JacTravel

Project Title: Caching Files for Availability and Price download

Project Summary: Provide a static method to push JacTravel Availability and Pricing to the external clients platforms.

Business Requirement

Tuesday, 29 January 2013

Company Confidential

Version Number 1.0

Document control

1.1 Reference documents

Title	Date	Author	Doc type
Caching Files for Availability and Price download	20/06/08	JAC Travel	Business specification

1.2 Revision History

Version	Date	Author	Comments
1.0	20/06/08	JAC Travel	1st Draft copy of specification

1.3 Change Authorisation

Position	Date	Signature

1.4 Circulation

Name	Company	Position

1.5 Document Approval

Name	Signature	Signoff Date

1 Project overview

The scope of this document is to provide a mechanism to push Availability and Price to our clients.
JacTravel will create several files every day extracting from JacTravel Database the latest data related to the Availability and all the Pricing Offers.
The combination of the information contained in these files will provide to clients the capability to generate the same price as if they were interrogating JacTravel XML API or JacTravel Website.
The first file called Availability_ddmmyyyy.csv will provide the daily availability and net price for all the hotels.
The second file called SpecialOffers_ddmmyyyy.csv will provide all the active offers for all the hotels.
The third file is the Child Policy file that describes the ID and Name of the Child Policy we include into the Availability file.
The fourth file is the Cancellation Policy file that describes the ID and Name of the Cancellation Policy Policy we include into the Availability file.
The fifth file is the Meal Plan file that describes the ID and Name of the Meal Plans we include into the Availability file.
The sixth file is the Minimum Stay file will provide all the active Minimum Stay offers for all the hotels.
These files will be posted in JacTravel FTP site for clients to download around 4 a.m. London time.
These files will cover a period from Today till the next XXX (90, 180) days.

Please Note that these files will be for SINGLE CUSTOMER ONLY. So if we need to provide the cache files to 10 customers we will need to extract 10 different cache files and will be created only for specific customers depending on their volumes and traffics.

2 Availability File

The name of the file will be Availability_ddmmyyyy.csv. The date will give the possibility to pick the correct file everyday in case there are multiple files available into the FTP directory.
The format is a CSV file for speed and memory size.
An example of the file will be provided in the Appendix at the end of the document.
Here the logical view of the data, a sort of column name that will be not included in the file.

```
BOF NewLine
serviceid,date
,totalmaxroom,totalsingle,totaldouble,totaltwin,totaltr
iple,totalquad,totalfam1,totalfam2,currency,
cancellationpolicyid,|,optionid,#,
roomname1,#,occupancy_id,#,price,#,mealplan,#
,child_policy_id,#,minimumstayid,;,
optionid,roomname2,#,occupancy_id,#,price,#,mealplan,
child_policy_id,;,………………………………
roomnamen,#,occupancy_id,#,price,#,mealplan,|,NewLine
EOF
```

The rules for creating this file are:

1. The file is a CSV (comma separate value)
2. See this example `|single deluxe#1#60#bed and breakfast;single superior#1#70#continental|`

3. We use | (pipe character) to start and finish a multi elements field
4. We use ; (semicolon) to separate elements within the multi elements fields separated by 2 pipes
5. We define an element an object that represent a sets of options separated by #
6. A multi element is created when we have a variable numbers of elements depending on the Service (hotels) we are representing. Some Service has 2 elements some other can have 10. This number is variable. We have multi elements to represent different room names with price.
7. The child policy is expressed as an ID and we will provide the full list of the Child Policy with IDs in a way to map it properly.
8. If there is Availability for one room type then there have to be at least one element in the room name multi-element with the correspondent occupancy_id
9. In case there is availability for one of the 2 family rooms (Occupancy id 7 and 8) then there has to be the child policy id populated. If there are no family rooms available then the child policy id will be 0.

