	v10.13.0	• Added the ability to communicate adult occupancy pricing for all of the adult occupancies supported by a room type in the OTA_HotelRatePlanNotifRQ message.	JR
4/16/19	v9.8.0 Rev G	• Removed all references to PING message.	JR
4/2/19	v9.8.0 Rev F	• Removed all references to asynchronous processing.	DLJ
10/18/18	v9.8.0 Rev E	• Removed the reference that SynXis CRS will send a once a day update.	CNO
10/3/18	v9.8.0 Rev D	• Updated the definition of the day of week (DOW) definition in the OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate feature.	CNO
9/21/17	v9.8.0 Rev C	• Removed references to the WSDL. One does not exist for this interface. • Removed references to asynchronous behavior. Only synchronous behavior is supported. • Corrected the namespaces in the examples in Chapter 3, HTNG Header Use Cases from soap2: to soap.	CNO
9/11/17	v9.8.0 Rev B	• Updated the type of MinAdvancedBookingOffset and MaxAdvancedBookingOffset to duration.	CNO
8/19/17	v9.8.0 Rev A	• Added a note that the ARI messages cannot be used with Booking Engine.	CNO
8/19/17	v9.8.0	• The ability to distribute restrictions to online travel agents.	CNO
6/22/17	v7.2.0 Rev B	• Updated version value in OTA_HotelInvCountNotifRQ, OTA_HotelAvailNotifRQ, and OTA_HotelRatePlanNotifRQ messages.	CNO
2/9/17	v7.2.0 Rev B	• Updated minor issues.	CNO
8/25/16	v7.2.0 Rev A	• Updated copyright statement.	CNO

About This Guide

Welcome to the Generic ARI Specification Guide. This guide contains message examples as well as element and attribute definitions for the Generic Availability, Rates, and Inventory interface. The following message types are defined.

HTNG Headers

- Synchronous Request
- Synchronous Response

OTA_HotelInvCountNotifRQ/RS: Update Inventory

- OTA_HotelInvCountNotifRQ
- OTA_HotelInvCountNotifRS

OTA_HotelAvailNotifRQ/RS: Update Restrictions

- OTA_HotelAvailNotifRQ
- OTA_HotelAvailNotifRS

OTA_HotelRatePlanNotifRQ/RS: Update Rate Amounts

- OTA_HotelRatePlanNotifRQ
- OTA_HotelRatePlanNotifRS

Soap Faults: Synchronous Communication Errors

- Soap Fault Response Message

Note: The Generic Push ARI functionality is **<u>not</u>** available to Booking Engines.

1 Overview

This documentation assumes advanced knowledge of HTTP, SOAP, web services, XML, and the client application platform/programming that uses these web services.

XML is expected to be Extensible; application expects XML message format will work even if new data is added. It is not a static, fixed file format. Hardcoding must not be used since additional elements may be added as the system evolves – e.g. TPA Extensions or sub-elements not currently in use.

For further information on basic concepts and technologies required for web services implementation, refer to the W3C website: SOAP (https://www.w3.org/TR/soap12-part0/), Web Services (http://www.w3.org/2002/ws/), and to the documentation for your development platform.

This interface description is provided by Sabre Hospitality (SH) for use by third parties to develop customized client systems that receive data from SynXis CRS. This interface is developed based on HTNG 2010A specification, using SOAP Version 1.2.

As availability and rates change in the SynXis CRS system, SynXis CRS sends ARI Upload messages to a client system via HTTPS Post. The messages are contained within a SOAP envelope (<Envelope xmlns="http://www.w3.org/2003/05/soap-envelope">) and follow the HTNG 2010A specifications. All messages must contain an HTNG approved Soap Header that complies with WS Security (authentication only) and WS Addressing standards. Rate and Restriction uploads are sent at the product level. Inventory uploads are sent at the room type level. Messages are grouped by hotel and channel.

Messages are sent synchronously. In the synchronous workflow, a message is processed by the client system and a synchronous response is returned.

Note: The Generic Push ARI functionality is **<u>not</u>** available to Booking Engines.

1.1 Description of Interface

1.1.1 *Channel, URI, and Product Assignments*

For ARI to be pushed out for a hotel, ARI channels must be assigned to the hotel in the SynXis CRS. Each channel has a default switch or endpoint URI per message type assigned but a hotel may select a different switch or endpoint URI to communicate ARI to. ARI will be distributed for products assigned to the hotel's channels.

1.1.2 *Switch or Endpoint Configurations*

Several configurations are made for each client system that is certified to receive ARI updates. These configurations are agreed upon and set as part of the certification process. The configurations for a client system include items such as contact email addresses, URI address by request type, login username/password, connection timeout, maximum number of communication failures, maximum number of connections, maximum number of re-attempts, maximum callback timespan, and time between re-attempts.

1.1.3 *Length of Stay Based ARI*

Updates for a hotel that are sent to a channel can be configured to receive ARI data by stay date or by arrival date/length of stay.

Generic ARI supports Length of Stay (LOS) pricing for certified OTAs. Arrival date/LOS based rate amounts can be communicated to the applicable OTAs. LOS Pricing supports up to 8 nights LOS. Hotels must be configured to deliver LOS pricing for each supported channel. By default, Day of Week pricing is sent.

Generic ARI provides support for Full Pattern Length of Stay (FPLOS) restrictions and includes FPLOS updates, which are Closed to Arrival by Length of Stay. FPLOS is used in conjunction with LOS Pricing and with the existing restrictions that are currently supported for a channel, not as a replacement. An FPLOS availability restriction update is included for a product/arrival date.

Note: Certification of LOS ARI is required prior to using this functionality.

1.1.4 *Manual Re-Status*

SH support staff can send a full update of inventory, availability, and rates. This is done by hotel, product, channel, and date range. A manual re-status can synchronize systems if the data is out of synch or a new property, product, or client system is initiated.

1.1.5 *Re-Attempts*

SynXis CRS re-attempts to send messages that were not received by a client system due to communication issues. Re-attempts are also done if a recoverable error is returned by the client system. SynXis CRS discontinues re-attempting to send a message when it is determined that the message cannot be processed or communication with a client system is not possible.

1.1.6 *System Downtime*

Client systems should notify SH of scheduled or unscheduled downtimes by contacting SH support staff. When a system downtime is reported, the ARI process for a client system is disabled and resumes when the client system is back up.

1.1.7 *Message Logging*

SynXis CRS logs all incoming and outgoing messages.

1.1.8 *Email Notifications*

If a problem in sending ARI to a client system has been detected, SH may send out email notifications to client system contact personnel and may disable the ARI process for a channel or switch/endpoint if necessary.

1.1.9 *Errors*

SynXis CRS supports HTNG approved errors, that report errors in the OTA_HotelInvCountNotifRS, OTA_HotelAvailNotifRS, and OTA_HotelRatePlanNotifRS response messages. Refer to the Error List in the Appendix. Only approved errors should be included in responses sent to SynXis CRS, and proper error handling is required.

1.1.10 *ARI Suspension*

When an ARI Availability or Rate message encounters a failure with OTA, the offending product in the message is flagged in the SynXis CRS database, provided that a given error code is marked as causing suspension. As a result, any further ARI is suspended and no longer generated for that product.

Hoteliers can review the list of suspended products on ARI in SynXis CRS to resolve the issue and release the products from suspension.

Note: This feature applies to those switches that have the "Suspend Product" option enabled. Disabling the "Suspend Product" option on a given switch does not automatically release the previously suspended products. Each suspended product needs to be released manually in SynXis CRS.

1.1.11 *Certification*

Every third party that wants to utilize the ARI interface provided in this document is required to go through a certification process. Once a switch/endpoint is certified for the ARI interface, it will be enabled for ARI updates.

1.1.12 *Synchronous Workflow*

SynXis CRS	Client System
Request	
	Message Processing.
	Soap Fault Response (for communication or message errors) or NotifRS Response (success or error) to indicate results of processing request.

1.1.13   *Message Descriptions*

Following are the messages defined in this interface:

- OTA_HotelInvCountNotifRQ sends inventory data updates.

- OTA_HotelAvailNotifRQ sends updates for availability data.

- OTA_HotelRatePlanNotifRQ sends updates for rate data. Where possible, messages are grouped, to send as few messages as possible. Each message is consolidated by hotel, channel, and message type. Availability and rate messages are consolidated by product. Each message contains the date ranges to update.

- Client systems using the synchronous workflow respond to SynXis CRS requests with the OTA_HotelInvCountNotifRS, OTA_HotelAvailNotifRS, and OTA_HotelRatePlanNotifRS messages. These messages update SynXis CRS with the results of processing the SynXis CRS request messages. If a failure occurs before a request message is processed, a Soap Fault response is returned to indicate the reason for failure.

2 HTNG Headers

This section describes the SOAP headers used with ARI interface.

2.1 HTNG Header Description

Every message in this interface requires an HTNG 2.X Soap header that adheres to WS Addressing and WS Security requirements. WS Addressing communicates message IDs between systems and address destinations. WS Security is for user authentication only. In-message signatures and message encryption are not supported.

The elements used and data in the HTNG Soap header depends upon which part of the synchronous communication process it is used for. See the HTNG Header Attribute Descriptions for an explanation of the field requirements.

2.2 HTNG Header Use Cases

2.2.1 *Use Case: Synchronous Request*

```
<Envelope xmlns="http://www.w3.org/2003/05/soap-envelope">
  <soap2:Header xmlns:soap2="http://www.w3.org/2003/05/soap-envelope" xmlns:htng="http://htng.org/1.3/Header"
  xmlns:wsa="http://www.w3.org/2005/08/addressing"
  xmlns:wss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:htnga="http://htng.org/PWSWG/2007/02/AsyncHeaders"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <wsa:Action>selected recipient action</wsa:Action>
    <wsa:ReplyTo>
      <wsa:Address>http://www.w3.org/2005/08/addressing/role/anonymous</wsa:Address>
    </wsa:ReplyTo>
    <wss:Security mustUnderstand="1">
      <wss:UsernameToken>
        <wss:Username>*****</wss:Username>
        <wss:Password>*******</wss:Password>
      </wss:UsernameToken>
    </wss:Security>
    <wsa:MessageID>27d613d9-50e0-452f-8a6c-ea0668fd3c04</wsa:MessageID>
    <wsa:To>recipientendpoint</wsa:To>
  </soap2:Header>
  <Body>
  …
  </Body>
</Envelope>
```

2.2.2 *Use Case: Synchronous Response*

```xml
<Envelope xmlns="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <wsa:MessageID>6210e5db-985a-11eb-a447-f8b46ab1dd50</wsa:MessageID>
    <wsa:RelatesTo>27d613d9-50e0-452f-8a6c-ea0668fd3c04</wsa:RelatesTo>
    <wsa:To>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:To>
    <wsa:Action>https://synxis/ARI_Synch_Service/ARI_Synch</wsa:Action>
    <wsa:ReplyTo>
      <wsa:Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:Address>
    </wsa:ReplyTo>
  </soap:Header>
  <Body>
  …
  </Body>
</Envelope>
```

2.3 HTNG Header Attribute Descriptions

Element/Attribute	Description / XPath	Type	Required?
MessageID	Unique ID for the message. Each message must have a new MessageID. **Parent XPath:** /Header	String	Yes
RelatesTo	UniqueID for the message which ties a response message back to the request. The value for this attribute should always be the MessageID from the request message. *This attribute is used in the synchronous response message. **Parent XPath:** /Header	String	*Yes
To	Synchronous requests contain the address to which the message is to be posted. Synchronous responses pass back the wsa:ReplyTo value from the request message. **Parent XPath:** /Header	String	Yes
Action	The requested action for this message. Synchronous responses echo back the request value. **Parent XPath:** /Header	String	Yes
ReplyTo			
Address	Synchronous requests contain the wsa:ReplyTo element with the WS addressing anonymous URI. The anonymous URI signifies that the responses are provided on the same connection synchronously. **Parent XPath:** /Header/ReplyTo	String	Yes
Security	WS Security elements are required only in request messages to authenticate into the system the message is sent to. Synchronous response messages do not require authentication.		
@mustUnderstand	The @mustUnderstand attribute indicates to the recipient of a SOAP message whether processing of an extension element is mandatory. If the attribute has the value "1", then the recipient must recognize the extension element and process it accordingly. If the recipient does not recognize the element, it must report a fault. This value is always "1" (true). **Parent XPath:** /Header/Security	Boolean	Yes
UsernameToken			
Username	Username credential for the system that receives the message. **Parent XPath:** /Header/Security/UsernameToken	String	Yes

Element/Attribute	Description / XPath	Type	Required?
Password	Password credential for the system that receives the message. **Parent XPath:** /Header/Security/UsernameToken	String	Yes

3 OTA_HotelInvCountNotifRQ/RS: Update Inventory

This section provides the message examples used to communicate room level inventory values to client systems.

3.1 OTA_HotelInvCountNotifRQ Message Description

- The OTA_HotelInvCountNotifRQ message is used to communicate room level inventory values to client systems.
- Each OTA_HotelInvCountNotifRQ message contains inventory for one room type for one hotel for one channel.
- Each Inventory element contains physical inventory counts for one room type for a date range.
- Each InvCounts element contains two InvCount elements. One to communicate the Left To Sell inventory value and one to communicate the Can Sell Up To inventory value.
- Room inventories with the same value are consolidated into date ranges.
- Room occupancy information is added to the room inventory updates if a given ARI switch is configured to have room occupancy information communicated.
  Note: Room occupancy information will be included along with the room inventory changes when they are sent out. When only room occupancy values change, the room inventory updates are not sent out.

