

Instant

Instant Mishor

(XML interface)

V3.40

1. Introduction

This documentation describes the Mishor server's XML interface, which allows interaction with the core reservation engine. The documentation is intended for individuals who are experienced with programming in the Internet environment:

- Developers of systems that need to retrieve products and services from Instant Travel.

Due to technological advancements in the e-commerce field and ongoing enhancements to Mishor, this documentation is subject to change. Tags may be added and the order of existing tags may be changed during version updates.

In future changes, we will do our utmost to maintain backward compatibility with previous specifications. Backward compatibility implies that all existing tags and their level will be maintained.

This documentation will be updated as required and redistributed to relevant parties.

2. Static Data

Static data dumps of countries, cities and hotels are available and will be supplied via your account manager

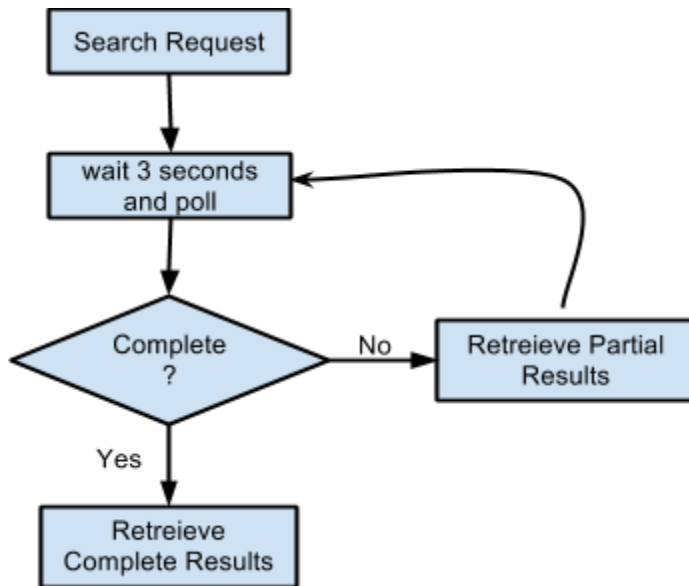
3. Authentication

Authentication is done by 2 measures:

1. Whitelisted IP - submitted during the development and certification process
2. XML node with a username and password - also provided during the development and certification process

4. Search Flow

Search is done using the “poll” methodology, allowing quicker response times, and providing the user with the flexibility to display results within a very short timeframe. This is done by sending a request that is responded immediately with a search token. this token is then used to “poll” the system for available results and see if the overall search is done:

