Best Practise Recommendations

1. Price And Availability Searches For Large Cities

If searching for those cities for which we have a large number of properties it is recommended to limit the results by forcing users to select a location or specific hotel(s). The cities for which we recommend this practise are:

Paris
London
Honolulu
Rome
New York
Barcelona
Bangkok
Orlando

If you do not limit your search criteria you might experience excessive response times, depending on the number of room types and the number of nights you are requesting for these destinations.
N.B.: This only applies to hotel price and availability searches.

2. Static Data Storage

All static data, especially item information, must be stored on your local server. We recommend that you update the static data every 4 to 6 months.
In order to retrieve most of the static data we recommend that you write a small program that enables you to first retrieve all country codes, then uses the country codes to retrieve all city codes, then uses the city codes to retrieve all item codes and finally uses the item codes to retrieve all item information.
If you have written a program to download all static date we suggest that when sending asynchronous requests to us, you wait until you have received a response for your first request before sending the second request.
A better solution to collect and update the static data would be to store it locally with a "last updated" timestamp. If a customer requests any item information the local database should be checked. If the item information is present and the timestamp is not older than a set number of days (we suggest 150 days) the information from the local data base is retrieved. If the item information is not present or if the timestamp is older than 150 days, the item information is requested from our server and saved in your local database with an updated timestamp.
We may also inform you of major changes to the information (e.g.: a change in category or star rating) by email and we would suggest that you then update this specific item immediately.

Since March 2009 we also offer a new method of downloading the static data. The Item Information Download request provides the client with a file containing all our static data. Currently it is only available for the hotel module. It is possible to request a file for a complete download of all our static data, a file for incremental updates and a file for yesterday's incremental updates. Specifications for this new functionality can be found in chapter 34 of the Static Data documentation.
Both methods of downloading static data, the SearchItemInformationRequest and the ItemInformationDownloadRequest, are available and can be used in conjunction or independently from each other.

3. Requesting Specific Properties

If you do not want to make a full city search to retrieve the available properties and prices, but would like to enquire for specific properties we recommend that instead of sending us a separate request for each property you merge these individual requests into one XML message. This is advisable for up to 25 properties. If you are requesting more than that we suggest to merge your requests into two (or more) XML messages.

The same logic should be applied if you are requesting prices and availability for multiple room types for specific properties; e.g.: you have three people travelling and you want to display prices and availability for a single room, a double room and a triple room so your customers can choose from the various combinations.

4. Compression

GTA does support compression and using it would be beneficial for you for the following reasons:

- It will increase the speed with which we send responses back to you
- It will reduce utilisation of your communication line
- Through the testing we have done we have seen a decrease of between 16% and 68% of the network time (internet)

We do support both, deflate and gzip. In order to receive a compressed response you will need to add either of the following to the HTTP-Header of your XML request:

- 'Accept-Encoding: gzip'
- 'Accept-Encoding: deflate'

Following is an example of HTTP-Header with compression:

POST /gtaapi/RequestListenerServlet HTTP/1.1
Content-Type: text/xml; charset=utf-8
Accept-Encoding: gzip
Content-Length: 717
Expect: 100-continue
Connection: Keep-Alive
Host: interface.demo.gta-travel.com

Please note that:

- We are not compressing responses that are smaller than 1024 bytes
- You will need to check the content length when reading the HTTP-Header

5. Time-Out For Booking Confirmations

If you are only requesting prices and availability for immediately available properties you might have set a time-out value at your end by which time you would expect to have received a confirmation from us. If this is the case and you no longer require a confirmation after the set time-out value you MUST send a cancellation request to us otherwise we will return a confirmation at a later stage and you would be liable for payment.

6. Failed Booking Requests

If you have any failed booking requests for which you did not receive a response at all or for which you received ANY error, specifically:

```
<Errors>
  <Error>
    <ErrorId>XML0017</ErrorId>
    <ErrorText><![CDATA[The following TOKEN was not found: REF_123-12345678912345]]>
    </ErrorText>
  </Error>
```

```
    </Errors>
```
you MUST send a cancellation request if you no longer require the booking. Occasionally it is possible that we processed and confirmed your request and you were unable to receive the response and would be liable for payment.