3.2 OTA_HotelInvCountNotifRQ/RS Use Cases

3.2.1 *Use Case: Update room type inventory values for multiple room types and dates*

```xml
<OTA_HotelInvCountNotifRQ xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelInvCountNotifRQ.xsd" EchoToken=""
TimeStamp="2015-10-01T09:41:51.982" Version="2.000">
  <UniqueID>
  </UniqueID>
  <Inventories HotelCode="10001">
    <Inventory>
      <StatusApplicationControl Start="2015-01-01" End="2015-02-28" InvTypeCode="King">
        <DestinationSystemCodes>
          <DestinationSystemCode>330</DestinationSystemCode>
        </DestinationSystemCodes>
      </StatusApplicationControl>
      <InvCounts>
        <InvCount CountType="2" Count="8"/>
        <InvCount CountType="3" Count="10"/>
      </InvCounts>
    </Inventory>
    <Inventory>
      <StatusApplicationControl Start="2015-03-01" End="2015-03-31" InvTypeCode="King">
        <DestinationSystemCodes>
          <DestinationSystemCode>330</DestinationSystemCode>
        </DestinationSystemCodes>
      </StatusApplicationControl>
      <InvCounts>
        <InvCount CountType="2" Count="5"/>
        <InvCount CountType="3" Count="5"/>
      </InvCounts>
    </Inventory>
    <Inventory>
      <StatusApplicationControl Start="2015-01-01" End="2015-01-01" InvTypeCode="Queen">
        <DestinationSystemCodes>
          <DestinationSystemCode>330</DestinationSystemCode>
        </DestinationSystemCodes>
      </StatusApplicationControl>
      <InvCounts>
        <InvCount CountType="2" Count="2"/>
        <InvCount CountType="3" Count="7"/>
      </InvCounts>
    </Inventory>
    <Inventory>
      <StatusApplicationControl Start="2015-01-02" End="2015-01-31" InvTypeCode="Queen">
        <DestinationSystemCodes>
```

```xml
        <DestinationSystemCode>330</DestinationSystemCode>
      </DestinationSystemCodes>
    </StatusApplicationControl>
    <InvCounts>
      <InvCount CountType="2" Count="4"/>
      <InvCount CountType="3" Count="6"/>
    </InvCounts>
  </Inventory>
  <UniqueID Type="16">
  </UniqueID>
 </Inventories>
</OTA_HotelInvCountNotifRQ>
```

3.2.2 *Use Case: Include Room Occupancy Information*

```xml
<OTA_HotelInvCountNotifRQ…>
 <UniqueID>
 </UniqueID>
 <Inventories…>
 …
 </Inventories>
 <TPA_Extensions>
   <GuestRooms>
     <GuestRoom InvTypeCode="ROOM1" MaxRollaways="0">
       <Occupancy MaxOccupancy="2">
       <Occupancy MaxOccupancy="1" AgeQualifyingCode="10">
       <Occupancy MaxOccupancy="1" AgeQualifyingCode="8">
     </GuestRoom>
     <GuestRoom InvTypeCode="ROOM2" MaxRollaways="1">
       <Occupancy MaxOccupancy="3">
       <Occupancy MaxOccupancy="2" AgeQualifyingCode="10">
       <Occupancy MaxOccupancy="2" AgeQualifyingCode="8">
     </GuestRoom>
     <GuestRoom InvTypeCode="ROOM3" MaxRollaways="1">
       <Occupancy MaxOccupancy="5">
       <Occupancy MaxOccupancy="3" AgeQualifyingCode="10">
       <Occupancy MaxOccupancy="3" AgeQualifyingCode="8">
     </GuestRoom>
     <GuestRoom InvTypeCode="ROOM4" MaxRollaways="1">
       <Occupancy MaxOccupancy="2" AgeQualifyingCode="10">
     </GuestRoom>
     <GuestRoom InvTypeCode="ROOM5" MaxRollaways="1">
       <Occupancy MaxOccupancy="3">
     </GuestRoom>
     <GuestRoom InvTypeCode="ROOM6" MaxRollaways="1">
       <Occupancy MaxOccupancy="4">
       <Occupancy MaxOccupancy="2" AgeQualifyingCode="10">
```

```
          <Occupancy MaxOccupancy="3" AgeQualifyingCode="8">
          <Occupancy MaxOccupancy="2" AgeQualifyingCode="8" MinAge="0" MaxAge="2">
          <Occupancy MaxOccupancy="2" AgeQualifyingCode="8" MinAge="3" MaxAge="10">
          <Occupancy MaxOccupancy="1" AgeQualifyingCode="8" MinAge="11" MaxAge="17">
        </GuestRoom>
        <GuestRoom InvTypeCode="ROOM7" MaxRollaways="1">
          <Occupancy MaxOccupancy="4">
          <Occupancy MaxOccupancy="2" AgeQualifyingCode="10">
          <Occupancy MaxOccupancy="0" AgeQualifyingCode="8" MinAge="3" MaxAge="10">
          <Occupancy MaxOccupancy="1" AgeQualifyingCode="8" MinAge="11" MaxAge="17">
        </GuestRoom>
      </GuestRooms>
    </TPA_Extensions>
</OTA_HotelInvCountNotifRQ>
```

3.2.3 *Use Case: Successful response*

Successful responses contain a Success element and are returned when a request message is successfully processed.

```
<OTA_HotelInvCountNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelInvCountNotifRS.xsd" TimeStamp="2015-10-01T09:41:59.982"
Version="2.000">
  <Success/>
</OTA_HotelInvCountNotifRS>
```

3.2.4 *Use Case: Successful response with warnings*

Successful responses may also contain Warnings if the request message was successfully processed but had nonfatal problems. The Warning Type and optional Code indicate the cause of the warning and are selected from a list of approved HTNG codes (See Appendix). The warning description provides additional details (if possible) on the cause of the problem.

```
<OTA_HotelInvCountNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelInvCountNotifRS.xsd" TimeStamp="2015-10-01T09:41:59.982"
Version="2.000">
  <Success/>
  <Warnings>
    <Warning Type="3" Code="320">Description of warning</Warning>
  </Warnings>
</OTA_HotelInvCountNotifRS>
```

3.2.5 *Use Case: Error response*

Request messages that could not be processed due to fatal problems result in Error responses. The Error Type and optional Code indicate the cause of the warning and must be selected from a list of approved HTNG codes (See Appendix). The error description provides additional details (if possible) on the cause of the problem.

```xml
<OTA_HotelInvCountNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelInvCountNotifRS.xsd" TimeStamp="2015-10-01T09:41:59.982"
Version="2.000">
  <Errors>
    <Error Type="3" Code="402">Invalid room type(ZZZ)</Error>
  </Errors>
</OTA_HotelInvCountNotifRS>
```

3.3 OTA_HotelInvCountNotifRQ Attribute Descriptions

Element/Attribute OTA_HotelInvCountNotifRQ	Description / XPath	Type
@TimeStamp	The instant of message creation. **Parent XPath:** /OTA_HotelInvCountNotifRQ	DateTime
@EchoToken	Simple string returned in the response as sent in the request. It can be used to match requests with responses to simulate session state. **Parent XPath:** /OTA_HotelInvCountNotifRQ	String
@Version	Mandatory OTA attribute. This value should be "2.000". **Parent XPath:** /OTA_HotelInvCountNotifRQ	Decimal
UniqueID	An empty UniqueID element is included. **Parent XPath:** /OTA_HotelInvCountNotifRQ	
Inventories		
@HotelCode	The code that uniquely identifies a single hotel property. This corresponds to the SynXis CRS Hotel ID value. **Parent XPath:** /OTA_HotelInvCountNotifRQ/Inventories	String
Inventory		
StatusApplicationControl		
@Start	The start date of the availability update. **Parent XPath:** /OTA_HotelInvCountNotifRQ/Inventories/Inventory/StatusApplicationControl	DateTime
@End	The end date of the availability update. **Parent XPath:** /OTA_HotelInvCountNotifRQ/Inventories/Inventory/StatusApplicationControl	DateTime

Element/Attribute OTA_HotelInvCountNotifRQ	Description / XPath	Type
@InvTypeCode	Code that identifies a single hotel room type. This corresponds to the SynXis CRS Room Code value. **Parent XPath:** /OTA_HotelInvCountNotifRQ/Inventories/Inventory/StatusApplicationControl	String
DestinationSystemCodes		
DestinationSystemCode	The destination system code defines a system to which information is provided. This corresponds to the SynXis CRS Channel Code value. All updates in a message are for the same channel that is defined by this code. **Parent XPath:** /OTA_HotelInvCountNotifRQ/Inventories/Inventory/StatusApplicationControl/DestinationSystemCodes	String
InvCounts		
InvCount		
@CountType	Type of inventory count reported. Refer to OTA Code List Inventory Count Type (INV). **Valid values:** "2" – (Definitive availability) Left To Sell– Number of rooms left to be sold. This value considers the number of rooms for the room type and channel that SynXis CRS is aware of having been sold. Channels using this value must send SynXis CRS reservation notification messages to keep the sold quantities in the SynXis CRS system updated or have integration with a PMS that is system of record for inventory and sends SynXis CRS inventory values. "3" – (Tentative availability) Can Sell Up To – Number of rooms that the hotel has set up to sell for the channel. Channels using this value must keep track of the number of rooms they have sold to know when the limit has been reached. **Parent XPath:** /OTA_HotelInvCountNotifRQ/Inventories/Inventory/InvCounts/InvCount	String
@Count	Number of rooms per CountType. **Parent XPath:** /OTA_HotelInvCountNotifRQ/Inventories/Inventory/InvCounts/InvCount	Integer
UniqueID		

Element/Attribute OTA_HotelInvCountNotifRQ	Description / XPath	Type
@Type	Refers to OTA code list UIT. "16" (Reference) is always sent. **Parent XPath:** /OTA_HotelInvCountNotifRQ/Inventories/UniqueID	Integer
TPA_Extensions		
GuestRooms		
GuestRoom	A GuestRoom element is included for each room type which is assigned to <u>one or more</u> products which are distributed to the OTA. **Parent XPath:** /OTA_HotelInvCountNotifRQ/TPA_Extensions/GuestRooms	
@InvTypeCode	Code for the room type. **Parent XPath:** /OTA_HotelInvCountNotifRQ/TPA_Extensions/GuestRooms/GuestRoom	String
@MaxRollaways	Indicates if the Allow Extra Bed configuration is enabled for the room type. **Valid Values:** "0" if the Allow Extra Bed configuration is <u>not</u> enabled for the room type. "1" if the Allow Extra Bed configuration is enabled for the room type. **Parent XPath:** /OTA_HotelInvCountNotifRQ/TPA_Extensions/GuestRooms/GuestRoom	Integer
Occupancy	Communicates the maximum occupancy for each age group (Adult/Child), if a value is set for a room type, in the @MaxOccupancy attribute along with the @AgeQualifyingCode attribute, AND/OR the total room occupancy in the @MaxOccupancy attribute, if a value is set for a room type. **Parent XPath:** /OTA_HotelInvCountNotifRQ/TPA_Extensions/GuestRooms/GuestRoom	

Element/Attribute OTA_HotelInvCountNotifRQ	Description / XPath	Type
@MaxOccupancy	The maximum number of guests (Adult / Child / Total) allowed in a room type. When Child Limits by Age Range option is enabled and one or more child age ranges have a value other than No Limit, the minimum and maximum age for the child age range is specified and the @MaxOccupancy attribute communicates the child age range occupancy value. If Allow Extra Bed configuration is enabled, then it adds +1 to the @MaxOccupancy value. **Note:** Child limits by age range are not adjusted due to the extra bed configuration so the configured value is communicated as is. **Valid Values** for Total Occupancy (used without the @AgeQualifyingCode): • If Total Guests Per Room is limited, then the Total Guests Per Room value. • If Total Guests Per Room is unlimited, then the Adult Occupancy value plus Child Occupancy value. **Note:** If Child Occupancy is unlimited in this case, then there is no Total Occupancy limit and the only occupancy limit is Adult Occupancy. **Valid Values** for Adult Occupancy (used with @AgeQualifyingCode="10"): • If Adult Occupancy is limited, then the Adult Occupancy value. • If Adult Occupancy is unlimited, there is no maximum Adult Occupancy limit, therefore this value is not communicated. **Valid Values** for Child Occupancy (used with @AgeQualifyingCode="8"): • If Child Occupancy is limited, then the Child Occupancy value. • If Child Occupancy is unlimited, there is no maximum Child Occupancy limit, therefore this value is not communicated. **Parent XPath:** /OTA_HotelInvCountNotifRQ/TPA_Extensions/GuestRooms/GuestRoom/Occupancy	Integer

Element/Attribute OTA_HotelInvCountNotifRQ	Description / XPath	Type
@AgeQualifyingCode	An age category. **Note:** In the case of total room occupancy, the @AgeQualifyingCode attribute is not included in the Occupancy element. **Valid Values:** • "10" – Adult. • "8" – Child. **Parent XPath:** /OTA_HotelInvCountNotifRQ/TPA_Extensions/GuestRooms/GuestRoom/Occupancy	Integer
@MinAge	Minimum age for child age range. **Parent XPath:** /OTA_HotelInvCountNotifRQ/TPA_Extensions/GuestRooms/GuestRoom/Occupancy	Integer
@MaxAge	Maximum age for child age range. **Parent XPath:** /OTA_HotelInvCountNotifRQ/TPA_Extensions/GuestRooms/GuestRoom/Occupancy	Integer

3.4 OTA_HotelInvCountNotifRS Attribute Descriptions

Element/Attribute OTA_HotelInvCountNotifRS	Description / XPath	Type	Required
@TimeStamp	The instant of message creation. **Parent XPath:** /OTA_HotelInvCountNotifRS	DateTime	Yes
@EchoToken	Simple string returned verbatim in the response as passed in the request. Can be used to match requests with responses to simulate session state. *Required if a value is passed in the request. **Parent XPath:** /OTA_HotelInvCountNotifRS	String	*Yes
@Version	Mandatory OTA attribute. **Parent XPath:** /OTA_HotelInvCountNotifRS	Decimal	Yes
Success	If included, this element indicates that the request message was successfully processed. *Either a Success element or Errors/Error element(s) must be included in every response. **Parent XPath:** /OTA_HotelInvCountNotifRS		*Yes
Warnings	Warnings indicate issues which did not prevent processing from succeeding.		
Warning	Description of cause for a non-fatal problem during request message processing. *At least one Warning element is required if a Warnings element is included. **Parent XPath:** /OTA_HotelInvCountNotifRS/Warnings	String	*Yes
@Type	This is an enumeration of HTNG approved error types (OTA code list EWT). See Appendix for supported values. *This is required if a Warning element is included. **Parent XPath:** /OTA_HotelInvCountNotifRS/Warnings/Warning	Integer	*Yes

Element/Attribute OTA_HotelInvCountNotifRS	Description / XPath	Type	Required
@Code	This is an enumeration of HTNG approved error codes (OTA code list ERR). See Appendix for supported values. **Parent XPath:** /OTA_HotelInvCountNotifRS/Warnings/Warning	String	
Errors	*Either a Success element or Errors/Error element(s) must be included in every response.		*Yes
Error	Description of cause for a fatal problem during request message processing. *At least one Error element is required if an Errors element is included. **Parent XPath:** /OTA_HotelInvCountNotifRS/Errors	String	*Yes
@Type	This is an enumeration of HTNG approved error types (OTA code list EWT). See Appendix for supported values. *This is required if an Error element is included. **Parent XPath:** /OTA_HotelInvCountNotifRS/Errors/Error	Integer	*Yes
@Code	This is an enumeration of HTNG approved error codes (OTA code list ERR). See Appendix for supported values. **Parent XPath:** /OTA_HotelInvCountNotifRS/Errors/Error	String	

4 OTA_HotelAvailNotifRQ/RS: Update Restrictions

This section provides the message examples used to communicate product level availability restrictions to client systems.