2.1 Data Definition

FIELD NAME	DATA TYPE	MANDATORY	EXAMPLE	NOTES
BOF	String		BOF	BOF describe the start of the file
NewLine				NewLine is the end of a BOF
service_id	Integer	Y	15765	Unique ID of the Service/Hotel
date	Date Type	Y	01-JUN-08	dd-MMM-yy
totalmaxroom	Integer	Y	5	This field shows the max numbers of rooms are available for a specific day
totalsingle	Integer	Y	3	This field shows the max numbers of rooms are available for a specific day for this room type
totaldouble	Integer	Y	3	This field shows the max numbers of rooms are available for a specific day for this room type
totaltwin	Integer	Y	3	This field shows the max numbers of rooms are available for a specific day for this room type

totaltriple	Integer	Y	3	This field shows the max numbers of rooms are available for a specific day for this room type
totalquad	Integer	Y	2	This field shows the max numbers of rooms are available for a specific day for this room type
totalfam1	Integer	Y	2	This field shows the max numbers of rooms are available for a specific day for this room type
totalfam2	Integer	Y	2	This field shows the max numbers of rooms are available for a specific day for this room type
currency	String	Y	Sterling	Identify the currency of the price
CancellationPolicyID	Integer	Y	10	This field shows the cancellation policy applied to the service
Start of the first Multi Element for the list of all the rooms that are sold at the hotel	Pipe separator		|	We are using pipe as the start and the end of an element
option_id	integer	Y	78465	Unique Ids of the room sold by the hotel
Field separator	#	Y	#	We are using # as the separator of an field
roomname1	string	Y	Double Deluxe	Name of the room sold by the hotel
Field separator	#	Y	#	We are using # as the separator of an field
occupancy_id	integer	Y	2	The ID of the room type the room name belong to
Field separator	#	Y	#	We are using # as the separator of an field
price	Float	Y	30.75	Price Per Passenger
Field separator	#	Y	#	We are using # as the separator of an field
mealplan_id	string	Y	Continental	The description of the meal plan included in the price
Field separator	#	Y	#	We are using # as the separator of an field
child_policy_id	Integer	Y	456	This field shows the Child policy ID
Field separator	#	Y	#	We are using # as the separator of an field
Minimum stay rule ID	Integer	N	12	This field shows the minimum stay rule for this option
Finish roomname1	semicolon	Y	;	We are using semicolon as the separator of an element

Field separator	#	Y	#	We are using # as the separator of an field
option_id	integer	Y	78466	Unique Ids of the room sold by the hotel
Field separator	#	Y	#	We are using # as the separator of an field
roomname2	string	Y	Double Superior	Name of the room sold by the hotel
Field separator	#	Y	#	We are using # as the separator of an field
occupancy_id	integer	Y	2	The ID of the room type the room name belong to
Field separator	#	Y	#	We are using # as the separator of an field
price	Float	Y	40.75	Price Per Passenger
Field separator	#	Y	#	We are using # as the separator of an field
mealplan_id	string	Y	Continental	The description of the meal plan included in the price
Field separator	#	Y	#	We are using # as the separator of an field
child_policy_id	Integer	Y	456	This field shows the Child policy ID
Field separator	#	Y	#	We are using # as the separator of an field
Minimum stay rule ID	Integer	N	12	This field shows the minimum stay rule for this option
Finish roomname2	semicolon	Y	;	We are using semicolon as the separator of an element
Field separator	#	Y	#	We are using # as the separator of an field
option_id	integer	Y	78470	Unique Ids of the room sold by the hotel
Field separator	#	Y	#	We are using # as the separator of an field
roomname**n**	string	Y	Triple	Name of the room sold by the hotel
Field separator	#	Y	#	We are using # as the separator of an field
occupancy_id	integer	Y	4	The ID of the room type the room name belong to
Field separator	#	Y	#	We are using # as the separator of an field
price	Float	Y	50.75	Price Per Passenger
Field separator	#	Y	#	We are using # as the separator of an field
mealplan_id	string	Y	B&B	The description of the meal plan included in the price

Field separator	#	Y	#	We are using # as the separator of an field
child_policy_id	Integer	Y	456	This field shows the Child policy ID
Field separator	#	Y	#	We are using # as the separator of an field
Minimum stay rule ID	Integer	Y	12	This field shows the minimum stay rule for this option
End of the Multi Element for the list of all the rooms that are sold at the hotel	Pipe separator		|	We are using pipe as the start and the end of an element
NewLine				NewLine is the end of a record
EOF	String		EOF	EOF describe the end of the file

3 Special Offer File

The name of the file is SpecialOffers_ddmmyyyy.csv and it will provide all the active offers for the specific hotels.
The date will give the possibility to pick the correct file everyday in case there are multiple files available into the FTP directory.
The format is a CSV file for speed and memory size. An example of the file will be provided in the Appendix at the end of the document.