7. Validation

Your system should perform all necessary validation to ensure only 'correct' data XML messages are passed through to us. We do not expect to receive invalid requests from you (e.g. invalid dates, invalid room combinations, too many nights).

8. Future Enhancements to the API

Although we aim to ensure that any enhancements made to the API are backwards compatible we may occasionally add new elements or attributes. We would suggest that you write your code in such a way that any unknown elements and attributes are being ignored. This is in case you do not have the time or resource to incorporate any new elements or attributes into your code between the time we notify you of them and the time we install them onto the server.

9. Merging of different message types

Different requests should not be merged in the same message; e.g.: a <SearchHotelPriceRequest> with a <HotelPriceBreakdownRequest>, as this can impact the response time or even cause an error. The routing of messages is dependent on the content of the request, the multi-type requests will be handled in a different, less efficient manner to simple messages. In future releases we will remove the functionality to send different request types within one XML message.
For clarification purposes there is no issue with multiple requests of the same request type being merged in one message although you should be aware of the fact that if you merge more than 25 sub-requests you will encounter performance issues. The maximum number of sub-requests which can be sent is 100. If more than 100 sub-requests in a single XML message are sent, the following error will be received:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
   <Errors>
      <Error>
         <ErrorId>XML0012</ErrorId>
         <ErrorText><![CDATA[Number of sub-requests cannot be more than 100]]></ErrorText>
      </Error>
   </Errors>
</Response>
```

10. Host Header

Please ensure that the host header is set to the host name that you're posting the request to, and not any intermediate proxy or anything else, e.g.: If you are posting to https://interface.gta-travel.com/gtaapi/RequestListenerServlet ensure the host value in the request header is set to 'interface.gta-travel.com' "

13. Schema errors

Whenever we return a HTTP-400 error you will need to check the body of the error. This contains a valid XML response that advises you of the fact that a schema error has occurred. It also informs you of the nature of the schema error which should make it easy for you to debug your code and correct the issue.

14. Forbidden error 403

Whenever we return a HTTP-403 error you will need to check the body of the error. On most occasions this will contain a valid XML response informing you of the nature of the error which should make it easy for you to debug your code and correct the issue.

15. Price and Availability Cache

We now offer three different requests to provide clients with the ability to download a XML file which contains the price and availability data through the API: the <PriceCacheRequest>, the <AvailabilityCacheRequest> and the <PriceAndAvailabilityCacheRequest>.

We do recommend that cache requests are only performed for major destinations. Performing cache requests for destination for which only a small amount of queries is received per day is not efficient. For smaller destination a real time availability and price search will be more efficient. Please note that the more requests are sent in a day the slower the response time will be.

Each requests can be for a maximum of 50 cities.

These three searches can only be used by clients on participation level 0.

Following are some useful information:

- Each reports is available for a 24 hours period on our servers; if the same request (with identical criteria) is sent within a 24-hour period our system will push back the response which was generated for the first request with those criteria; please note that all responses with the same criteria requested within a 24-hour period will have the same token reference of the first response which was generated as this is the response which will be re-pushed;
- One currency can be specified per request; if no currency is specified in the request the rates will be returned in the default currency set up for the XML ID;
- The maximum date range which each cache request can be perform for is 31 days;
- We do identify in the <PriceCacheRequest> and the <PriceAndAvailabilityCacheRequest> if rates are applicable for a minimum number of nights;
- We do identify in the <PriceCacheRequest> and the <PriceAndAvailabilityCacheRequest> if rates are applicable for a minimum number of pax;
- Identical request except with for a different nationality can be sent;
- The maximum start deferral is 330 days;
- The maximum number of requests which can be sent in parallel is 50; our system does counts all files still being processed and puts the requests in a queue system. No new requests should be sent until responses for all the requests sent in parallel have been received.

Please note that some of these set ups are configurable under certain circumstances.

The frequency which an <AvailabilityCacheRequest> can be performed for are as follows:

- If the date range is less than 31 days this request can be performed every 4 hours
- If the check in is greater than 29 days in the future and the date range is less than 91 days this request can be performed every 12 hours

- If the check in is greater than 89 days in the future and the date range is less than 361 days this request can be performed every 12 hours

The frequency which a <PriceAndAvailabilityCacheRequest> and a <PriceCacheRequest> can be performed for are as follows:

- If the date range is less than 361 days this request can be performed every 24 hours.