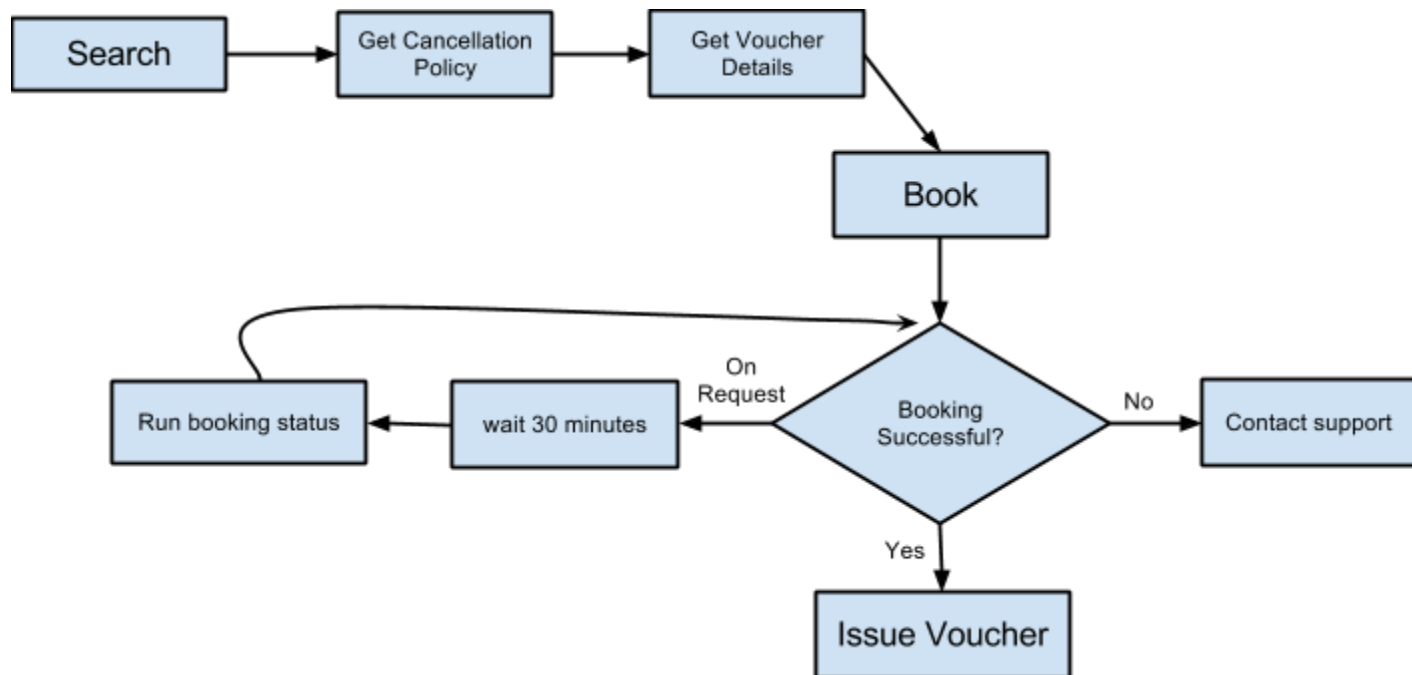


5. Booking Flow

After collecting the search results, and before running the booking request it is mandatory to run a "get cancellation policy" request to receive the latest cancellation policy and additional remarks. It is also mandatory to run the "Get voucher details" call in order to get the exact information to put on the voucher. Once this is done a booking request can be sent.

If the booking is successful, you can go ahead and issue a voucher to the client, if it fails - please contact our support department. However, there is a 3rd possibility: in some cases, when a booking is not approved and not rejected, it will be put "on request". In this case, it is your responsibility to run a booking status request every 30 minutes or more to check whether the booking was approved or rejected.

This process is pictured in the image below:



6. City Search

The city search does not return results, it returns a search token that is used for “city poll”.

6.1 Request

```
<request>
  <auth>
    <username>username</username>
    <password>123456</password>
    <agent>agent</agent>
  </auth>
  <method>search</method>
  <param name='checkInDate'>2011-04-16</param>
  <param name='checkOutDate'>2011-04-17</param>
  <param name='onRequest'>1</param>
  <param name='locationId'>4223</param>
  <param name='customerCountryCode'>CA</param>
  <param name='language'>en</param>
  <param name='inventory'>1</param>
  <currencies mode="restricted">
    <currency title="EUR" default="1" />
    <currency title="GBP" />
  </currencies>
  <room>
    <guest type='A' count='2' />
    <guest type='C' count='2' age='14' />
  </room>
  <room>
    <guest type='A' count='2' />
    <guest type='C' count='2' age='14' />
  </room>
  <room>
    <guest type='A' count='2' />
  </room>
</request>
```

Parameter	Type	Values	Comments
method	String (6)	Search	
checkInDate	Date	YYYY-MM-DD	
checkOutDate	Date	YYYY-MM-DD	
onRequest	Boolean	0 or 1	Include on request

			rooms (1) or immediate confirmation only
locationId	Int	City Id	
customerCountryCode	String (2)	2 letter ISO Code	
language	String (2)	2 letter language code	
inventory	Int	1 or 2	1 for B2C, 2 for B2B
currencies @mode	String	Restricted / Transaction	if mode is Restricted - currency nodes are mandatory. otherwise not mandatory.
currency @title	Currency		
currency @default	Boolean		This attribute should be set to 1 only on one currency

Currency handling:

The system allows you to either limit your results to specific currencies, or get the results in the original transaction currency. by default, the system uses “Transaction” mode to pass on the hotel currency. if you can only accept certain types of currencies, you can set the mode to “Restricted” and specify your supported currencies. If you have more than 1 currency set in restricted mode, you must include the “default” attribute and set it to 1 on at least one currency.

Rooms Structure:

Each room is a separate node, specifying the guests in each room. Each guest type (Adult or Child) has an amount (“count”) and children also have an “age” property.

For example: to search for a room with 1 adult and 3 children ages 5,5 and 7 the room node should look like:

```
<room>
  <guest type='A' count='1' />
  <guest type='C' count='2' age='5' />
  <guest type='C' count='1' age='7' />
</room>
```

6.1.1 Simple search with no polls

If for some reason you do not want to employ our polling mechanism to get quick results, you

Instant

can send the following parameters:

```
<param name='noPolls'>1</param>
<param name='searchTimeout'>10</param>
```

The “noPolls” will tell the system to keep the connection open (i.e. not return a search token) until the “searchTimeout” expired, then it will return the full set of results exactly like the “poll response”. searchTimeout is defined in Seconds.

6.2 Response

```
<mishor>
  <job>
    <jobid>d1.TeNAhwCm8A62c</jobid>
    <status>working</status>
    <lastUpdate>2011-08-16 13:40:23</lastUpdate>
    <server>http://mishor.server.com/service</server>
  </job>
</mishor>
```

The response will return the following information:

Parameter	Type	Values	Comments
jobid	String (16)		
status	String	“working” OR “done”	
lastUpdate	Timestamp		
server	URI		the following “poll” result should be sent to the URI returned in this response

Note: You must use the server URI you receive in this response for ALL corresponding requests in the same session. failure to do so will result in errors

7. City Poll

Note: it is required to run a “city search” before running a city poll, in order to get the “session id”.