The rules for creating this file are:

1. Each row represent "an individual special offer for a specific hotel for the specified interval period"
2. In case an offer is valid for one day only Valid From and Valid To will have the same date.

3.1 Data Definition

FIELD NAME	DATA TYPE	MANDATORY	EXAMPLE	NOTES
ServiceID	Integer	Y	15765	Unique ID of the Service/Hotel
ServiceName	String	Y	Bernini Bristol Hotel	Name of the Service/Hotel
RuleValidFrom	Date Type	Y	2009-04-01 00:00:00.000	yyyy-mm-dd
RuleValidTo	Date Type	Y	2009-08-31 00:00:00.000	yyyy-mm-dd
Group separator	#	Y	#	We are using # as the separator of a group
Room types	String	Y	Quad Room	Rooms to which the special is applicable each separated with a coma and the whole list of rooms starts and ends with #

Group separator	#	Y	#	We are using # as the separator of a group
Specials	String	Y	20% off Sunday night for 3 night stay	The name of the Specials
SpecialsDescription	String	Y	20% off Sunday night for 3 night stay arriving Friday, Saturday or Sunday	The Message for the Users to describe the Specials
MinimumStayNights	Integer	N	Values from 1 to 25	How many nights you have to stay to get a full night free or a discount. Min 1 max 25
MinimumPax	Integer	N	Values from 1 to 99	Minimum pax required to avail the offer. Min 1, max 99
CalculationUnits	Integer	Y	1	These are the units based on which supplements will be calculated. This value must be read in relation to the next two fields - Criteria and Offer. Assuming CalculationUnits is 1, If the criteria field is Night/Day and Offer field is Free, it would imply 1 nights free.

Criteria	Integer	Y	Min 1, max 4	Night/Day = 1
				Unit = 2
				Passenger = 3
				Per Unit = 4

Offer	Integer	Y	Min 1, Max 6	Free = 1 Reduced by % = 2 Reduced by amount = 3 Other = 4 Supplement % = 5 Supplement Amount = 6
Amount	Double	Y	3	Will be 0.0 if not present. This value will be greater then zero only if Offer is one of the following (Reduced by %, Reduced by amount, Supplement %, Supplement Amount) This value must be read in relation to the previous three fields – CalculationUnits, Criteria and Offer. Assuming CalculationUnits is 1, Offer is 2 (Reduce by %), and Crietria is 1 (Night/Day), If the Amount field is 3, it would imply 3% reduction on the selling price of 1 night. Which night is decided by the value of OfferNight or BookingNight fields. Would be formatted to 2 decimal places.
MaximumLimit	Integer	Y	1	How many time an offer can be applied in a specific booking period. So if you have a 9 days stay and you have a Stay3Pay2 offer available, if you have Maximum = 1 means that the offer is applied only one time. If you have Maximum = 2 , means 2 times.
OfferNight	String	N	Mon to Sun or -1	This is the day of the week on which offer is applied. Assuming CalculationUnits is 1, Offer is 2 (Reduce by %), Crietria is 1

				(Night/Day), Amount field is 3, if the OfferNight field has a value “Sun” it would imply 3% reduction on the selling price of that night which is a Sunday assuming one of the week day between booking start date and booking end date is a Sunday. If there is no Sunday between booking start and end date, this offer will not be applicable. If OfferNight value is present, BookingNight value would not be present and vice versa. That is both are mutually exclusive. If Offernight and Bookingnight is -1, it would imply that the supplement is applied to the whole booking provided booking start date >= RuleValidFromDate and Booking end date <= Rule valid to date. If booking start date is < RuleValidFromDate or Booking end date > RuleValidToDate or both, then the offer would be applied to only those dates which are intersecting. For example if the booking start date is 14th of April and booking end date is 16th of Apriil whereas RuleValidFrom date is 10th of April and RuleValidTo date is 14th of April, then the supplement would apply to only 1 intersecting night that is the night of the 14th of April.
BookingNight	String	N	1 to 25	This is the night of the booking on which offer is applied. Assuming CalculationUnits is 1, Offer is 2 (Reduce by %), Crietria is 1 (Night/Day), Amount field is 3, if the BookingNight field has a value 1 it would imply 3% reduction on the selling price of first night. If BookingNight value is present, OfferNight value would not be present and vice versa. That is both are mutually exclusive. If Offernight and Bookingnight is -1, it would imply that the supplement is applied to the whole booking provided booking start date >= RuleValidFromDate and Booking end date <= Rule valid to date. If booking start date is < RuleValidFromDate or Booking end date > RuleValidToDate or both, then the offer would be applied to only those dates which are intersecting. For example if the booking start date is 14th of April and booking end date is 16th of Apriil whereas RuleValidFrom date is 10th of April and RuleValidTo date is 14th of April, then the supplement would apply to only 1 intersecting night that is the night of the 14th of April. Assuming offer night is 1 the offer would be applied to the 1st night of the intersecting dates in case if booking start date is < RuleValidFromDate or Booking end date > RuleValidToDate or both. If however booking start