4.1 OTA_HotelAvailNotifRQ Message Description

- The OTA_HotelAvailNotifRQ message communicates product level availability restriction values to client systems. The restriction types that are supported are Open, Close, Close to Arrival, Close to Departure, Min and Max Length of Stay, Min and Max Length of Stay Thru, and Min and Max Lead Days.

- Each OTA_HotelAvailNotifRQ message contains restrictions for one hotel and product.

- Each AvailStatusMessage element contains one restriction type. Min and Max Length of Stay are grouped into one AvailStatusMessage element. Min and Max Length of Stay Thru are grouped into one AvailStatusMessage element. Min and Max Lead Days are grouped into one AvailStatusMessage element.

- When Length of Stay (LOS) based ARI is in use, stay restrictions are provided as usual. Additionally, Full Pattern Length of Stay (FPLOS) updates, which are Closed to Arrival by LOS are included.

- Restrictions with the same value are consolidated into date ranges.

- Each AvailStatusMessage element contains a counter that is used for error handling to determine which update(s) in a message failed.

- There is a hierarchy of restriction levels (@RestrictionStatus\Restriction value) which must be adhered to. For example, if a Master Close is sent for a product via the OTA_HotelAvailNotifRQ for specific date(s), and other Restriction types (Arrival, Departure) have varying values (Open for example) affecting the same date(s), those Open for Arrival/Departure should not be interpreted as making the product available on the said date(s). The Master Close (or Open) has priority over the Arrival and Departure Restriction types.

- Additionally, all previously existing values for a given Restriction Type (Master, Arrival, Departure) must be retained, if a new update is received for given date(s) which may not contain an update to one or more other Restriction Types. For example:
  - An OTA_HotelAvailNotifRQ is sent by SynXis CRS, containing multiple restriction values for varying dates. In that message there is a Master Close on 10/12/2030 and a Close to Arrival on 10/12/2030.
  - Another OTA_HotelAvailNotifRQ is sent by SynXis CRS, also containing multiple restriction values for varying dates. In that message there is a Master Open for 10/12/2030 and no data present in the XML for the same date for Restriction type Arrival.
    The original Close to Arrival must be retained, and the original product level restriction (Master Close) should be reversed to Open.

- When an OTA Partner Switch Configuration is setup to communicate hurdles (hurdle attribute set to Enabled), and is not using LOS/FPLOS pricing, the hurdle value is included within ARI messaging sent to the OTA Partners. If hurdle attribute is set to Disabled and/or LOS/FPLOS pricing is used, then hurdles are communicated via a Close restriction if the product fails to clear the hurdle.
  - An attribute to Enable or Disable hurdles for the target switch is available with a default setting of Disabled.
  - OTA messaging allows for a single hurdle value to be included, thus the hurdle is evaluated on single occupancy only.
  - With **Yield As** setup for the target product, the availability check calculates a hurdle value by determining the difference between the Yield As price and the associated hurdle value and applies that difference to the target product's price to generate an Implied Hurdle value, which is communicated via ARI to the target OTA Partner.
  - When the calculation between the Yield As price and the hurdle results in a negative difference, which when applied to the target product's price, causes the Implied Hurdle value to be less than 0 (zero), then the hurdle value captured will be 0 instead.
    Note: When the Yield As calculation is being determined for ARI Hurdles, and Margin Management applies to the target OTA Channel, the Margin Management adjustment is not included within the Yield As calculation to determine the hurdle value to be sent for the target rate.

4.2 OTA_HotelAvailNotifRQ/RS Use Cases

4.2.1 *Use Case: Update multiple types of restrictions for a product*

```xml
<OTA_HotelAvailNotifRQ xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelAvailNotifRQ.xsd" EchoToken="" TimeStamp="2015-10-01T09:41:51.982"
Version="6.00" MessageContentCode="3">
  <AvailStatusMessages HotelCode="10001">
    <AvailStatusMessage>
      <UniqueID Type="16" ID="1"/>
      <StatusApplicationControl Start="2015-01-01" End="2015-02-28" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true" Weds="true"
      Thur="true" Fri="true" Sat="true" Sun="true">
        <DestinationSystemCodes>
          <DestinationSystemCode>330</DestinationSystemCode>
        </DestinationSystemCodes>
      </StatusApplicationControl>
      <RestrictionStatus Restriction="Master" Status="Open"/>
    </AvailStatusMessage>
    <AvailStatusMessage>
      <UniqueID Type="16" ID="2"/>
      <StatusApplicationControl Start="2015-03-01" End="2015-03-31" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true" Weds="true"
      Thur="true" Fri="true" Sat="true" Sun="true">
        <DestinationSystemCodes>
          <DestinationSystemCode>330</DestinationSystemCode>
        </DestinationSystemCodes>
      </StatusApplicationControl>
      <RestrictionStatus Restriction="Arrival" Status="Close"/>
    </AvailStatusMessage>
    <AvailStatusMessage>
      <UniqueID Type="16" ID="3"/>
      <StatusApplicationControl Start="2015-04-01" End="2015-04-30" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true" Weds="true"
      Thur="true" Fri="true" Sat="true" Sun="true">
        <DestinationSystemCodes>
          <DestinationSystemCode>330</DestinationSystemCode>
        </DestinationSystemCodes>
      </StatusApplicationControl>
      <RestrictionStatus Restriction="Departure" Status="Close"/>
    </AvailStatusMessage>
    <AvailStatusMessage>
      <UniqueID Type="16" ID="4" />
      <StatusApplicationControl Start="2015-05-01" End="2015-12-31" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true" Weds="true"
      Thur="true" Fri="true" Sat="true" Sun="true">
        <DestinationSystemCodes>
          <DestinationSystemCode>330</DestinationSystemCode>
        </DestinationSystemCodes>
      </StatusApplicationControl>
      <RestrictionStatus Restriction="Master" Status="Close"/>
    </AvailStatusMessage>
    <AvailStatusMessage>
```

```
        <UniqueID Type="16" ID="5"/>
        <StatusApplicationControl Start="2015-01-01" End="2015-01-31" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true" Weds="true"
        Thur="true" Fri="true" Sat="true" Sun="true">
          <DestinationSystemCodes>
            <DestinationSystemCode>330</DestinationSystemCode>
          </DestinationSystemCodes>
        </StatusApplicationControl>
        <LengthsOfStay>
          <LengthOfStay MinMaxMessageType="SetMinLOS" Time="2" TimeUnit="Day"/>
          <LengthOfStay MinMaxMessageType="SetMaxLOS" Time="5" TimeUnit="Day"/>
        </LengthsOfStay>
      </AvailStatusMessage>
      <AvailStatusMessage>
        <UniqueID Type="16" ID="6"/>
        <StatusApplicationControl Start="2015-02-01" End="2015-02-28" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true" Weds="true"
        Thur="true" Fri="true" Sat="true" Sun="true">
          <DestinationSystemCodes>
            <DestinationSystemCode>330</DestinationSystemCode>
          </DestinationSystemCodes>
        </StatusApplicationControl>
        <LengthsOfStay>
          <LengthOfStay MinMaxMessageType="SetForwardMinStay" Time="3" TimeUnit="Day"/>
          <LengthOfStay MinMaxMessageType="SetForwardMaxStay" Time="6" TimeUnit="Day"/>
        </LengthsOfStay>
      </AvailStatusMessage>
    </AvailStatusMessages>
</OTA_HotelAvailNotifRQ>
```

4.2.2 *Use Case: Closed to Arrival by LOS stay restrictions (FPLOS)*

Generic ARI provides support for Full Pattern Length of Stay (FPLOS) restrictions and includes FPLOS updates, which are Closed to Arrival by Length of Stay. FPLOS is used in conjunction with LOS Pricing and with the existing restrictions that are currently supported for a channel and not as a replacement. An FPLOS availability restriction update is included for a product/arrival date.

Note: A hotel must be enabled for LOS Pricing on a per-channel basis.

The FPLOS value passed in the LengthOfStay element is a string that includes a series of Boolean (Y or N) values. The number of values included is specified by the FixedPatternLength value, which is currently 8. The first Boolean value specifies if Length of Stay 1 is open or closed to arrival, the second Boolean value specifies if length of stay 2 is open or closed to arrival, and so on. A value of Y indicates that the corresponding Length of Stay is open to arrival. A value of N indicates that the corresponding length of stay is closed to arrival.

The following message example shows how you could set an FPLOS restriction.

```
<AvailStatusMessage>
  <StatusApplicationControl Start="2015-12-03" End="2015-12-06" RatePlanCode="RACK" InvTypeCode="KING" Mon="false" Tue="false"
  Weds="true" Thur="true" Fri="true" Sat="true" Sun="false">
    <DestinationSystemCodes>
      <DestinationSystemCode>327</DestinationSystemCode>
    </DestinationSystemCodes>
  </StatusApplicationControl>
```

```
    <LengthsOfStay FixedPatternLength="8">
      <LengthOfStay MinMaxMessageType="FullPatternLOS">
        <LOS_Pattern FullPatternLOS="YNNNYYYY">
        </LOS_Pattern>
      </LengthOfStay>
    </LengthsOfStay>
    <UniqueID Type="16" ID="6">
    </UniqueID>
</AvailStatusMessage>
```

4.2.3 *Use Case: Update Min Lead Days and Max Lead Days*

4.2.3.1 Update Min and Max Lead Days restrictions

This message example changes Min Lead Days to P7D and Max Lead Days to P14D.

```
<AvailStatusMessage>
  <UniqueID Type="16" ID="1"/>
  <StatusApplicationControl Start="2015-05-01" End="2015-12-31" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true"
   Weds="true" Thur="true" Fri="true" Sat="true" Sun="true">
    <DestinationSystemCodes>
      <DestinationSystemCode>330</DestinationSystemCode>
    </DestinationSystemCodes>
  </StatusApplicationControl>
  <RestrictionStatus MinAdvancedBookingOffset="P7D" MaxAdvancedBookingOffset="P14D"/>
</AvailStatusMessage>
```

4.2.3.2 Remove Min and Max Lead Days

```
<AvailStatusMessage>
  <UniqueID Type="16" ID="1" />
  <StatusApplicationControl Start="2015-05-01" End="2015-12-31" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true"
   Weds="true" Thur="true" Fri="true" Sat="true" Sun="true">
    <DestinationSystemCodes>
      <DestinationSystemCode>330</DestinationSystemCode>
    </DestinationSystemCodes>
  </StatusApplicationControl>
  <RestrictionStatus MinAdvancedBookingOffset="P0D" MaxAdvancedBookingOffset="P0D"/>
</AvailStatusMessage>
```

4.2.3.3 Update Min Lead Days and No Changes to Max Lead Days

This message example changes the Min Lead Days to 3 and makes no change to the Max Lead Days value.

```xml
<AvailStatusMessage>
  <UniqueID Type="16" ID="1"/>
  <StatusApplicationControl Start="2015-05-01" End="2015-12-31" InvTypeCode="King" RatePlanCode="Rack" Mon="true" Tue="true"
   Weds="true" Thur="true" Fri="true" Sat="true" Sun="true">
    <DestinationSystemCodes>
      <DestinationSystemCode>330</DestinationSystemCode>
    </DestinationSystemCodes>
  </StatusApplicationControl>
  <RestrictionStatus MinAdvancedBookingOffset="P3D"/>
</AvailStatusMessage>
```

4.2.4 *Use Case: Include Hurdle Value*

For this use case, the ARI Hurdle attribute is enabled for an applicable OTA Partner, and the target hotel is not using LOS Pricing for the relevant OTA Channel.

Hurdle data (hurdle adjustments, deltas, etc.) is taken into consideration for single occupancy only, and the actual hurdle value is returned in the HurdleRate element. This can be used, along with the rate pricing received in the RatePlanNotif message for the given product and dates, to determine if the hurdle value set is met by a given occupancy's price.