7.1 Request

```
<request sessionid="d1.TeNAhwCm8A">
  <auth>
    <username>username</username>
    <password>123456</password>
    <agent>agent</agent>
  </auth>
  <method>poll</method>
  <timestamp>2011-08-16 13:40:23</timestamp>
  <param name='hotelId'>0</hotelId>
  <param name="showproviders">0</param>
  <param name="shownonbillable">1</param>
  <param name="showsuggested">1</param>
  <param name="shownoncommissionable">1</param>
</request>
```

Parameter	Type	Values	Comments
request @sessionid		the same sessionid returned during the initial search	
method	String	“poll”	
timestamp	Timestamp	initially “lastupdate” value from search response, then “timestamp” value from previous poll	
@hotelId	Int	the id of the requested hotel	Should be 0 on city poll
@showproviders	Bool	show the provider names	
@shownonbillable	Bool	Show non-billable results	
@showsuggested	Bool	Show the suggested selling price	Default 1
@shownoncommissionable	Bool	Show non-commissionable results	

Note: most implementations do not need to use non-billable and non-commissionable flags

7.1.1 City search with room results

If you would like to receive room results along with the hotel results, add the following parameter to your poll request:


```
<param name="returnResults">4</param>
```

The response will be a mix between the city poll response and the hotel poll response, returning the city nodes AND the rooms/room nodes defined below.

Note: Using this method significantly hurts the performance of the system and will cause delays in the response.

7.2 Response

```
<?xml version="1.0" encoding="utf-8"?>
<mishor>
  <session id="d1.TeNDxb6ioA" status="working"/>
  <hotels DefaultCurrency="EUR">
    <hotel id="81" minsuggestedroomusdprice="1545.15"
minSuggestedRoomDefPrice="1201.7095095" minroomusdprice="1417.15"
minroomusdpricelimit="1404.68" minRoomDefPrice="1102.1600695"
minRoomDefPriceLimit="1092.4617764" boards="RO" minboard="RO" minusdpackageprice="0"
minPackageDefPrice="0" payatthehotel="0" />
    <hotel id="86" minsuggestedroomusdprice="701.1"
minSuggestedRoomDefPrice="545.266503" minroomusdprice="634.08"
minroomusdpricelimit="637.36" minRoomDefPrice="493.1430384"
minRoomDefPriceLimit="495.6939928" boards="RO" minboard="RO" minusdpackageprice="0"
minPackageDefPrice="0" payatthehotel="0" />
    <hotel id="89" minsuggestedroomusdprice="716.27"
minSuggestedRoomDefPrice="557.0646671" minroomusdprice="652.78"
minroomusdpricelimit="651.15" minRoomDefPrice="507.6865894"
minRoomDefPriceLimit="506.4188895" boards="RO,BB" minboard="RO"
minusdpackageprice="0" minPackageDefPrice="0" payatthehotel="0" />
  </hotels>
  <timestamp>2011-08-16 13:43:23</timestamp>
</mishor>
```

Parameter	Type	Values	Comments
session @status	String	"working"/"done"	
hotels @DefaultCurrency	Currency	the currency context	
hotel @id	Int		
hotel @minroomusdprice	Float	Lowest room price in USD	
hotel @minRoomDefPrice	Float	Lowest room price in the default currency	
hotel @boards	String	Available boards for this hotel	

Instant

hotel @minboard	String	The board of the room with the lowest price.	
hotel @minusdpackageprice	Float	Lowest package price in USD	
hotel @minPackageDefPrice	Float	Lowest package price in the default currency	
hotel @commissionable	Bool	Can a commission be added to the price	
hotel @minsuggestedroomusdprice	Float	Suggested sale price in USD	
hotel @minSuggestedRoomDefPrice	Float	Suggested sale price in the default currency	
hotel @minnoncommissionable	Float	Lowest non-commissionable price	0 if N/A
hotel @minusdroomnonbillableprice	Float	Lowest non-billable price	0 if N/A
hotel @minroomusdpricelimit	Float	Minimum selling price allowed in USD	Property cannot be sold below this price
hotel @minRoomDefPriceLimit	Float	Minimum selling price allowed in the default currency	
hotel @minprovidername	String	Lowest provider	
hotel @minpackageproviders	String	Comma separated provider list	

Polls can continue until the "status" is set to "done" or until the client decides they have enough results. It is not mandatory to keep polling until the "done" status is returned, but make sure you wait AT LEAST 3-5 seconds from receiving the response before running additional polls.

NOTE: each poll will return only new or updated results since the last "timestamp".