				date >= RuleValidFromDate and Booking end date <= Rule valid to date then the offer would be applied to the first night of the booking.
ArrivalDays	String	Y	Fri/Sat/Sun	Display the days you have to arrive in the hotel to avail the offer. In case all days are ok than you will see ANY. Multi days are accepted.
StayDays	String	Y	Sun	Display which day you have to stay at the hotel to get the offer. In case all days are ok than you will see ANY. Multi days are accepted.

4 Appendix

4.1 Availability File example

Include the example

4.2 Special Offer File example

Service ID	**Service Name**	**Valid From**	**Valid To**	**Room Types**	**Specials**	**Special Description**	MinimumStayNights	MinimumPax	CalculationUnits	Criteria	**Offer**	Amount	MaximumLimit	OfferNight	BookingNight	ArrivalDays	StayDays
13199	ZZZ Test Extranet Hotel	2009-04-01 00:00:00.000	2009-08-31 00:00:00.000	Single Room, Double Room, Twin Room	Stay 5 nights pay for 4	Stay 5 nights pay for 4	5	1	1	1	1	0.00	1	-1	1	Any	Any

4.3 Price Logic Calculation

In this chapter we are going to explain the business logic that is used to calculate the price of a booking using the Availability and Price file and the Special Offer files. The following scenario will cover all the possibility we have in our booking system so you'll be able to reproduce locally in your environment and make the best use of the Cache.

Scenario:
One single room booked for 5 nights at "ZZZ Test Extranet Hotel" from 15th June 2009 to the 20th of June 2009.
A supplement "Stay 5 nights, pay for 4" is in force from 1st of April 2009 till 30th of Aug 2009

Below id the sell rate for "ZZZ Test Extranet Hotel" for the stay period:
15th June – £ 100
16th June – £ 100
17th June – £ 100
18th June – £ 100
19th June – £ 100

The Special offer cache file would contain a coma separated list of values for the ZZZ Test Extranet Hotel as follows:

13199,ZZZ Test Extranet Hotel,2009-04-01 00:00:00.000,2009-08-30 00:00:00.000,#,Single Room, Double Room, Twin Room,#,Stay 5 nights pay for 4,Stay 5 nights pay for 4,5,1,1,1,1,0.00,1,-1,1,Any,Any

Following are the steps one would have to follow to calculate special offers and derive an accurate pricing:

Step 1: Figure out the price per day between booking start date (15th of June) and booking end date (20th June) for the service having ID 13199. This would be done by parsing the PriceAndAvialiability cache file. In this case the price per day of the booking turns out as £ 100.00 per day as follows:
15th June – £ 100
16th June – £ 100
17th June – £ 100
18th June – £ 100
19th June – £ 100

Step 2: Next step is to determine if any special is applicable for the service in question. This would be done by parsing the SpecialOffers cache file for the service 13199.
One or more specials could be applicable between booking start date and booking end date. This can be determined by comparing the booking start date and end date with the RuleValidFrom and RuleValidTo dates in the cache file.
If more then one rules apply to a given booking, the top 1 is selected. Top1 rule would be the one with minimum RuleValidFrom.
For example
If 2 rules apply between the booking start date 15th June and booking end date 20th June. The first rule has a RuleValidFrom date as 16th June and RuleValidTo date is 18th June and 2nd rule has a RuleValidFrom date as 19th June and RuleValidTo date as 25th June, the 1st rule would be applied as that is the rule having minimum RuleValidFrom date among the two rules.