For further details on the Hurdle functionality, please refer to the OTA_HotelAvailNotifRQ Message Description.

```xml
<AvailStatusMessage>
  <StatusApplicationControl Start="2023-02-24" End="2023-02-26" RatePlanCode="RATE4" InvTypeCode="ROOM1" Mon="false" Tue="false"
   Weds="false" Thur="false" Fri="true" Sat="true" Sun="true">
    <DestinationSystemCodes>
      <DestinationSystemCode>411</DestinationSystemCode>
    </DestinationSystemCodes>
  </StatusApplicationControl>
  <HurdleRate Amount="59.12" CurrencyCode="EUR">
  </HurdleRate>
  <UniqueID Type="16" ID="6">
  </UniqueID>
</AvailStatusMessage>
```

4.2.5 *Use Case: Successful response*

```xml
<OTA_HotelAvailNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelAvailNotifRS.xsd" EchoToken="" TimeStamp="2015-10-01T09:41:59.982"
Version="6.00" MessageContentCode="3">
  <Success/>
</OTA_HotelAvailNotifRS>
```

4.2.6 *Use Case: Successful response with warnings*

```xml
<OTA_HotelAvailNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelAvailNotifRS.xsd" EchoToken="" TimeStamp="2015-10-01T09:41:59.982"
Version="6.00" MessageContentCode="3">
  <Success/>
  <Warnings>
    <Warning Type="3" Code="356" RecordID="2">Warning text</Warning>
  </Warnings>
</OTA_HotelAvailNotifRS>
```

4.2.7 *Use Case: Error response*

```xml
<OTA_HotelAvailNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelAvailNotifRS.xsd" EchoToken="" TimeStamp="2015-10-01T09:41:59.982"
Version="6.00" MessageContentCode="3">
  <Errors>
    <Error Type="3" Code="450" RecordID="2">Error text</Error>
  </Errors>
</OTA_HotelAvailNotifRS>
```

4.3 OTA_HotelAvailNotifRQ Attribute Descriptions

Element/Attribute OTA_HotelAvailNotifRQ	Description / XPath	Type
@TimeStamp	The instant of message creation. **Parent XPath:** /OTA_HotelAvailNotifRQ	DateTime
@EchoToken	Simple string returned verbatim in the response as passed in the request. Can be used to match requests with responses to simulate session state. **Parent XPath:** /OTA_HotelAvailNotifRQ	String
@Version	Mandatory OTA attribute. Valid value: "6.00". **Parent XPath:** /OTA_HotelAvailNotifRQ	Decimal
@MessageContentCode	The attribute refers to OTA code list MCC. It includes RateAvail, RoomAvail, RoomRateAvail, SegmentAvail, SegmentRoomAvail, HouseAvail, and HurdleRateUpdate. This value is always "3" for RoomRateAvail (product). **Parent XPath:** /OTA_HotelAvailNotifRQ	Integer
AvailStatusMessages		
@HotelCode	The code that uniquely identifies a single hotel property. This corresponds to the SynXis CRS Hotel ID value. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages	String
AvailStatusMessage		
UniqueID		
@Type	Refers to OTA code list UIT. "16" (Reference) is always sent. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/UniqueID	Integer
@ID	A unique incremental number for each AvailStatusMessage element that identifies the specific availability update. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/UniqueID	Integer

Element/Attribute OTA_HotelAvailNotifRQ	Description / XPath	Type
StatusApplicationControl		
@Start	The start date of the availability update. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/StatusApplicationControl	DateTime
@End	The end date of the availability update. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/StatusApplicationControl	DateTime
@InvTypeCode	The code that uniquely identifies a single hotel room type. This corresponds to the SynXis CRS Room Code value. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/StatusApplicationControl	String
@RatePlanCode	The code that uniquely identifies a single hotel rate type. This corresponds to the SynXis CRS Rate Code value. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/StatusApplicationControl	String
@Sun, Mon, Tue, Weds, Thur, Fri, Sat	Boolean value that represents whether that day of week should be included in the update (default is true if attribute is excluded). If one is sent all must be sent. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/StatusApplicationControl	Boolean
DestinationSystemCodes		
DestinationSystemCode	The destination system code defines a system to which information is to be provided. This corresponds to the SynXis CRS Channel Code value. All updates in a message are for the same channel which is defined by this code. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/StatusApplicationControl/ DestinationSystemCodes	String
RestrictionStatus	Used for Open, Close, Closed to Arrival, Lead Days, Closed to Departure, Min Lead Days and Max Lead Days restrictions.	
@Restriction	Defines the type of restriction. SynXis CRS supported enumerations:	String

Element/Attribute **OTA_HotelAvailNotifRQ**	Description / XPath	Type
	Master – Global level. Arrival – Applies only for the day of arrival. Departure – Applies only for the day of departure. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/RestrictionStatus	
@Status	The value of the restriction. **Valid values:** "Open" or "Close". **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/RestrictionStatus	String
@MinAdvancedBooking Offset	Included when Min Sell Date or Min Lead Days is returned for the product, date, or channel in SynXis CRS. The value is based on the most restrictive Min Lead Days as defined in SynXis CRS. When the Min Lead Days is removed, the value is "0". **Note:** This attribute is pulling data from two different restriction types available in SynXis CRS - Min Sell Date and Min Lead Days - which have similar related functionality and end result. The hotel partners may be using one or both of the functionalities. There may be scenarios between those two SynXis CRS restrictions which cause the @MinAdvancedBookingOffset to be greater than the @MaxAdvancedBookingOffset. If the receiving system is unable to handle this scenario directly, the recommendation is to consider the product(s) and date(s) as unavailable based on the values sent. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/RestrictionStatus	Duration P+value+D
@MaxAdvancedBooking Offset	Included when the Max Lead Days is updated for the product, date, or channel in SynXis CRS. The value is based on the most restrictive value as defined in SynXis CRS. **Note:** This attribute is pulling data from two different restriction types available in SynXis CRS - Max Sell Date and Max Lead Days - which have similar related functionality and end result. The hotel partners may be using one or both of the functionalities. There may be scenarios between those two SynXis CRS restrictions which cause the @MaxAdvancedBookingOffset to be less than the @MinAdvancedBookingOffset. If the receiving system is unable to handle this scenario directly, the recommendation is to consider the product(s) and date(s) as unavailable based on the values sent.	Duration P+value+D

Element/Attribute **OTA_HotelAvailNotifRQ**	Description / XPath	Type
	Parent XPath: /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/RestrictionStatus	
LengthsOfStay		
@FixedPatternLength	Shows the number of Length of Stay values included in the ARI data. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/LengthsOfStay	
LengthOfStay	Used for Min Length of Stay, Min Length of Stay Thru, Max Length of Stay and Max Length of Stay Thru restrictions.	
@MinMaxMessageType	Defines the value of the LOS. SynXis CRS supported enumerations: • SetMinLOS – use to set minimum stay from arrival. • FullPatternLOS – use to set closed to arrival FPLOS. • SetForwardMinStay – use to set minimum stay thru. • SetMaxLOS – use to set maximum stay from arrival. • SetForwardMaxStay – use to set maximum stay thru. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/LengthsOfStay/LengthOfStay	String
@TimeUnit	This is always "Day". **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/LengthsOfStay/LengthOfStay	String
@Time	# of Days value for length of stay restriction. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/LengthsOfStay/LengthOfStay	Integer
LOS_Pattern		
@FullPatternLOS	Defines if each Length of Stay is open or closed using a series of Boolean (Y or N) values. Y indicates the corresponding LOS is open to arrival. N indicates the corresponding LOS is closed to arrival. **Parent XPath:**	String

Element/Attribute OTA_HotelAvailNotifRQ	Description / XPath	Type
	/OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/LengthsOfStay/LengthOfStay/LOS_Pattern	
HurdleRate	Contains hurdle value details. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage	
@Amount	The monetary amount of the hurdle rate. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/HurdleRate	Decimal
@CurrencyCode	Rate Specific Currency if applicable to the target rate object, otherwise Hotel Specific Currency. **Parent XPath:** /OTA_HotelAvailNotifRQ/AvailStatusMessages/AvailStatusMessage/HurdleRate	String

4.4 OTA_HotelAvailNotifRS Attribute Descriptions

Element/Attribute OTA_HotelAvailNotifRS	Description / XPath	Type	Required
@TimeStamp	The instant of message creation. **Parent XPath:** /OTA_HotelAvailNotifRS	DateTime	Yes
@EchoToken	Simple string returned verbatim in the response as passed in the request. Can be used to match requests with responses to simulate session state. *Required if a value is sent in the request. **Parent XPath:** /OTA_HotelAvailNotifRS	String	*Yes
@Version	Mandatory OTA attribute. **Parent XPath:** /OTA_HotelAvailNotifRS	Decimal	Yes
@MessageContentCode	The attribute refers to OTA code list MCC which includes RateAvail, RoomAvail, RoomRateAvail, SegmentAvail, SegmentRoomAvail, HouseAvail, and HurdleRateUpdate. This value is always "3" for RoomRateAvail (product). **Parent XPath:** /OTA_HotelAvailNotifRS	Integer	Yes
Success	If included, this element indicates that the request message was successfully processed. *Either a Success element or Errors/Error element(s) are required in every response. **Parent XPath:** /OTA_HotelAvailNotifRS		*Yes
Warnings	Warnings indicate issues which did not prevent processing.		
Warning	Description of cause for a non-fatal problem during request message processing. *If a Warnings element is included one or more Warning elements are required. **Parent XPath:** /OTA_HotelAvailNotifRS/Warnings	String	*Yes

Element/Attribute OTA_HotelAvailNotifRS	Description / XPath	Type	Required
@Type	This is an enumeration of HTNG approved error types (OTA code list EWT). See Appendix for supported values. *This attribute is required if a warning element is included. **Parent XPath:** /OTA_HotelAvailNotifRS/Warnings/Warning	Integer	*Yes
@Code	This is an enumeration of HTNG approved error codes (OTA code list ERR). See Appendix for supported values. *This attribute is required if a warning element is included. **Parent XPath:** /OTA_HotelAvailNotifRS/Warnings/Warning	String	*Yes
@RecordID	If the receiving system identifies which AvailStatusMessage element(s) in the request had an issue, the unique incremental value from the request's UniqueID/@ID is provided here. **Parent XPath:** /OTA_HotelAvailNotifRS/Warnings/Warning	Integer	
Errors	*Either a Success element or Errors/Error element(s) are required in every response.		*Yes
Error	Description of cause for a fatal problem during request message processing. *If an Errors element is included one or more Error elements are required. **Parent XPath:** /OTA_HotelAvailNotifRS/Errors	String	*Yes
@Type	This is an enumeration of HTNG approved error types (OTA code list EWT). See Appendix for supported values. *This attribute is required if an error element is included. **Parent XPath:** /OTA_HotelAvailNotifRS/Errors/Error	Integer	*Yes
@Code	This is an enumeration of HTNG approved error codes (OTA code list ERR). See Appendix for supported values. *This attribute is required if an error element is included. **Parent XPath:** /OTA_HotelAvailNotifRS/Errors/Error	String	*Yes

Element/Attribute OTA_HotelAvailNotifRS	Description / XPath	Type	Required
@RecordID	If the receiving system identifies which AvailStatusMessage element(s) in the request had an issue, the unique incremental value from the request's UniqueID/@ID is entered here. **Parent XPath:** /OTA_HotelAvailNotifRS/Errors/Error	Integer	

5 OTA_HotelRatePlanNotifRQ/RS: Update Rate Amounts

This section provides the message examples used to communicate rate amount values to the client systems.

5.1 OTA_HotelRatePlanNotifRQ Message Description

- The OTA_HotelRatePlanNotifRQ message communicates rate amount values to the client systems. Rate amounts are communicated for all adult occupancies supported by a room type, including 'before' and 'after' tax occupancy rate amounts for each occupant.

- Each OTA_HotelRatePlanNotifRQ message contains rate amounts for one product for a hotel.

- If the applicable room has a guest limit value of 1 or 2, then 2 BaseByGuestAmt elements are included in the OTA_HotelRatePlanNotifRQ message to communicate single and double occupancy.
  Note: Guest Limit is calculated as: [Guest per room value + 1 if 'Allow Extra Bed' is set to true].

- If the applicable room has a guest limit value of 3, 4, 5 or [x], then 3, 4, 5 or [x] BaseByGuestAmt elements are included in the OTA_HotelRatePlanNotifRQ message.

- When Length of Stay (LOS) based ARI is used, rate amounts are provided for the total price of a stay by arrival data/length of stay. LOS pricing supports LOS values of 1-8.

- To support date range functionality, rate amounts with the same values are consolidated into a date range of values in a RatePlan element.

- Each RatePlan element contains a counter for error handling to determine which update in a message failed.

- If child pricing is enabled for the ARI switch, rate amounts can also be communicated for child occupancies.

- Child pricing can be communicated in three ways:
  - By child occupancies supported by a room type. The value for the child occupancy for a room type must be set up.
  - By child age ranges, if 'Child Age Range Pricing' attribute is enabled and child age ranges are set up for a hotel.
  - By child occupancies and child age ranges. This requires the 'Child Age Range Pricing' and 'Child Age Range Offset by Occupancy' attributes enabled for a hotel. Child age ranges and the value for the child occupancy for a room type must be set up.

    Note: A rate season must be configured for a product/date in SynXis CRS, which includes the price offsets for children, either by child occupancy (as minimum 'Extra Child' offset must be set up), by child age ranges, or both depending on how child pricing is communicated. Otherwise, 'zero' prices are communicated for children. When child age ranges are not configured, these prices are not communicated.

- With child pricing by age, the price for a child in unknown age is always communicated without specifying the @MinAge and @MaxAge attributes, either as a configured offset value or as a 'zero' price, if the offset is not configured.

- For a room to be booked with a child as a guest, there always has to be at least 1 adult occupancy. Child prices are sent as offsets for the specified number of children, as defined by the @NumberOfGuests attribute (if using occupancy pricing) or as offsets per 1 child of a specific age range (if using child age range pricing), or as a combination of both.

- With child pricing by occupancy, if the maximum child occupancy is above 5, then the value of 'Extra child' offset as configured in SynXis CRS is added to the price for the previous child occupancy.