8. Hotel Search

Same as "City search" but includes the param "hotelId".

For example:

```
<param name='hotelId'>70457</param>
```

The response is identical to the one on the city search.

Note: it is not mandatory to run an additional hotel search after running city search for the same parameters. it is sufficient to run a "Hotel Poll" request with the required hotelId.

9. Hotel Poll

9.1 Request

```
<request sessionid="d1.TeNAhwCm8A">  
  <auth>  
    <username>username</username>  
    <password>123456</password>  
    <agent>agent</agent>  
  </auth>  
  <method>poll</method>  
  <param name='hotelId'>1234</param>  
</request>
```

9.2 Response

```
<?xml version="1.0" encoding="utf-8"?>  
<mishor>  
  <session id="a1.TkvGOhuYkA49b" status="working" />  
  <hotels>  
    <hotel id="88235">  
      <packages>  
        <package board="RO" usdprice="2758">  
          <rooms>  
            <room resultid="173805" roomsCount="1" />  
            <room resultid="174578" roomsCount="1" />  
          </rooms>  
          <cancellationPolicies>  
            <cancellationPolicy startDate="2011-09-19" endDate="2011-09-28" nightsPenalty="1"  
          />  
          </cancellationPolicies>  
          <onrequest>0</onrequest>  
        </package>  
        <package board="RO" usdprice="2830">  
          <rooms>  
            <room resultid="173804" roomsCount="1" />  
            <room resultid="174579" roomsCount="1" />  
          </rooms>  
        </package>  
      </packages>  
    </hotel>  
  </hotels>  
</mishor>
```

```
</rooms>
<cancellationPolicies>
  <cancellationPolicy startDate="2011-09-19" endDate="2011-09-28" nightsPenalty="1"
/>
  </cancellationPolicies>
</onrequest>0</onrequest>
</package>
<package board="RO" usdprice="3152">
  <rooms>
    <room resultid="173300" roomsCount="1" />
    <room resultid="174576" roomsCount="1" />
  </rooms>
  <cancellationPolicies>
    <cancellationPolicy startDate="2011-09-19" endDate="2011-09-28" nightsPenalty="1"
/>
  </cancellationPolicies>
</onrequest>0</onrequest>
</package>
</packages>
<rooms>
  <room>
    <guests>
      <guest type="A" count="2" />
    </guests>
    <roomname><![CDATA[Twin Standard]]></roomname>
    <cancellationPolicies>
      <cancellationPolicy startDate="" endDate="" nightsPenalty="" />
    </cancellationPolicies>
    <resultid>307870</resultid>
    <onrequest>0</onrequest>
    <price_suggested>4242</price_suggested>
    <boardname>Breakfast Buffet</boardname>
    <boardcode>BB</boardcode>
    <minrooms>0</minrooms>
    <billable>1</billable>
    <providerDisplayName>instanttravel</providerDisplayName>
    <commissionable>1</commissionable>
    <provider_price_minimum>0</provider_price_minimum>
    <price>3237.93</price>
    <maxrooms>1</maxrooms>
    <roomtypeid>de1cc8b7c4d8fccdc951cb8512fca630</roomtypeid>
    <usdprice>4163</usdprice>
    <currency>EUR</currency>
  </room>
  <room>
    <guests>
      <guest type="A" count="2" />
    </guests>
    <roomname><![CDATA[Double Standard]]></roomname>
    <cancellationPolicies>
      <cancellationPolicy startDate="" endDate="" nightsPenalty="" />
    </cancellationPolicies>
    <resultid>307873</resultid>
    <onrequest>0</onrequest>
    <price_suggested>4242</price_suggested>
    <boardname>Breakfast Buffet</boardname>
    <boardcode>BB</boardcode>
    <minrooms>0</minrooms>
    <billable>1</billable>
    <providerDisplayName>instanttravel</providerDisplayName>
```

```

<commissionable>1</commissionable>
<provider_price_minimum>0</provider_price_minimum>
<price>3237.93</price>
<maxrooms>1</maxrooms>
<roomtypeid>41037691c34ec152d5671e0c409f8d63</roomtypeid>
<usdprice>4163</usdprice>
<currency>EUR</currency>
</room>
</rooms>
</hotel>
</hotels>
</mishor>

```

9.2.1 General

Parameter	Type	Values	Comments
session @status	String	"working"/"done"	
hotel @id	Int		

9.2.2 Packages

Parameter	Type	Values	Comments
board	String	Board base for the package	
usdprice	Float	Package price in USD	
onrequest	Bool		
room @resultid	Int	The id of the room in the room list	
room @roomsCount	Int	How many rooms of the same room type	

9.2.3 Cancellation Policies

This section applies for both the packages and the rooms

Parameter	Type	Values	Comments
cancellationPolicy @startDate	Date	Start date of cancellation policy rule	
cancellationPolicy	Date	End date of cancellation	

@endDate		policy rule	
cancellationPolicy @nightsPenalty	Int	Number of nights charged as penalty	

Note: It is possible for a room to have multiple cancellation policies for different dates.