In the above example only one rule is applicable between booking start and booking end date and the rule name is "Stay 5 nights, pay for 4".

Step 3: Check if the rule in question is applicable to the room type being booked. The SpecialOffers cache file would contain room type names to which the rule in question applies. The room type names begin and end with a hash (#) symbol.
In scenario presented above the rule "Stay 5 nights, pay for 4" applies to Single Room, Double Room, Twin Room.

Step 4: Check if the value of "MinimumStayNights" in the SpecialOffer cache file is more than or equal to the number of nights in the booking. If the number of booking nights is less then the value of "MinimumStayNights" in the SpecialOffer cache file, the Special would not apply. In above example the "MinimumStayNights" is "5" and the booking number of nights is also equals to 5.

Step 5: Check if the "MinimumPax" which can be obtained by parsing the SpecialOffers cache file is greater then or equal to the number of passengers in the booking. If the total number of passengers in the booking is less then the value of "MinimumPax" in the SpecialOffers cache file, then the rule would not apply. In the above example the "MinimumPax" is "1" and the number of passengers in the booking is also "1"(as we are booking single room)

Step 6: In the next step "ArrivalDays "and "StayDays" would be verified. "Arrival Days" stands for the days you have to arrive in the hotel to avail the offer. Arrival days would be presented either as a blank space (' ') separated list of week days such as "Mon Tue Wed" or there would be an "Any" word. This is the same as check-in week day. To verify the "ArrivalDay", check what is the week day on the check-in date (i.e. Mon, Tue etc). In the SpecialOffers cache file if the value is "Any" for "ArrivalDays", it would mean any arrival day. In other words your code should disregard or ignore ArrivalDays value in the SpecialOffers cache file. If however the value of "ArrivalDays" is "Sun", "Mon" etc., you need to compare the week day on the booking start date to the value of "ArrivalDays" in the cache file. If both match, then the special would apply. If not the Special would not apply. That is if the week day on the check in date is "Mon" and if the value of "ArrivalDays" in the SpecialOffers cache file is "Sun", the Special would not apply. In the scenario presented above the checin date of the booking is 15th of June 2009 which is a "Mon" and the value of "ArrivalDays" in the SpecialOffers cache file is "Any". This means the code will ignore the check in day comparison.
In a similar fashion "StayDay" would imply the week days that you would occupy the room in the hotel during your stay. "Staydays" would be presented either as a blank space (' ') separated list of week days such as "Mon Tue Wed" or there would be an "Any" word. It is not the same as Check-in days (or "ArrivalDays"). Your check-in day could be Monday, but the days you occupy the hotel (that is the days of the week on which you actually occupy the room) could be from Monday to Thursday. To verify the "StayDay" check what are the week days for each date between booking start date and booking end date. The Special offer would be availed if any one the week days between booking start and booking end date match the week days specified in the SpecialOffers cache file. If none of the week days mentioned in the SpecialOffers cache file match the week days between bookings start date and booking end date, then the special would not apply. In the SpecialOffers cache file if the value is "Any" for "StayDays", it would mean any stay day. In other words your code should disregard or ignore StayDays value in the SpecialOffers cache file.

Steps 2 to 6 are validation steps wherein your code should validate the fact whether the special would be availed or not. If at the end of step 6 your code determines that the special can be availed, you need to write code for steps 7 to 10 to actually calculate the special offer. If any of the steps 2,3,4,5 and 6 fail the validation it would mean that Special cannot be availed.

To calculate the final price, follow the following steps.