- The maximum child occupancy supported by a SynXis CRS room type is determined as follows:
    - If 'Allow Extra Bed' is set to true, then it adds +1 to the calculated maximum child occupancy value.
    - If 'Child Occupancy' is limited, then the maximum child occupancy is 'Child Occupancy' value.
    - If 'Child Occupancy' is unlimited and 'Total Guests per Room' is limited, then the maximum child occupancy is 'Total Guests per Room' value minus (-) 1.
    - If 'Child Occupancy' is unlimited and 'Total Guests per Room' is unlimited, then the maximum child occupancy is 'Adult Occupancy' value minus (-) 1.
- When a rate season is configured for a product/date in SynXis CRS, which does not include offsets for children by occupancy and the extra child offset is set up, then the child prices are calculated (the 'Extra Child' offset is multiplied by the number of children) and communicated for each child quantity, from 1 up to the maximum child occupancy supported by a room type.
- A breakdown of tax/fee details is included in the OTA_HotelRatePlanNotifRQ message and communicated to an OTA which supports receiving tax/fee details.
  Note: When determining if an update needs to be communicated, only the changes in the final price are considered, the changes in the tax breakdown are not tracked.
- If the Include Tax Details configuration is <u>not</u> enabled for a given ARI switch, then the tax breakdown is not communicated, and the Taxes element is not included.
- Communicating tax/fee breakdowns for both per night and per stay taxes/fees is possible for hotels setup for LOS pricing. In the case of non-LOS pricing, the per stay product taxes/fees are not applied to pricing. However, the package taxes may be applied to product pricing when static packages are applied to product pricing.
  Note: Per stay taxes/fees are included in LOS pricing only if an OTA chooses to receive a tax breakdown in the message.
- For the hotels setup to use child pricing, the child offset pricing (and the tax/fee breakdown if applicable) is included only for non-LOS pricing.

5.2 OTA_HotelRatePlanNotifRQ/RS Use Cases

5.2.1 *Use Case: Update rate amounts for a product – Adult occupancy*

The following example communicates ARI rate plan updates for a product. In this message, occupancy detail shows single and double adult occupancy. @AgeQualifyingCode="10" specifies an adult guest.

```
<OTA_HotelRatePlanNotifRQ xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" MessageContentCode="8" Version="6.00" TimeStamp="2015-07-11T14:53:29.7710856Z">
  <RatePlans HotelCode="59422">
    <RatePlan RatePlanNotifType="Delta" RatePlanCode="stdOCCP4" End="2015-07-11" Start="2015-07-11">
      <DestinationSystemsCode>
        <DestinationSystemCode>6083</DestinationSystemCode>
      </DestinationSystemsCode>
      <Rates>
        <Rate InvTypeCode="OCCP4" CurrencyCode="PLN" Sun="false" Sat="false" Fri="false" Thur="true" Weds="false" Tue="false"
        Mon="false">
          <BaseByGuestAmts>
            <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="161.78" AmountBeforeTax="144"></BaseByGuestAmt>
            <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="161.78" AmountBeforeTax="144"></BaseByGuestAmt>
          </BaseByGuestAmts>
        </Rate>
      </Rates>
      <UniqueID ID="1" Type="16"></UniqueID>
    </RatePlan>
  </RatePlans>
</OTA_HotelRatePlanNotifRQ>
```

The following message example shows occupancy rate amounts for 5 adult occupants.

```
<BaseByGuestAmts>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="161.78" AmountBeforeTax="144"></BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="161.78" AmountBeforeTax="144"></BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="3" AmountAfterTax="161.78" AmountBeforeTax="144"></BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="4" AmountAfterTax="161.78" AmountBeforeTax="144"></BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="5" AmountAfterTax="161.78" AmountBeforeTax="144"></BaseByGuestAmt>
</BaseByGuestAmts>
```

5.2.2 *Use Case: Update rate amounts for a product – Child pricing*

5.2.2.1 Child pricing by occupancy

The following example shows child pricing when the 'Child Age Range Pricing' is not enabled for a hotel and there is a season with offsets created for a product/date in SynXis CRS.

In this message a maximum child occupancy of 6 is configured. The child prices that are set up on a season are communicated for each child occupancy supported on a room type, from 1 up to 5, and a value of the 'Extra Child' offset is added to the price for 5 children and communicated for 6 children.

@AgeQualifyingCode="10" specifies an adult guest and @AgeQualifyingCode="8" specifies a child.

```
<Rate Mon="true" Tue="true" Weds="true" Thur="true" Fri="true" Sat="true" Sun="true" CurrencyCode="USD" InvTypeCode="DDRA">
  <BaseByGuestAmts>
    <BaseByGuestAmt AmountBeforeTax="100.00" AmountAfterTax="110.00" NumberOfGuests="1" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="200.00" AmountAfterTax="220.00" NumberOfGuests="2" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="300.00" AmountAfterTax="330.00" NumberOfGuests="3" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="400.00" AmountAfterTax="440.00" NumberOfGuests="4" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="540.00" AmountAfterTax="550.00" NumberOfGuests="5" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="600.00" AmountAfterTax="660.00" NumberOfGuests="6" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="700.00" AmountAfterTax="770.00" NumberOfGuests="7" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="60.00" AmountAfterTax="66.00" NumberOfGuests="1" AgeQualifyingCode="8"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="120.00" AmountAfterTax="132.00" NumberOfGuests="2" AgeQualifyingCode="8"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="180.00" AmountAfterTax="198.00" NumberOfGuests="3" AgeQualifyingCode="8"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="240.00" AmountAfterTax="244.00" NumberOfGuests="4" AgeQualifyingCode="8"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="300.00" AmountAfterTax="330.00" NumberOfGuests="5" AgeQualifyingCode="8"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="360.00" AmountAfterTax="396.00" NumberOfGuests="6" AgeQualifyingCode="8"></BaseByGuestAmt>
  </BaseByGuestAmts>
</Rate>
```

5.2.2.2 Child pricing by age range

The following example shows child pricing when the 'Child Age Range Pricing' is enabled for a hotel and child age ranges are configured for a hotel and a rate season is configured for a product/date in SynXis CRS.

In this message the offsets by child age ranges and an offset for a child in unknown age are set up. The prices by child age ranges are communicated for each child age range that is configured for a hotel and a price for a child in unknown age is communicated without specifying the child age range.

```
<Rate Mon="true" Tue="true" Weds="true" Thur="true" Fri="true" Sat="true" Sun="true" CurrencyCode="USD" InvTypeCode="DDRA">
  <BaseByGuestAmts>
    <BaseByGuestAmt AmountBeforeTax="100.00" AmountAfterTax="110.00" NumberOfGuests="1" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="200.00" AmountAfterTax="220.00" NumberOfGuests="2" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="300.00" AmountAfterTax="330.00" NumberOfGuests="3" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="400.00" AmountAfterTax="440.00" NumberOfGuests="4" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="60.00" AmountAfterTax="66.00" AgeQualifyingCode="8" MinAge="0" MaxAge="2"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="70.00" AmountAfterTax="77.00" AgeQualifyingCode="8" MinAge="3" MaxAge="9"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="80.00" AmountAfterTax="88.00" AgeQualifyingCode="8" MinAge="10" MaxAge="18"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="70.00" AmountAfterTax="77.00" AgeQualifyingCode="8"></BaseByGuestAmt>
  </BaseByGuestAmts>
</Rate>
```

5.2.2.3 Child pricing by occupancy and age range with configured age ranges

The following example shows child pricing when the 'Child Age Range Pricing' and the 'Child Age Range Offset By Occupancy' are enabled for a hotel and child age ranges are configured for a hotel and a rate season with offsets is configured for a product/date in SynXis CRS.

In this message the maximum child occupancy of 3 and 3 child age ranges are configured. The child pricing is communicated for each child occupancy supported on a room type, in this case up to 3, and for each child age range that is set up for a hotel.

```
<Rate Mon="true" Tue="true" Weds="true" Thur="true" Fri="true" Sat="true" Sun="true" CurrencyCode="USD" InvTypeCode="DDRA">
  <BaseByGuestAmts>
    <BaseByGuestAmt AmountBeforeTax="100.00" AmountAfterTax="110.00" NumberOfGuests="1" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="200.00" AmountAfterTax="220.00" NumberOfGuests="2" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="300.00" AmountAfterTax="330.00" NumberOfGuests="3" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="400.00" AmountAfterTax="440.00" NumberOfGuests="4" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="60.00" AmountAfterTax="66.00" NumberOfGuests="1" AgeQualifyingCode="8" MinAge="0" MaxAge="2">
    </BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="120.00" AmountAfterTax="132.00" NumberOfGuests="2" AgeQualifyingCode="8" MinAge="0" MaxAge="2">
    </BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="180.00" AmountAfterTax="198.00" NumberOfGuests="3" AgeQualifyingCode="8" MinAge="0" MaxAge="2">
    </BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="70.00" AmountAfterTax="77.00" NumberOfGuests="1" AgeQualifyingCode="8" MinAge="3" MaxAge="9">
    </BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="140.00" AmountAfterTax="154.00" NumberOfGuests="2" AgeQualifyingCode="8" MinAge="3" MaxAge="9">
    </BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="210.00" AmountAfterTax="231.00" NumberOfGuests="3" AgeQualifyingCode="8" MinAge="3" MaxAge="9">
    </BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="80.00" AmountAfterTax="88.00" NumberOfGuests="1" AgeQualifyingCode="8" MinAge="10" MaxAge="18">
    </BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="160.00" AmountAfterTax="176.00" NumberOfGuests="2" AgeQualifyingCode="8" MinAge="10" MaxAge="18">
    </BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="240.00" AmountAfterTax="264.00" NumberOfGuests="3" AgeQualifyingCode="8" MinAge="10" MaxAge="18">
    </BaseByGuestAmt>
  </BaseByGuestAmts>
</Rate>
```

5.2.2.4 Child pricing by occupancy and age range without configured age ranges

The following example shows child pricing when the 'Child Age Range Pricing' and the 'Child Age Range Offset By Occupancy' are enabled for a hotel and child age ranges are not configured for a hotel and a rate season is configured for a product/date in SynXis CRS.

In this message the maximum child occupancy of 3 and the offset for a child in unknown age is configured. The prices for children in unknown age are communicated for each number of children, in this case from 1 up to 3.

```
<Rate Mon="true" Tue="true" Weds="true" Thur="true" Fri="true" Sat="true" Sun="true" CurrencyCode="USD" InvTypeCode="DDRA">
  <BaseByGuestAmts>
    <BaseByGuestAmt AmountBeforeTax="100.00" AmountAfterTax="110.00" NumberOfGuests="1" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="200.00" AmountAfterTax="220.00" NumberOfGuests="2" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="300.00" AmountAfterTax="330.00" NumberOfGuests="3" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="400.00" AmountAfterTax="440.00" NumberOfGuests="4" AgeQualifyingCode="10"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="60.00" AmountAfterTax="66.00" NumberOfGuests="1" AgeQualifyingCode="8"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="120.00" AmountAfterTax="132.00" NumberOfGuests="2" AgeQualifyingCode="8"></BaseByGuestAmt>
    <BaseByGuestAmt AmountBeforeTax="180.00" AmountAfterTax="198.00" NumberOfGuests="3" AgeQualifyingCode="8"></BaseByGuestAmt>
  </BaseByGuestAmts>
</Rate>
```

5.2.3 *Use Case: Update rate amounts for a product by length of stay*

The following message example communicates arrival date/length of stay based rate amounts for a hotel to an OTA. LOS Pricing is supported up to eight nights. A Rate element is added for each length of stay that is included in the ARI data for the product/date and includes @RateTimeUnit, @MinLOS, and @MaxLOS attributes. The prices communicated are for total stay amounts, not prices per night.

Note: OTA certification is required to use LOS Pricing.

In this message, the KING room supports up to two guests. The hotel will charge $100 before tax for a one-night stay for one and two guests. Note that for the two-night stay and beyond, the hotel is charging the same rate per night, but the message shows the total value for the 2-8 nights.

```
<Rates>
  <Rate RateTimeUnit="FullDuration" MinLOS="1" MaxLOS="1" InvTypeCode="KING" CurrencyCode="USD" Sun="false" Sat="false" Fri="false"
  Thur="false" Weds="false" Tue="false" Mon="false">
    <BaseByGuestAmts>
      <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="110" AmountBeforeTax="100"/>
      <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="110" AmountBeforeTax="100"/>
    </BaseByGuestAmts>
  </Rate>
  <Rate RateTimeUnit="FullDuration" MinLOS="2" MaxLOS="2" InvTypeCode="KING" CurrencyCode="USD" Sun="false" Sat="false" Fri="false"
  Thur="false" Weds="false" Tue="false" Mon="true">
    <BaseByGuestAmts>
      <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="220" AmountBeforeTax="200"/>
      <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="220" AmountBeforeTax="200"/>
    </BaseByGuestAmts>
  </Rate>
  …
  <Rate RateTimeUnit="FullDuration" MinLOS="8" MaxLOS="8" InvTypeCode="KING" CurrencyCode="USD" Sun="false" Sat="false" Fri="false"
  Thur="false" Weds="false" Tue="false" Mon="true">
    <BaseByGuestAmts>
      <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="880" AmountBeforeTax="800"/>
      <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="880" AmountBeforeTax="800"/>
    </BaseByGuestAmts>
  </Rate>
</Rates>
```

5.2.4 *Use Case: Include Meal Plans*

The ARI switch has to be configured to have meal plan information communicated in order to add meal plan information to the rate updates.
The rate plan can be configured to include no meals, include breakfast, or different meal plan types. The values are returned in the **MealsIncluded** element.
Note: In order to associate a meal plan with the rate plan, **Breakfast Included** must be enabled in SynXis CRS.