Note: It is possible that no cancellation policies will be returned. In this case, a separate “get cancellation policy” request is required before booking.

9.2.4 Rooms

Parameter	Type	Values	Comments
guests	Node		to which guests does the room apply
roomname	String	The full name of the room	
resultid	Int	the ID of the room	
onrequest	Bool	0=immediate confirmation	
boardname	String	Full board name	
boardcode	String	Normalized 2 letter code	
minrooms	Int	Minimum rooms allowed for booking	
maxrooms	Int	Maximum rooms allowed for booking	
billable	Bool		
providerDisplayName	String	The name of the provider from which the booking will be made	
commissionable	Bool	Whether the price is the final price or can be marked up	
price	Float	The price of the room, in @currency	
currency	String	Currency ISO code	
usdprice	Float	Price in USD	mainly used to make

Instant

			sorting easier
roomtypeid	String	room code	
price_suggested	Float	The suggested selling price in @currency	
provider_price_minimum	Float	The minimum selling price in @currency	

10. Booking

10.1 Request

```
<request sessionid='t.TTRPBL6AQA'>
  <auth>
    <username>mishor</username>
    <password>123456</password>
    <agent>i am an agent</agent>
  </auth>
  <method>reserve</method>
  <param name='checkInDate'>2011-04-16</param>
  <param name='checkOutDate'>2011-04-17</param>
  <param name='clientRef'>123</param>
  <param name='customerCountryCode'>UK</param>
  <customer firstName='Ozzy' lastName='Osborne' title='Mr' />
  <room resultId='282236' ord='1'>
    <guests>
      <guest firstName='Ozzy' lastName='Osborne' title='M' age='33' />
      <guest firstName='Ozzy a' lastName='Osborne' title='M' age='33' />
    </guests>
  </room>
  <room resultId='282235' ord='2'>
    <guests>
      <guest firstName='Ozzy' lastName='Osborne' title='M' age='33' />
      <guest firstName='Ozzy a' lastName='Osborne' title='M' age='33' />
    </guests>
  </room>
  <remarks>
    <remark><![CDATA[testremark]]></remark>
    <remark><![CDATA[remarktest]]></remark>
  </remarks>
</request>
```

Note: it is only possible to book one hotel in each reservation request. All resultId's should be from the same hotelID.

10.1.1 Rooms

Parameter	Type	Values	Comments
request @sessionid	String	the same sessionid returned during the initial search	
method	String	reserve	
checkInDate	Date	Check in date	
checkOutDate	Date	Check out date	
clientRef	String	your internal reference string	

customerCountryCode	String	2 letter country ISO	
customer @firstName	String	Lead pax first name	
customer @lastName	String	Lead pax last name	
room @resultid	Int	the room ID from the search result	
room @ord	Int	Ordinal number of the room in the reservation	
guest @firstName	String	first name of the guest in the room	
guest @lastName	String	last name of the guest in the room	
guest @title	String	see Guest Title lookup for possible values	
guest @age	Int	the age of the guest	mandatory for children, adults can have any age above 21.
remark	String	free text	

10.2 Response

10.2.1 Successful

```

<response status="ok">
  <booking status="ok" retry="false">
    <pending>0</pending>
    <remarks><![CDATA[ ** Please be aware, that each individual guest will be
responsible for incidental charges, to be paid upon departure. To guarantee these
charges, we require a credit card at time of check-in.We will be unable to register
any guest that is not able to provide sufficient credit to cover their charges upon
arrival.**]]></remarks>
    <room resultid='282236' reservationId='81' aetherref='205147'>
      <reservationReference>11365459</reservationReference>
    </room>
  </booking>
  <booking status="ok" retry="false">
    <pending>0</pending>
    <remarks><![CDATA[ ** Please be aware, that each individual guest will be
responsible for incidental charges, to be paid upon departure. To guarantee these
charges, we require a credit card at time of check-in.We will be unable to register

```

```

any guest that is not able to provide sufficient credit to cover their charges upon
arrival.**]]></remarks>
  <room resultid='282235' reservationId='82' aetherref='205147'>
    <reservationReference>11365460</reservationReference>
  </room>
</booking>
</response>

```

Parameter	Type	Values	Comments
booking @status	String	ok / fail	
booking @retry	Bool		
pending	Bool	is booking “on request”	if value is set to 1, it is mandatory to run a “booking status” request
remarks	String		remarks MUST appear of the voucher issued to the client
room @reservationId	Int	system reference number of the booking	this element will be used to cancel or amend the booking
room @aetherref	Int	reference in the agent system	this will be used for communication with the FIT team
room->reservationReference	String		this element must appear on the voucher issued to the client

Note: When booking more than 1 room, in some cases you will get more than 1 booking node, like in the example above.