Step 7: Verify the values of "OfferNight" and "BookingNight" from the SecialOffers cache file. Both are mutually exclusive and both are optional that is either one (or both) can have a value -1. If "OfferNight" has -1 as a value and "BookingNight" has a proper value, it would imply that your code should ignore "Offernight" and vice versa. It can happen that both "BookingNight" as well as "OfferNight" can have -1 value. This would imply that the special would be applied to the whole stay as explained later.
"Offernight" is the day of the week on which offer is applied. If the "OfferNight"=1 this means that the offer will be calculated on Sale Price of Monday. If it is 2 the offer will be calculated on Sale Price of Tuesday and so on. Assuming CalculationUnits is 1, Offer is 2 (Reduce by %), Crietria is 1 (Night/Day), Amount field is 3, if the OfferNight field has a value "Sun" it would imply 3% reduction on the selling price of that night which is a Sunday, assuming one of the week day between booking start date and booking end date is a Sunday. If there is no Sunday between booking start and booking end date, this offer will not be applicable. If both Offernight and Bookingnight are -1, it would imply that the supplement is applied to the whole booking provided booking start date >= RuleValidFromDate and Booking end date <= Rule valid to date. If booking start date is < RuleValidFromDate or Booking end date > RuleValidToDate or both, then the offer would be applied to only those dates which are intersecting. For example if the booking start date is 14th of April and booking end date is 16th of April whereas RuleValidFrom date is 10th of April and RuleValidTo date is 14th of April, then the supplement would apply to only 1 intersecting night that is the night of the 14th of April.

"BookingNight" is the night of the booking on which offer is applied. It could be any value between 1 and 99 both inclusive. Assuming CalculationUnits is 1, Offer is 2 (Reduce by %), Crietria is 1 (Night/Day), Amount field is 3, if the BookingNight field has a value 1 it would imply 3% reduction on the selling price of first night. If BookingNight value is present, OfferNight value would not be present and vice versa. That is both are mutually exclusive. If Offernight and Bookingnight is -1, it would imply that the supplement is applied to the whole booking provided booking start date >= RuleValidFromDate and Booking end date <= Rule valid to date. If booking start date is < RuleValidFromDate or Booking end date > RuleValidToDate or both, then the offer would be applied to only those dates which are intersecting. For example if the booking start date is 14th of April and booking end date is 16th of April whereas RuleValidFrom date is 10th of April and RuleValidTo date is 14th of April, then the supplement would apply to only 1 intersecting night that is the night of the 14th of April.
Assuming offer night is 1 the offer would be applied to the 1st night of the intersecting dates in case if booking start date is < RuleValidFromDate or Booking end date > RuleValidToDate or both. If however booking start date >= RuleValidFromDate and Booking end date <= Rule valid to date then the offer would be applied to the first night of the booking.

In the scenario presented above we have "BookingNight"=1 and hence the Special will be applied to the sell price of Night 1 of the booking i.e. 15th June and as a result the sell price on 15th of June would be zero.

Step 8: Next verify the value of "CalculationUnits" in the SpecialOffers Cache file. In our example the "CalculationUnits" =1. These are the units based on which supplements will be calculated. This value must be read in relation to the two fields - Criteria and Offer.
Assuming CalculationUnits is 1, if the criteria field is "Night/Day" and Offer field is "Free", it would imply 1 night free as is the case in the example given above. Note that if the Offer field has the value 1 (Free), Amount field would always be zero. Alternatively if the Offer field has the value 2, or 5 the amount field would need to be treated as a % value. Similarly if the Offer field has a value 3 or 6, then the Amount field would have to be treated as an absolute value which may or may not contain a fractional quantity.

Step 9: Verify the value of "Criteria" field in the special offer cache file. In the example the "Criteria" field has a value 1 which translates to "Night/Day". This implies that 1 night.

Step 10: Next verify the values of "Offer" and "Amount" fields. In above mentioned case the "Offer" is "Free" and "Amount" is 0. This translates to 1 night free. As mentioned earlier, Amount field would have a value only if the Offer field has the value 2, 3, 5 or 6.

While calculating the special offer, following 6 fields are important: "CalculationUnits", "Criteria", "Offer", "Amount", "OfferNight" and "BookingNight".
In scenario presented above, the calculation of special offer would be done in the following manner:
CalculationUnits = 1
Criteria = 1 (which is "Night / Day")
Offer = 1 (which is "Free")
Amount = 0 (Because offer is 1 and hence amount would be ignored)
OfferNight = -1 (since there is -1, this field would be ignored)
BookingNight = 1 (This would translate to 1st night of the booking which is 15th June 2009 in the scenario presented above)

The above translates to 1st night of the booking would be free.