5.2.4.1 No Meals Included in Rate

In the following example, **Breakfast Included** is disabled in SynXis CRS and no meal plan is associated with the rate.

```
<RatePlan RatePlanNotifType="Delta" RatePlanCode="stdOCCP4" End="2019-07-11" Start="2019-07-11">
  …
  <Rates>
    <Rate InvTypeCode="OCCP4" CurrencyCode="USD">
      …
      <MealsIncluded Breakfast="false" Lunch="false" Dinner="false"/>
    </Rate>
  </Rates>
  …
</RatePlan>
```

5.2.4.2 Breakfast Included in Rate

In the following example, **Breakfast Included** is enabled in SynXis CRS but no meal plan is associated with the rate.

```
<RatePlan RatePlanNotifType="Delta" RatePlanCode="stdOCCP4" End="2019-07-11" Start="2019-07-11">
  …
  <Rates>
    <Rate InvTypeCode="OCCP4" CurrencyCode="USD">
      …
      <MealsIncluded Breakfast="true" Lunch="false" Dinner="false"/>
    </Rate>
  </Rates>
  …
</RatePlan>
```

5.2.4.3 All Inclusive Meal Plan Selected

In the following example, **Breakfast Included** is enabled and **All Inclusive** meal plan type is selected in SynXis CRS.

```
<RatePlan RatePlanNotifType="Delta" RatePlanCode="stdOCCP4" End="2019-07-11" Start="2019-07-11">
  …
  <Rates>
    <Rate InvTypeCode="OCCP4" CurrencyCode="USD">
      …
      <MealsIncluded Breakfast="true" Lunch="true" Dinner="true" MealPlanCodes="1"/>
    </Rate>
  </Rates>
  …
</RatePlan>
```

5.2.5 *Use Case: Include Tax / Fee Breakdown*

5.2.5.1 Non-LOS Pricing – Price is tax inclusive

Use case details:
- Include Tax Details configuration is enabled for ARI Switch.
- Hotel Uses Per-Tax Inclusivity configuration is <u>not</u> enabled for hotel.
- Price Season for Product has Price includes tax configuration enabled.
- Pricing for product is as follows:
  - 1 Adult Before Tax = $130.00
  - 1 Adult After Tax = $130.00
  - 2 Adult Before Tax = $150.00
  - 2 Adult After Tax = $150.00

```xml
<BaseByGuestAmts>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="130.00" AmountBeforeTax="130.00"/>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="150.00" AmountBeforeTax="150.00"/>
</BaseByGuestAmts>
```

5.2.5.2 Non-LOS Pricing – Price has inclusive and exclusive taxes

Use case details:
- Include Tax Details configuration is enabled for ARI Switch.
- Hotel Uses Per-Tax Inclusivity configuration is enabled for hotel.
- Product uses the following taxes:
  - <u>Inclusive</u> Per Night Surcharge of $10
  - <u>Exclusive</u> Per Night Lodging Tax of $20
  - <u>Inclusive</u> Per Stay City Tax of $50 (not included in non-LOS pricing)
  - <u>Exclusive</u> Per Stay Resort Fee of $25 (not included in non-LOS pricing)
- Pricing for product is as follows:
  - 1 Adult Base Rate = $100
  - 1 Adult Before Tax = $90.00 (100 – 10)
  - 1 Adult After Tax = $120.00 (100 + 20)
  - 2 Adult Base Rate = $120
  - 2 Adult Before Tax = $110.00 (120 -10)
  - 2 Adult After Tax = $140.00 (120 + 20)

```xml
<BaseByGuestAmts>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="120.00" AmountBeforeTax="90.00">
    <Taxes Amount="30.00">
      <Tax Code="16" Amount="10.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Nightly Surcharge</Text>
        </TaxDescription>
```

```xml
        <TaxDescription Name="Description">
          <Text>This is my $10 per night inclusive surcharge</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="true"/>
        </TPA_Extensions>
      </Tax>
      <Tax Code="8" Amount="20.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Nightly Lodging Tax</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $20 per night exclusive lodging tax</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
    </Taxes>
  </BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="140.00" AmountBeforeTax="110.00">
    <Taxes Amount="30.00">
      <Tax Code="16" Amount="10.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Nightly Surcharge</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $10 per night inclusive surcharge</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="true"/>
        </TPA_Extensions>
      </Tax>
      <Tax Code="8" Amount="20.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Nightly Lodging Tax</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $20 per night exclusive lodging tax</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
    </Taxes>
  </BaseByGuestAmt>
</BaseByGuestAmts>
```

5.2.5.3 LOS Pricing – ARI Switch not configured to receive tax details

Use case details:

- Include Tax Details configuration is not enabled for ARI Switch.
- Hotel Uses Per-Tax Inclusivity configuration is not enabled for hotel.
- Price Season for Product does not have Price includes tax configuration enabled.
- Product uses the following taxes:
    - Per Night Bed Tax of $10
    - Per Night City Tax of $20
    - Per Stay Resort Fee of $50 (ignored when tax breakdown is not included)
- Total Pricing for Stay is as follows:
    - LOS 1 - 1 Adult Before Tax = $100.00
    - LOS 1 - 1 Adult After Tax = $130.00 (100 + 10 + 20)
    - LOS 1 - 2 Adult Before Tax = $120.00
    - LOS 1 - 2 Adult After Tax = $150.00 (120 + 10 + 20)
    - LOS 2 - 1 Adult Before Tax = $200.00 (100 + 100)
    - LOS 2 - 1 Adult After Tax = $260.00 (100 + 100 + 10 + 10 + 20 + 20)
    - LOS 2 - 2 Adult Before Tax = $240.00 (120 + 120)
    - LOS 2 - 2 Adult After Tax = $300.00 (120 + 120 + 10 + 10 + 20 + 20)

```
<Rate RateTimeUnit="FullDuration" MinLOS="1" MaxLOS="1" InvTypeCode="KING" CurrencyCode="USD">
  <BaseByGuestAmts>
    <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="130.00" AmountBeforeTax="100.00"/>
    <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="150.00" AmountBeforeTax="120.00"/>
  </BaseByGuestAmts>
</Rate>
<Rate RateTimeUnit="FullDuration" MinLOS="2" MaxLOS="2" InvTypeCode="KING" CurrencyCode="USD">
  <BaseByGuestAmts>
    <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="260.00" AmountBeforeTax="200.00"/>
    <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="300.00" AmountBeforeTax="240.00"/>
  </BaseByGuestAmts>
</Rate>
```

5.2.5.4 LOS Pricing – Price has exclusive taxes

Use case details:

- Include Tax Details configuration is enabled for ARI Switch.
- Hotel Uses Per-Tax Inclusivity configuration is <u>not</u> enabled for hotel.
- Price Season for Product does <u>not</u> have Price includes tax configuration enabled.
- Product uses the following taxes:
  - Per Night Surcharge of $10
  - Per Night Lodging Tax of $20
  - Per Stay Resort Fee of $50
- Total Pricing for Stay is as follows:
  - LOS 1 - 1 Adult Before Tax = $100.00
  - LOS 1 - 1 Adult After Tax = $180.00 (100 + 10 + 20 + 50)
  - LOS 1 - 2 Adult Before Tax = $120.00
  - LOS 1 - 2 Adult After Tax = $200.00 (120 + 10 + 20 + 50)
  - LOS 2 - 1 Adult Before Tax = $200.00 (100 + 100)
  - LOS 2 - 1 Adult After Tax = $310.00 (100 + 100 + 10 + 10 + 20 + 20 + 50)
  - LOS 2 - 2 Adult Before Tax = $240.00 (120 + 120)
  - LOS 2 - 2 Adult After Tax = $350.00 (120 + 120 + 10 + 10 + 20 + 20 + 50)

```
<Rate RateTimeUnit="FullDuration" MinLOS="1" MaxLOS="1" InvTypeCode="KING" CurrencyCode="USD">
  <BaseByGuestAmts>
    <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="180.00" AmountBeforeTax="100.00">
      <Taxes Amount="80.00">
        <Tax Code="16" Amount="10.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
          <TaxDescription Name="Name">
            <Text>Nightly Surcharge</Text>
          </TaxDescription>
          <TaxDescription Name="Description">
            <Text>This is my $10 per night exclusive surcharge</Text>
          </TaxDescription>
          <TPA_Extensions>
            <AdditionalConfigurations Inclusive="false"/>
          </TPA_Extensions>
        </Tax>
        <Tax Code="8" Amount="20.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
          <TaxDescription Name="Name">
            <Text>Nightly Lodging Tax</Text>
          </TaxDescription>
          <TaxDescription Name="Description">
            <Text>This is my $20 per night exclusive lodging tax</Text>
          </TaxDescription>
          <TPA_Extensions>
```

```xml
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
      <Tax Code="12" Amount="50.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="12">
        <TaxDescription Name="Name">
          <Text>Resort Fee</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $50 per stay exclusive resort fee</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
    </Taxes>
  </BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="200.00" AmountBeforeTax="120.00">
    <Taxes Amount="80.00">
      <Tax Code="16" Amount="10.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Nightly Surcharge</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $10 per night exclusive surcharge</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
      <Tax Code="8" Amount="20.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Nightly Lodging Tax</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $20 per night exclusive lodging tax</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
      <Tax Code="12" Amount="50.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="12">
        <TaxDescription Name="Name">
          <Text>Resort Fee</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $50 per stay exclusive resort fee</Text>
        </TaxDescription>
```

```xml
              <TPA_Extensions>
                <AdditionalConfigurations Inclusive="false"/>
              </TPA_Extensions>
            </Tax>
          </Taxes>
        </BaseByGuestAmt>
      </BaseByGuestAmts>
    </Rate>
    <Rate RateTimeUnit="FullDuration" MinLOS="2" MaxLOS="2" InvTypeCode="KING" CurrencyCode="USD">
      <BaseByGuestAmts>
        <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="310.00" AmountBeforeTax="200.00">
          <Taxes Amount="110.00">
            <Tax Code="16" Amount="10.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-02" ChargeFrequency="1">
              <TaxDescription Name="Name">
                <Text>Nightly Surcharge</Text>
              </TaxDescription>
              <TaxDescription Name="Description">
                <Text>This is my $10 per night exclusive surcharge</Text>
              </TaxDescription>
              <TPA_Extensions>
                <AdditionalConfigurations Inclusive="false"/>
              </TPA_Extensions>
            </Tax>
            <Tax Code="8" Amount="20.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-02" ChargeFrequency="1">
              <TaxDescription Name="Name">
                <Text>Nightly Lodging Tax</Text>
              </TaxDescription>
              <TaxDescription Name="Description">
                <Text>This is my $20 per night exclusive lodging tax</Text>
              </TaxDescription>
              <TPA_Extensions>
                <AdditionalConfigurations Inclusive="false"/>
              </TPA_Extensions>
            </Tax>
            <Tax Code="12" Amount="50.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-02" ChargeFrequency="12">
              <TaxDescription Name="Name">
                <Text>Resort Fee</Text>
              </TaxDescription>
              <TaxDescription Name="Description">
                <Text>This is my $50 per stay exclusive resort fee</Text>
              </TaxDescription>
              <TPA_Extensions>
                <AdditionalConfigurations Inclusive="false"/>
              </TPA_Extensions>
            </Tax>
          </Taxes>
        </BaseByGuestAmt>
```

```xml
    <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="350.00" AmountBeforeTax="240.00">
      <Taxes Amount="110.00">
        <Tax Code="16" Amount="10.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-02" ChargeFrequency="1">
          <TaxDescription Name="Name">
            <Text>Nightly Surcharge</Text>
          </TaxDescription>
          <TaxDescription Name="Description">
            <Text>This is my $10 per night exclusive surcharge</Text>
          </TaxDescription>
          <TPA_Extensions>
            <AdditionalConfigurations Inclusive="false"/>
          </TPA_Extensions>
        </Tax>
        <Tax Code="8" Amount="20.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-02" ChargeFrequency="1">
          <TaxDescription Name="Name">
            <Text>Nightly Lodging Tax</Text>
          </TaxDescription>
          <TaxDescription Name="Description">
            <Text>This is my $20 per night exclusive lodging tax</Text>
          </TaxDescription>
          <TPA_Extensions>
            <AdditionalConfigurations Inclusive="false"/>
          </TPA_Extensions>
        </Tax>
        <Tax Code="12" Amount="50.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-02" ChargeFrequency="12">
          <TaxDescription Name="Name">
            <Text>Resort Fee</Text>
          </TaxDescription>
          <TaxDescription Name="Description">
            <Text>This is my $50 per stay exclusive resort fee</Text>
          </TaxDescription>
          <TPA_Extensions>
            <AdditionalConfigurations Inclusive="false"/>
          </TPA_Extensions>
        </Tax>
      </Taxes>
    </BaseByGuestAmt>
  </BaseByGuestAmts>
</Rate>
```

5.2.5.5 Non-LOS Pricing – Child Pricing – Adult Only Tax

Use case details:

- Include Tax Details configuration is enabled for ARI Switch.
- Hotel Uses Per-Tax Inclusivity configuration is not enabled for hotel.
- Price Season for Product does not have Price includes tax configuration enabled.