In case of failure, the booking @status attribute will be set to “fail” and an additional “error” node will be added with a description of the error.

Important note: it is possible in some cases that when booking multiple rooms some will fail and some will succeed. it is mandatory to check the booking @status for every booking node.

11. Cancellation Policy

11.1 Request

```
<request sessionid='t.TXTkHJpMA'>
  <auth>
    <username>mishor</username>
    <password>123456</password>
    <agent>aether</agent>
  </auth>
  <method>cancellationpolicy</method>
  <room resultid='76382035' />
</request>
```

Parameter	Type	Values	Comments
request @sessionid	String	the same session id returned during the initial search	
method	String	cancellationpolicy	
room @resultid	Int	the room id to get the cancellation policy for	

Note: you can send only 1 room node in the cancellation policy request. To check more than 1 room, you will need to send multiple requests.

11.2 Response

```
<response>
  <room resultid='183877'>
    <cancellationPolicies>
      <cancellationPolicy startDate="2011-09-15" endDate="2011-09-28" nightsPenalty="1" />
    </cancellationPolicies>
  </room>
  <remarks><![CDATA[ remarks remarks ]]></remarks>
</response>
```

Parameter	Type	Values	Comments
room @resultid	Int	the room id	

Instant

cancellationPolicy @startDate	Date	Start date of cancellation policy rule	
cancellationPolicy @endDate	Date	End date of cancellation policy rule	
cancellationPolicy @nightsPenalty	Int	Number of nights charged as penalty	
remarks	String		

Note: It is possible for a room to have multiple cancellation policies for different dates.

Note: It is possible to receive multiple remarks nodes, all remarks must appear on the client voucher

13. Get Voucher Details

This call is mandatory. it should be run before the booking request and the response details should be on the voucher issued to the client.

13.1 Request

```
<request sessionid='a1.ToAucZ0MAaf3'>
  <auth>
    <username>mishor</username>
    <password>123456</password>
    <agent>i am an agent</agent>
  </auth>
  <method>voucher-data</method>
  <resultids>
    <resultid>1</resultid>
    <resultid>2</resultid>
  </resultids>
</request>
```

Parameter	Type	Values	Comments
method	String	voucher-data	
resultids @resultsid	Int	the Id of the result about to be booked	could be multiple nodes

13.2 Response

```
<response>
  <splitbookings>1</splitbookings>
  <resultids>
    <resultid id="1">
      <name><![CDATA[Campanile]]></name>
      <address><![CDATA[ZONE D'ACTIVITES DU PARC DE ROISSY ALLEE DES VERGERS
      ROISSY EN FRANCE]]></address>
      <phone><![CDATA[33-+33-1-34-29-80-40]]></phone>
      <fax><![CDATA[33-+33-1-34-29-80-39]]></fax>
      <rating><![CDATA[2]]></rating>
    </resultid>
    <resultid id="2">
      <name><![CDATA[Campanile]]></name>
      <address><![CDATA[ZONE D'ACTIVITES DU PARC DE ROISSY ALLEE DES VERGERS
      ROISSY EN FRANCE]]></address>
```

```
<phone><![CDATA[33-+33-1-34-29-80-40]]></phone>  
<fax><![CDATA[33-+33-1-34-29-80-39]]></fax>  
<rating><![CDATA[2]]></rating>  
</resultid>  
</resultids>  
</response>
```

Parameter	Type	Values	Comments
splitbookings	Bool	Should the bookings be split into multiple bookings	
result @id	Int	the id reference	
name	CDATA String	the name of the property	
address	CDATA String	the address of the property	
phone	CDATA String	the phone number of the property	
fax	CDATA String	the fax number of the property	
rating	CDATA String	the star rating of the property	

Note:

1. It is mandatory to send all of the result ids in one "voucher-data" call
2. It is mandatory to display these details on the voucher issued to the client and NOT the details on the static feed.
3. if the "splitbookings" node value is 1, it is mandatory to create multiple booking requests. some of the bookings may fail, so use a careful business logic to decide whether or not to allow going through with the booking, as some or all of the bookings may be non-refundable.

14. Cancel booking

14.1 Request

```
<request>
  <auth>
    <username>mishor</username>
    <password>123456</password>
    <agent>i am an agent</agent>
  </auth>
  <method>cancel</method>
  <reservations>
    <reservationId>82</reservationId>
    <reservationId>83</reservationId>
  </reservations>
</request>
```

Parameter	Type	Values	Comments
method	String	cancel	
reservationId	Int	the Id of the reservation to be cancelled	could be multiple nodes

14.2 Response

```
<response>
  <cancellation reference='2' reservationId='82' status='ok'>
    <penalty nights='1'/>
  </cancellation>
  <cancellation reference='3' reservationId='83' status='ok'>
    <penalty nights='1'/>
  </cancellation>
</response>
```

Parameter	Type	Values	Comments
cancellation @reference	Int	reference number for the cancellation request	
cancellation @reservationId	Int	the Id of the reservation that was cancelled	

Instant

cancellation @status	string	ok / fail	
penalty @nights	Int	the number of nights charged as penalty according to the current cancellation rule	

Note: It is possible that some rooms will be cancelled and some will not.