Hence the final price per day would be
15th June – £ 0.00 (Special Availed)
16th June – £ 100
17th June – £ 100
18th June – £ 100
19th June – £ 100

4.4 Child Policy File

In the main Availability and Price file we have used Child Policy ID to simplify the readiness of the file.

Child_Policy_ID	ChildPolicyDescription	FromAge	ToAge	PercentageDiscount
2	3-12=50%	0	2	0%
2	3-12=50%	3	12	50%
2	3-12=50%	13	18	100%
3	0-11=Free	0	11	0%
3	0-11=Free	12	18	100%
4	6-12=50%	0	5	0%
4	6-12=50%	6	12	50%
4	6-12=50%	13	18	100%
5	5-12=50%/13-16=75%	0	4	0%
5	5-12=50%/13-16=75%	5	12	50%
5	5-12=50%/13-16=75%	13	16	75%
5	5-12=50%/13-16=75%	17	18	100%
6	0-12=Free Room Only	0	12	0%
6	0-12=Free Room Only	13	18	100%
7	5-12=50%	0	4	0%
7	5-12=50%	5	12	50%
7	5-12=50%	13	18	100%

4.5 Cancellation Policy File

In the main Availability and Price file we have used Cancellation Policy ID to simplify the readiness of the file.

Field Name	Data Type	Mandatory	Example	Note
Cancellation Policy ID	Integer	Y	2	This is the unique Cancellation Policy ID
Cancellation Policy Name	String	Y	Standard 8 day(s) prior to arrival	This is the Cancellation Policy Name for this Cancellation policy ID
Description	String	Y	8 day(s) prior to arrival	Description of this Cancellation Policy
Start of the first Multi Element for the details of Cancellation Policy	Pipe separator		|	We are using pipe as the start and the end of an element
Number Of Days	Integer	Y	8	This specifies number of days prior to booking, during which this cancellation policy will be in effect
Field separator	#	Y	#	We are using # as the separator of an field
Amount	Decimal	Y	100.00	Depending on “Charge Type” this specifies either: 1. Percentage 2. Number of days 3. Fixed dollar amount per option booked
Finish Details	**semicolon**	**Y**	**;**	**We are using semicolon as the separator of an element**
Start of the first Multi Element for the details of Cancellation Policy	Pipe separator		|	We are using pipe as the start and the end of an element
Number Of Days	Integer	Y	8	This specifies number of days prior to booking, during which this cancellation policy will be in effect
Field separator	#	Y	#	We are using # as the separator of an field
Amount	Decimal	Y	100.00	Depending on “Charge Type” this specifies either: 1. Percentage 2. Number of days 3. Fixed dollar amount per option booked

End of the first Multi Element for the details of Cancellation Policy	Pipe separator		|	We are using pipe as the start and the end of an element
Charge Type	String	Y	Percentage of first night	This specifies the policy type applied. There a currently 4 policy types which can be applied to cancellation policy: 1. Number of nights 2. Fixed fee 3. Percentage of full cost 4. Percentage of first night
NewLine				NewLine is the end of a record
EOF	String		EOF	EOF describe the end of the file

4.6 Meal Plans File

In the main Availability and Price file we have used Meal Plan Ids to simplify the readiness of the file.

Field Name	Data Type	Mandatory	Example	Note
Meal Plan ID	Integer	Y	2	This is the Unique Meal Plan ID
Meal Plan Name	String	Y	Bed & Continental Breakfast	This is the Name of the Meal Plan
Break Fast	String	Y	BREAKFAST	This is when current Meal Plan includes the Break fast
Lunch	String	Y	LUNCH	This is when current Meal Plan includes the Lunch
Dinner	String	Y	DINNER	This is when current Meal Plan includes the Dinner

4.7 Minimum Stay File

In the main Availability and Price file we have used Minimum Stay Rule IDs to simplify the readiness of the file.

Field Name	Data Type	Mandatory	Example	Note
Min Stay Rule ID	Integer	Y	3	This is the unique Min Stay Rule ID
Minimum Stay Rule Name	String	Y	Minimum 5 nights stay	This is the Rule Name for this Rule ID
Minimum Stay Rule Description	String	Y	Minimum 5 nights stay	This is the Description for this Rule
Minimum Nights	Integer	Y	5	This specifies number of nights user will have to stay to book the Option with this rule.