- Product uses the following taxes:
  - Per Night Per Adult Exclusive Tax of $20
- Configured pricing for product is as follows:
  - 1 Adult Before Tax = $100.00
  - 2 Adult Before Tax = $200.00
  - 1 Child Before Tax = $20.00
  - 2 Child Before Tax = $40.00

In this use case the tax applies only to the adult pricing, therefore, no tax/fee is detailed in the child offset price that is included. By the same token, if the tax applied only to the child offset price, no tax/fee would be detailed in the adult only price included.

```xml
<BaseByGuestAmts>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="120.00" AmountBeforeTax="100.00">
    <Taxes Amount="20.00">
      <Tax Code="16" Amount="20.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Per Night Per Adult Tax</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $20 per night per adult exclusive tax</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
    </Taxes>
  </BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="240.00" AmountBeforeTax="200.00">
    <Taxes Amount="40.00">
      <Tax Code="16" Amount="40.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Per Night Per Adult Tax</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my $20 per night per adult exclusive tax</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
    </Taxes>
  </BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="8" NumberOfGuests="1" AmountAfterTax="20.00" AmountBeforeTax="20.00"/>
  <BaseByGuestAmt AgeQualifyingCode="8" NumberOfGuests="2" AmountAfterTax="40.00" AmountBeforeTax="40.00"/>
</BaseByGuestAmts>
```

5.2.5.6 Non-LOS Pricing – Child Pricing – Adult and Child Tax

Use case details:

- Include Tax Details configuration is enabled for ARI Switch.
- Hotel Uses Per-Tax Inclusivity configuration is not enabled for hotel.
- Price Season for Product does not have Price includes tax configuration enabled.
- Product uses the following taxes:
    - Per Night Flat Charge Exclusive Tax of 10%
- Configured pricing for product is as follows:
    - 1 Adult Before Tax = $100.00
    - 2 Adult Before Tax = $200.00
    - 1 Child Before Tax = $20.00
    - 2 Child Before Tax = $40.00

In this use case, the tax applies to both adult pricing and child offset pricing, therefore, the tax/fee is detailed for both adult only price and child offset price included.

```xml
<BaseByGuestAmts>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="1" AmountAfterTax="110.00" AmountBeforeTax="100.00">
    <Taxes Amount="10.00">
      <Tax Code="16" Amount="10.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Per Night Flat 10 Percent Tax</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my per night flat 10 percent exclusive tax</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
    </Taxes>
  </BaseByGuestAmt>
  <BaseByGuestAmt AgeQualifyingCode="10" NumberOfGuests="2" AmountAfterTax="220.00" AmountBeforeTax="200.00">
    <Taxes Amount="20.00">
      <Tax Code="16" Amount="20.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
        <TaxDescription Name="Name">
          <Text>Per Night Flat 10 Percent Tax</Text>
        </TaxDescription>
        <TaxDescription Name="Description">
          <Text>This is my per night flat 10 percent exclusive tax</Text>
        </TaxDescription>
        <TPA_Extensions>
          <AdditionalConfigurations Inclusive="false"/>
        </TPA_Extensions>
      </Tax>
```

```
        </Taxes>
    </BaseByGuestAmt>
    <BaseByGuestAmt AgeQualifyingCode="8" NumberOfGuests="1" AmountAfterTax="22.00" AmountBeforeTax="20.00">
        <Taxes Amount="2.00">
            <Tax Code="16" Amount="2.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
                <TaxDescription Name="Name">
                    <Text>Per Night Flat 10 Percent Tax</Text>
                </TaxDescription>
                <TaxDescription Name="Description">
                    <Text>This is my per night flat 10 percent exclusive tax</Text>
                </TaxDescription>
                <TPA_Extensions>
                    <AdditionalConfigurations Inclusive="false"/>
                </TPA_Extensions>
            </Tax>
        </Taxes>
    </BaseByGuestAmt>
    <BaseByGuestAmt AgeQualifyingCode="8" NumberOfGuests="2" AmountAfterTax="44.00" AmountBeforeTax="40.00">
        <Taxes Amount="4.00">
            <Tax Code="16" Amount="4.00" EffectiveDate="2019-12-01" ExpireDate="2019-12-01" ChargeFrequency="1">
                <TaxDescription Name="Name">
                    <Text>Per Night Flat 10 Percent Tax</Text>
                </TaxDescription>
                <TaxDescription Name="Description">
                    <Text>This is my per night flat 10 percent exclusive tax</Text>
                </TaxDescription>
                <TPA_Extensions>
                    <AdditionalConfigurations Inclusive="false"/>
                </TPA_Extensions>
            </Tax>
        </Taxes>
    </BaseByGuestAmt>
</BaseByGuestAmts>
```

5.2.6 *Use Case: Include Cancel Policy Details*

The ARI switch has to be configured to have cancel policy information communicated in order to add cancel policy information to the rate updates.

The cancel policy details are returned in the **CancelPolicies** element.

The cancel policy communicated is the cancel policy that is applicable to the product stay that is included in the Rate element. When the cancel policy has the Staggered Penalty Charge option enabled, the staggered charge which has the <u>largest</u> cancel charge value is used.

Note: RatePlan elements with consecutive dates are <u>not</u> consolidated unless <u>all</u> values (including cancel policy information) included in the RatePlan element are exactly the same.

5.2.6.1 Cancel Type is Always Charge Cancel Fee

Cancel Policy has the following configurations:
- Cancel Type = Always Charge Cancel Fee
- Cancel Charge Type = # nights of stay
- Number of Nights = 1
- Charge Includes Tax = Not Enabled

```xml
<Rate InvTypeCode="KING" CurrencyCode="USD">
  …
  <CancelPolicies>
    <CancelPenalty PolicyCode="CP1">
      <AmountPercent TaxInclusive="false" NmbrOfNights="1"/>
    </CancelPenalty>
  </CancelPolicies>
</Rate>
```

5.2.6.2 Cancel Type is Charge Within Days

Cancel Policy has the following configurations:
- Cancel Type = Charge Within Days
- Days = 3
- Cancel Charge Type = % of amt of stay
- Percentage of Amount of Stay = 10
- Charge Includes Tax = Enabled

```xml
<Rate InvTypeCode="KING" CurrencyCode="USD">
  …
  <CancelPolicies>
    <CancelPenalty PolicyCode="CP2">
      <Deadline OffsetTimeUnit="Day" OffsetUnitMultiplier="3" OffsetDropTime="BeforeArrival"/>
      <AmountPercent TaxInclusive="true" Percent="10"/>
    </CancelPenalty>
  </CancelPolicies>
</Rate>
```

5.2.6.3 Cancel Type is Charge Within Time

Cancel Policy has the following configurations:
- Cancel Type = Charge Within Time
- Cancel Time = 11:00
- Cancel Charge Type = local currency
- Amount in Local Currency = 50
- Charge Includes Tax = Not Enabled

Arrival date for update = 8/10/2019

```xml
<Rate InvTypeCode="KING" CurrencyCode="USD">
  …
  <CancelPolicies>
    <CancelPenalty PolicyCode="CP5">
      <Deadline AbsoluteDeadline="2019-08-10T11:00:00-05:00""/>
      <AmountPercent TaxInclusive="false" CurrencyCode="USD" Amount="50.00"/>
    </CancelPenalty>
  </CancelPolicies>
</Rate>
```

5.2.7 *Use Case: Successful response*

```xml
<OTA_HotelRatePlanNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelRatePlanNotifRS.xsd" EchoToken=""
TimeStamp="2015-10-01T09:41:59.982" Version="1.001" MessageContentCode="8">
  <Success/>
</OTA_HotelRatePlanNotifRS>
```

5.2.8 *Use Case: Successful response with warnings*

```xml
<OTA_HotelRatePlanNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelRatePlanNotifRS.xsd" EchoToken=""
TimeStamp="2015-10-01T09:41:59.982" Version="1.001" MessageContentCode="8">
  <Success/>
  <Warnings>
    <Warning Type="3" Code="356" RecordID="2"/>
  </Warnings>
</OTA_HotelRatePlanNotifRS>
```

5.2.9 *Use Case: Error response*

```xml
<OTA_HotelRatePlanNotifRS xmlns="http://www.opentravel.org/OTA/2003/05" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opentravel.org/OTA/2003/05/OTA_HotelRatePlanNotifRS.xsd" EchoToken=""
TimeStamp="2015-10-01T09:41:59.982" Version="1.001" MessageContentCode="8">
  <Errors>
    <Error Type="3" Code="249" RecordID="2"/>
  </Errors>
</OTA_HotelRatePlanNotifRS>
```

5.3 OTA_HotelRatePlanNotifRQ Attribute Descriptions

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@TimeStamp	The instant of message creation. **Parent XPath:** /OTA_HotelRatePlanNotifRQ	DateTime
@EchoToken	Simple string returned verbatim in the response as passed in the request. Can be used to match requests with responses to simulate session state. **Parent XPath:** /OTA_HotelRatePlanNotifRQ	String
@Version	Mandatory OTA attribute. **Parent XPath:** /OTA_HotelRatePlanNotifRQ	Decimal
@MessageContentCode	The attribute refers to OTA code list MCC which includes Rate, Package and Group. This value is always "8" for Rate. **Parent XPath:** /OTA_HotelRatePlanNotifRQ	Integer
RatePlans		
@HotelCode	The code that uniquely identifies a single hotel property. This corresponds to the SynXis CRS Hotel ID value. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans	String
RatePlan		
@RatePlanCode	The code that uniquely identifies a single hotel rate type. This corresponds to the SynXis CRS Rate Code value. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan	String
@RatePlanNotifType	This enumeration indicates whether the upload/download overwrites the existing data or adds to it. **Valid value:** "Delta" **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan	String

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@Start	The start date of the rate update. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan	DateTime
@End	The end date of the rate update. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan	DateTime
UniqueID		
@Type	Refers to OTA code list UIT. "16" (Reference) is always sent. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/UniqueID	Integer
@ID	A unique incremental number for each RatePlan element that identifies the specific rate update. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/UniqueID	Integer
DestinationSystemsCode		
DestinationSystemCode	The destination system code defines a system to which information is to be provided. This corresponds to the SynXis CRS Channel Code value. All updates in a message are for the same channel which is defined by this code. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/DestinationSystemsCode	String
Rates		
Rate		
@RateTimeUnit	Set value to "FullDuration". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate	String
@MinLOS	Indicates the minimum Length of Stay of ARI data that is processed. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate	String

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@MaxLOS	Indicates the maximum Length of Stay of ARI data that is processed. Supports a maximum value of "8". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate	String
@InvTypeCode	Code that uniquely identifies a single hotel room type. This corresponds to the SynXis CRS Room Code value. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate	String
@CurrencyCode	The currency associated with the rate that is updated. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate	String
@Sun, Mon, Tue, Weds, Thur, Fri, Sat	Value for each day of week for the dates included in the @Start and @End values. This data is informational only for this message. For example, if the RatePlan element includes Start="2020-10-10" and End="2020-10-14, the day of week flags are set as follows: Mon="true" Tue="true" Weds="true" Thur="false" Fri="false" Sat="true" Sun="true". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate	Boolean
BaseByGuestAmts		
BaseByGuestAmt	This element is included for each adult occupancy value, as follows: If the applicable room has a guest limit value of 1 or 2, 2 BaseByGuestAmt elements are always included in the OTA_HotelRatePlanNotifRQ message to communicate single and double occupancy. **Note:** Guest Limit is calculated as: [Guest per room value + 1 if 'Allow Extra Bed' is set to true]. If the applicable room has a guest limit value of 3, 4, 5 or [x], then 3, 4, 5 or [x] BaseByGuestAmt elements are included in the OTA_HotelRatePlanNotifRQ message. If child pricing is enabled, this element is also included for each child age range (if using child age range pricing) or for each child occupancy supported by a room type (if using child occupancy pricing), or for a combination of both. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts	

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@AgeQualifyingCode	Defines an age range or age category of a guest (e.g. under 21, over 65, teen, infant). Refer to OTA Code List Age Qualifying Code (AQC). "10" (adult) or "8" (child) is always sent. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt	String
@NumberOfGuests	The number of guests associated with this rate amount. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt	Integer
@AmountAfterTax	The nightly rate amount including nightly taxes and fees. When LOS Pricing is in use, the value used for this attribute is the Total 'after taxes' amount for the entire stay. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt	Decimal
@AmountBeforeTax	The nightly rate amount not including any taxes or fees. When LOS Pricing is in use, the value used for this attribute is the Total 'before taxes' amount for the entire stay. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt	Decimal
@MinAge	Used for child pricing. Indicates the minimum age of the child age range that the pricing is for. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt	String
@MaxAge	Used for child pricing. Indicates the maximum age of the child age range that the pricing is for. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt	String
Taxes	This element is included if the Include Tax Configuration is enabled for a given ARI switch and when one or more taxes/fees were applied to the adult pricing or child offset pricing being communicated. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt	
@Amount	The total amount of taxes and fees applied to the adult pricing or to the child offset price in the BaseByGuestAmt element. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes	Decimal

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
Tax	A Tax element is included for each tax or fee which was applied to the adult pricing or child offset pricing which is communicated for each date it applies. Under the following conditions the multiple Tax elements are consolidated and the Tax/@EffectiveDate and Tax/@ExpireDate values specify the consolidated date range. • The dates specified in the Tax elements are consecutive, • The same unique (by GUID) SynXis CRS tax or fee is specified, AND • The same Tax/@Amount value is specified. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes	

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@Code	OpenTravel FTT (fee tax type) code for the tax type configured for the tax or fee in SynXis CRS. **Valid Values:** "1" - (Bed Tax) "2" - (City Hotel Fee) "3" - (City Tax) "4" - (County Tax) "5" - (Energy Tax) "6" - (Federal Tax) "7" - (Food & Beverage Tax) "8" - (Lodging Tax) "9" - (Maintenance Fee) "27" - (Miscellaneous Fee / Miscellaneous Tax) "10" - (Occupancy Tax) "11" - (Package Fee) "12" - (Resort Fee) "28" - (Room Tax) "13" - (Sales Tax) "14" - (Service Charge) "15" - (State Tax) "16" - (Surcharge) "17" - (Total Tax) "18" - (Tourism Tax) "19" - (VAT/GST Tax) **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes/Tax	String
@Amount	The amount of tax or fee applied to the adult pricing or to the child offset price for the date. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes/Tax	Decimal