15. Amendments

15.1 Request

15.1.1 General amendment request

```
<request>
  <auth>
    <username>mishor</username>
    <password>123456</password>
    <agent>i am an agent</agent>
  </auth>
  <method>amend</method>
  <reservations>
    <reservation id="284">
      <param name="clientref">dude</param>
      <param name="hotelid">123</param>
      <param name="checkin">2011-06-03</param>
      <param name="checkout">2011-06-05</param>
      <param name="status">2</param>
      <param name="comments">special request</param>
      <param name="boardType">BB</param>
      <guests ord="1">
        <guest firstName="Ozzy a" lastName="Osborne a" title="M" age="33" />
        <guest firstName="Ozzy b" lastName="Osborne b" title="M" age="33" />
      </guests>
      <guests ord="2">
        <guest firstName="Ozzy a" lastName="Osborne a" title="M" age="33" />
        <guest firstName="Ozzy b" lastName="Osborne b" title="M" age="33" />
      </guests>
    </reservation>
  </reservations>
</request>
```

Parameter	Type	Values	Comments
method	String	amend	
reservation @id	Int	the Id of the reservation to be amended	
clientref	String	your reference	
hotelid	Int	the hotel id to be	

Instant

		changed	
checkin	Date		
checkout	Date		
status	Int		see "reservation statuses" section
comments	String	Special requests	
boardType	String		see "board types" section
guests @ord	Int	ordinal number of the room	
guest	Node	guest information	

Note: it is recommended to send ONLY the "param" nodes that require change. for example, to change only the guest name in the first room, you should send:

```
<request>
  <auth>
    <username>mishor</username>
    <password>123456</password>
    <agent>i am an agent</agent>
  </auth>
  <method>amend</method>
  <reservations>
    <reservation id="284">
      <guests ord="1">
        <guest firstName="Ozzy a" lastName="Osborne a" title="M" age="33" />
        <guest firstName="Ozzy b" lastName="Osborne b" title="M" age="33" />
      </guests>
    </reservation>
  </reservations>
</request>
```

15.2 Response

```
<response status="ok"><amendId>123</amendId></response>
```

Parameter	Type	Values	Comments
amendId	Int	identifier to be used for "amendment-status"	

16. Amendment Status Check

16.1 Request

```
<request>
  <auth>
    <username>mishor</username>
    <password>123456</password>
    <agent>i am an agent</agent>
  </auth>
  <method>amd-stat</method>
  <amends>
    <amendId>5</amendId>
  </amends>
</request>
```

Parameter	Type	Values	Comments
method	String	amd-stat	
amendId	Int	the id from the amendment response	

Note: Multiple “amendId” nodes can be sent in one request

16.2 Response

```
<response status="ok">
  <reservations>
    <reservation id="284" amendId="5" amendStatus="2">
      <param name="hotelid">123</param>
      <param name="clientprice">132</param>
      <param name="reservationReference">9</param>
      <param name="checkout">2011-06-05</param>
      <param name="provider">dude</param>
      <param name="cancellation">2011-05-05</param>
      <param name="comments">comment</param>
      <guests ord="2">
        <guest firstName="Ozzy a" lastName="Osborne a" title="M" age="33" />
        <guest firstName="Ozzy b" lastName="Osborne b" title="M" age="33" />
      </guests>
    </reservation>
  </reservations>
</response>
```

Parameter	Type	Values	Comments
reservation @amendStatus	Int	see "Amendment Status" section	
hotelid	Int	the changed hotelid	if applicable
reservationReference	String	the changed reservation reference	if applicable
checkin	Date		if applicable
checkout	Date		if applicable
comments	String	Special requests	if applicable
guests @ord	Int	The room ordinal number	if applicable
guest	Node		if applicable

Note: only relevant nodes will be returned in this response. if a node requested in not returned - it has not been approved.

17. Reservation Status Check

17.1 Request

```
<request>
  <auth>
    <username>mishor</username>
    <password>123456</password>
    <agent>i am an agent</agent>
  </auth>
  <method>res-status</method>
  <reservationId>41</reservationId>
</request>
```

Parameter	Type	Values	Comments
method	String	res-status	
reservationId	Int	the reservation ID	

17.2 Response

```
<response>
  <reservation id="41">
    <bookingdate>2011-08-18</bookingdate>
    <checkindate>2011-09-25</checkindate>
    <checkoutdate>2011-09-28</checkoutdate>
    <hotelid>66020</hotelid>
    <price currency="USD">304.636</price>
    <reservationReference><![CDATA[500416:254]]></reservationReference>
    <status>Confirmed</status>
    <cancellationPolicies>
      <cancellationPolicy startDate="2011-09-20" endDate="2011-09-28" nightsPenalty="1" />
    </cancellationPolicies>
    <boardcode>RO</boardcode>
    <roomname><![CDATA[Twin Room]]></roomname>
    <roomcount>1</roomcount>
    <remarks><![CDATA[Rollaways are based upon availability and at a fee for they are not guaranteed.
    Payable through HOTELBEDS USA, acting as agent for the service operating company, details of which can be
    provided upon request. VAT: 59-2952685. Reference: 254-500416.
    ]]></remarks>
```

```

<guests>
  <guest name="sdf" lastname="dfgdf" type="Adult" age="33" ord="1" />
  <guest name="sdf" lastname="dfgdf" type="Adult" age="33" ord="1" />
</guests>
</reservation>
</response>

```

Parameter	Type	Values	Comments
reservation @id	Int	the ID of the reservation	
bookingdate	Date		
checkindate	Date		
checkoutdate	Date		
hotelid	Int		
price @currency	Currency		
price	Float	the room price	
reservationReference	String		
status	String		See available statuses section
cancellationPolicies	Node		See cancellation policies section
boardCode	String		See board code section
roomname	String		
roomcount	Int	How many rooms are in this reservation	
guests	Node		see guests section
remarks	String		

18. Currency Rates

18.1 Request

```
<request>
  <auth>
    <username>user</username>
    <password>pass</password>
    <agent>agent</agent>
  </auth>
  <method>currency-rates</method>
  <currencies>
    <currency from="EUR" to="USD" />
    <currency from="AUD" to="USD" />
  </currencies>
</request>
```

Parameter	Type	Values	Comments
method	String	currency-rates	
currency @from	String	From currency	
currency @to	String	To currency	

18.2 Response

```
<response>
  <currencies>
    <currency from="EUR" to="USD">
      <rate date="2012-01-19">1.2853</rate>
    </currency>
    <currency from="AUD" to="USD">
      <rate date="2012-01-19">1.04191</rate>
    </currency>
  </currencies>
</response>
```

Parameter	Type	Values	Comments
rate @date	Date	date of the currency value	

Instant

rate	float	Rate	
------	-------	------	--

Appendix A - Static Data and Lookup Tables

1. Guest Titles

Code	Value
M	Mr.
F	Mrs.
MS	Ms.
C	Child
DR	Dr.
PR	Professor

2. Reservation Statuses

Code	Value
1	Confirmed
2	Cancelled by client
3	Cancelled by hotel
8	Pending
10	Rejected
11	Failed pending
0	Failed

3. Board Types

Code	Value
RO	Room Only

BB	Breakfast
HB	Half Board
FB	Full Board
AI	All inclusive

4. Amendment Statuses

Code	Value
0	Pending
1	Rejected
2	Confirmed
3	Partly Confirmed
4	New
5	Deleted

5. Reservation Statuses

Status
Confirmed
Pending
Rejected
Cancelled

6. Active Currencies

Currency

USD
EUR
AUD
INR
NOK
BRL
CAD
CLP
CNY
NZD
DKK
HKD
ILS
JPY
KWD
ZAR
CHF
MYR
MXN
PHP
PLN
QAR
RUB
SAR
SGD
GBP
SEK
THB
TRY
AED

Appendix B - General

1. Contact and Support

For all technical questions, send an email to xml@innstant.com

Other matters can be discussed with your account manager

2. Terms of use

All and any use of this documentation is made under & subject to the agreements made between you and the Innstant Group in general and specifically the Non disclosure agreement & confidentiality agreement and the commercial agreement.

All rights & title of this documentation is fully reserved to the Innstant Group.

3. Changes to the API

Version	Change	Date
3.13	Added a mandatory "Get Voucher Details" call	2011-10-10
3.20	<ul style="list-style-type: none">• Added "Inventory" to search• Added minsuggestedroomusdprice, minnoncommissionable, minroomusdpricelimit, minusdroomnonbillableprice, commissionable to city poll response• Added provider_price_minimum, price_suggested to hotel poll response	2012-01-29
3.22	Added noPolls, searchTimeout and returnResults on city search and city poll.	2012-02-20
3.28	Added documentation for currency rates	2012-05-15
3.29	Added provider response to city search	2012-05-31
3.30	Added active currency list	2012-06-03
3.31	Added aetherref to booking response	2012-06-13
3.32	Added "splitbooking" node to voucher-data call	2012-07-15
3.40	Added multi currency support	2013-01-10