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@EffectiveDate	The date the charge was applied for Per Night charges and package charges or the arrival date for Per Stay charges (format = YYYY-MM-DD). **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes/Tax	Date
@ExpireDate	The date the charge was applied for Per Night charges and package charges or the last night of the stay for Per Stay charges (format = YYYY-MM-DD). **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes/Tax	Date
@ChargeFrequency	OpenTravel CHG (charge type) code of "1" (Daily) for Per Night charges and package charges or "12" (Per Stay) for Per Stay charges. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes/Tax	String
TaxDescription		
@Name	Determines whether the associated TaxDescription/Text element is the name for the tax or fee or a description for it. **Valid Values:** • "Name" - the Text element is a tax or fee name. • "Description" - the Text element is a tax or fee description. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes/Tax/ TaxDescription	String
Text	Depending on the value of the @Name attribute either the default tax/fee name or description. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes/Tax/ TaxDescription	
TPA_Extensions		
AdditionalConfigurations		

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@Inclusive	Indicates if a tax/fee is inclusive or exclusive. **Valid Values:** "true" if the tax or fee mapped is inclusive, "false" otherwise. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/BaseByGuestAmts/BaseByGuestAmt/Taxes/Tax/TPA_Extensions/AdditionalConfigurations	Boolean
CancelPolicies	A CancelPolicies element is <u>not</u> included in the Rate element when: - Cancel Type for the Cancel Policy is "Never Charge Cancel Fee", OR - Cancel Charge Type for the Cancel Policy is "None". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate	
CancelPenalty	The cancel policy communicated is the cancel policy that is applicable to the product stay that is included in the Rate element. When the cancel policy has the Staggered Penalty Charge option enabled, the staggered charge which has the <u>largest</u> cancel charge value is used. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies	
@PolicyCode	The code for the cancel policy. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty	String
AmountPercent		
@TaxInclusive	Indicates whether the charge includes tax. **Valid Values:** "true" if the Charge Includes Tax option is enabled for the cancel policy, "false" otherwise. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/AmountPercent	Boolean

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@NmbrOfNights	The number of nights for penalty charge or cancel charge for staggered charge. Included when the cancel charge type for cancel policy is "# nights of stay". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/AmountPercent	Integer
@Percent	The percentage of amount of stay or cancel charge for staggered charge. Included when the cancel charge type for cancel policy is "% of amt of stay". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/AmountPercent	Decimal
@Amount	The amount in local currency or cancel charge for staggered charge. Included when the cancel charge type for cancel policy is "local currency". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/AmountPercent	Decimal
@CurrencyCode	The currency code that is currently included in the Rate element. Included when the cancel charge type for cancel policy is "local currency". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/AmountPercent	String
Deadline	This element is included when the cancel type for cancel policy is <u>not</u> "Always Charge Cancel Fee". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty	
@AbsoluteDeadline	Date/Time (in hotel's time zone) for the cancel penalty deadline. Included when the cancel policy has a cancel type of "Charge Within Time" or "Charge Within Days/Time". Format: YYYY-MM-DDTHH:MM:SS[+/-]HH:MM Example: 2019-08-10T11:00:00-05:00 **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/Deadline	String

Element/Attribute OTA_HotelRatePlanNotifRQ	Description / XPath	Type
@OffsetTimeUnit	The unit of time that applies to the deadline. Included when the cancel policy has a cancel type of "Charge Within Days" or "Charge Within Hours". **Valid Values:** • "Day" when cancel type is "Charge Within Days". • "Hour" when cancel type is "Charge Within Hours". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/Deadline	String
@OffsetUnitMultiplier	The number of units of @OffsetTimeUnit. Included when the cancel policy has a cancel type of "Charge Within Days" or "Charge Within Hours". **Valid Values:** • Days value from cancel policy when cancel type is "Charge Within Days". • Hours value from cancel policy when cancel type is "Charge Within Hours". **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/Deadline	Integer
@OffsetDropTime	Indicates when the deadline drop time goes into effect. Included when the cancel policy has a cancel type of "Charge Within Days" or "Charge Within Hours". **Valid Value:** "BeforeArrival" **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/CancelPolicies/CancelPenalty/Deadline	String
MealsIncluded	Provides information about the meal plan and the meals included in the rate. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate	
@Breakfast	Indicates if breakfast is included in the rate. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/MealsIncluded	Boolean
@Lunch	Indicates if lunch is included in the rate. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/MealsIncluded	Boolean

Element/Attribute **OTA_HotelRatePlanNotifRQ**	Description / XPath	Type
@Dinner	Indicates if dinner is included in the rate. **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/MealsIncluded	Boolean
@MealPlanCodes	OpenTravel MPT (meal plan type) code to indicate the type of meal plan that is selected for the rate. If no meal plan is selected for the rate, then this attribute is not included. **Valid Values:** "1" - (All inclusive) "2" - (American/Full board) "3" - (Bed & breakfast) "16" - (Bermuda) "19" - (Breakfast) "4" - (Buffet breakfast) "5" - (Caribbean breakfast) "6" - (Continental breakfast) "17" - (Dinner bed and breakfast plan) "7" - (English breakfast) "11" - (Full breakfast) "12" - (Half board / Modified American plan) **Parent XPath:** /OTA_HotelRatePlanNotifRQ/RatePlans/RatePlan/Rates/Rate/MealsIncluded	String

5.4 OTA_HotelRatePlanNotifRS Attribute Descriptions

Element/Attribute OTA_HotelRatePlanNotifRS	Description / XPath	Type	Required
@TimeStamp	The instant of message creation. **Parent XPath:** /OTA_HotelRatePlanNotifRS	DateTime	Yes
@EchoToken	Simple string returned verbatim in the response as passed in the request. Can be used to match requests with responses to simulate session state. *Required if a value is sent in the request. **Parent XPath:** /OTA_HotelRatePlanNotifRS	String	*Yes
@Version	Mandatory OTA attribute. **Parent XPath:** /OTA_HotelRatePlanNotifRS	Decimal	Yes
@MessageContentCode	The attribute refers to OTA code list MCC which includes Rate, Package and Group. This value is always "8" for Rate. **Parent XPath:** /OTA_HotelRatePlanNotifRS	Integer	Yes
Success	If included, this element indicates that the request message was successfully processed. *Either a Success element or Errors/Error element(s) are required in every response. **Parent XPath:** /OTA_HotelRatePlanNotifRS		*Yes
Warnings	Warnings indicate issues which did not prevent processing.		
Warning	Description of cause for a non-fatal problem during request message processing. *If a Warnings element is included one or more Warning elements are required. **Parent XPath:** /OTA_HotelRatePlanNotifRS/Warnings	String	*Yes

Element/Attribute OTA_HotelRatePlanNotifRS	Description / XPath	Type	Required
@Type	This is an enumeration of HTNG approved error types (OTA code list EWT). See Appendix for supported values. *This attribute is required if a warning element is included. **Parent XPath:** /OTA_HotelRatePlanNotifRS/Warnings/Warning	Integer	*Yes
@Code	This is an enumeration of HTNG approved error codes (OTA code list ERR). See Appendix for supported values. *This attribute is required if a warning element is included. **Parent XPath:** /OTA_HotelRatePlanNotifRS/Warnings/Warning	String	*Yes
@RecordID	If the receiving system identifies which RatePlan element(s) in the request had an issue the unique incremental value from the request's UniqueID/@ID is provided here. **Parent XPath:** /OTA_HotelRatePlanNotifRS/Warnings/Warning	Integer	
Errors	*Either a Success element or Errors/Error element(s) are required in every response.		*Yes
Error	Description of cause for a fatal problem during request message processing. *If an Errors element is included one or more Error elements are required. **Parent XPath:** /OTA_HotelRatePlanNotifRS/Errors	String	*Yes
@Type	This is an enumeration of HTNG approved error types (OTA code list EWT). See Appendix for supported values. *This attribute is required if an error element is included. **Parent XPath:** /OTA_HotelRatePlanNotifRS/Errors/Error	Integer	*Yes
@Code	This is an enumeration of HTNG approved error codes (OTA code list ERR). See Appendix for supported values. *This attribute is required if an error element is included. **Parent XPath:** /OTA_HotelRatePlanNotifRS/Errors/Error	String	*Yes

Element/Attribute **OTA_HotelRatePlanNotifRS**	Description / XPath	Type	Required
@RecordID	If the receiving system identifies which RatePlan elements in the request have issues the unique incremental value from the request's UniqueID/@ID is entered here. **Parent XPath:** /OTA_HotelRatePlanNotifRS/Errors/Error	Integer	

6 Soap Faults: Synchronous Communication Errors

This section provides the message examples used to communicate errors that occur prior to processing a request message.

6.1 Soap Fault Message Description

- A Soap Envelope containing a Soap Body with a Soap Fault should be returned in place of a "NotifRS" response message in the Synchronous workflow when an error occurs prior to the point of processing the request message. These failures would include communication errors, incorrectly formed xml, etc.

- SynXis CRS supports SOAP version 1.2 including SOAP Faults. For a full explanation of SOAP Faults, please refer to the W3C website: https://www.w3.org/TR/2007/REC-soap12-part0-20070427/

6.2 Soap Fault Use Case

6.2.1 *Use Case: Synchronous Soap Fault Response*

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:wss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:syx="http://synxis.com/ws/2009/10/">
  <soap:Header>
    <wsa:MessageID>9971BF72-F42D-49ee-99DB-BEC28B6EDAF7</wsa:MessageID>
    <wsa:RelatesTo>214A50B2-E62E-4f8b-BD97-62ABE31E15C2</wsa:RelatesTo>
    <wsa:To>https://client/ARI_Synch_Service</wsa:To>
    <wsa:Action>https://client/ARI_Synch_Service/ARI_Synch</wsa:Action>
  </soap:Header>
  <soap:Body>
    <soap:Fault>
      <soap:Code>
        <soap:Value>soap:Sender</soap:Value>
        <soap:Subcode>
          <soap:Value>syx:Authentication</soap:Value>
        </soap:Subcode>
      </soap:Code>
      <soap:Reason>
        <soap:Text>Invalid Password</soap:Text>
      </soap:Reason>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

6.3 Soap Fault Attribute Descriptions

Element/Attribute	Description / XPath	Type	Required
Fault			
Code			
Value	Enumeration of Soap Fault codes. The type of error that is returned determines what action is taken. **Valid values:** • VersionMismatch • MustUnderstand • DataEncodingUnknown • Sender • Receiver **Parent XPath:** /Fault/Code	String	Yes
Subcode			No
Value	Additional SynXis CRS generated fault codes. When possible the appropriate SynXis CRS fault should be returned. SynXis CRS determines the action to take depending upon the specific error that is returned. *This is required if the optional Subcode element is included. **Parent XPath:** /Fault/Code/Subcode	String	*Yes
Reason			
Text	Free form description for the cause of the error. This is used for informational purposes in diagnosing system issues. **Parent XPath:** /Fault/Reason	String	Yes

Appendix: Certification Process

Third parties who wish to use the method calls provided in this specification must first complete an integration certification process with Sabre Hospitality (SH).

The following steps outline the process:

1. SH provides the vendor with a non-disclosure agreement (NDA).

2. SH discusses possible integration options with the vendor/customer.

3. The vendor signs and returns the NDA to the SH product manager, who then forwards the document to our legal team for review.

4. After the NDA is approved, SH provides the vendor with a copy of the applicable interface specification.

5. SH provides the vendor with the test property/properties on our certification platform. A property summary is provided, which includes basic configuration and connection details.
   The vendor can use the SynXis CRS test properties to post API messages for development, internal QA testing, and certification. Channel partners have interface-only access to the test properties.
   Note: If you have questions, please send an email to: SHSIntegrationCertification@sabre.com

6. The vendor develops the message formats to be certified. Responses should include both a success and an error format.

7. The vendor tests the interface.

8. Once the vendor completes the development and initial testing, they submit the samples of message formats for review:
   - RQ and RS files with soap envelope for Synch processing APIs;
   - RQ, RQ_ACK, RS and RS_ACK files with soap envelope for Asynch processing APIs.
   Note: The files must be sent in a .zip file, using the following naming convention: ***xxxxx*.sabre.zip**. Each file should contain a single xml and the name of the file should clearly indicate the use case and step it pertains to. Please email the files to: SHSIntegrationCertification@sabre.com

9. SH reviews the certification files received and responds to the vendor with the results of the review.

10. SH schedules end-to-end certification testing with an SH team member and the vendor.

11. SH and the vendor perform certification testing. Certification testing consists of the vendor posting messages to / receiving messages from our certification system and an SH team member verifying those messages.
    Note: SH provides the vendor with a sample test plan.

12. Upon successful completion of the certification testing, SH and the vendor contact the beta customer to schedule the implementation at the agreed-upon pilot property.

13. SH and the vendor implement the interface at the agreed-upon pilot property.

14. After the implementation, SH and the vendor monitor the pilot property for a minimum of two weeks.

15. If no issues arise after two weeks of successful production testing at the beta site, SH considers the interface certification process complete.
    Additional remarks:
    - At each stage of the certification process, address any outstanding issues before progressing to the next step.
    - Any expressed timelines are approximate and are subject to change.
    - The certification environment is subject to frequent development updates. Brief system outages might necessarily occur on occasion without prior notification.

Appendix: Supported Error/Warning Codes

This section provides supplemental information that may be necessary as you build the Generic ARI messages.

Supported Error Warning Types (OTA code list "EWT") for use with OTA_HotelAvailNotifRS and OTA_HotelRatePlanNotifRS messages.

1	Unknown
2	No implementation
3	Biz rule
4	Authentication
5	Authentication timeout
6	Authorization
7	Protocol violation
8	Transaction model
9	Authentical model

Supported Error Codes (OTA code list "ERR") for use with OTA_HotelAvailNotifRS and OTA_HotelRatePlanNotifRS messages.

Enumeration	Description
15	Invalid Date
135	End Date is Invalid
136	Start Date is Invalid
187	System Currently Unavailable
249	Invalid Rate Code
320	Invalid Value
321	Required Field Missing
356	Invalid Action/Status Code
361	Invalid Hotel
400	Invalid Property Code
402	Invalid Room Type
404	Invalid Start/End Date Combination
447	Unable to Update – Simultaneous Updates
448	System Error
450	Unable To Process